TURING

by Christos H. Papadimitriou

christos@cs.berkeley.edu

©copyright 1998 by Christos Papadimitriou
Turing, Alan Mathison (1912-1954). British mathematician and philosopher, broadly considered the founder of Computer Science. As an undergraduate at King’s College (Cambridge) in 1934 he demonstrated the impossibility of automatic mathematical reasoning, answering one of the paramount open mathematical questions of that age. He did so by devising a simple, yet all-powerful, hypothetical computational device, since called the Turing machine, which has had tremendous influence on computational research and culture. During the Second World War, Turing was instrumental in breaking the German cryptographic code Enigma, a feat that greatly contributed to the victory by the Allies. After the war he worked on launching the British computer industry, did seminal work in Biology, was a pioneer of chess-playing by computer, and defined the field of Artificial Intelligence by proposing the so-called Turing test. He was tried in 1952 for homosexuality, and was condemned by the court to treatment by mind- and body-altering drugs. He died two years later, apparently by his own hand.

(From http://www.netcyclopedia.com.)

Be thou a spirit of health or goblin damned? […]
Be thy intents wicked or charitable?

(From http://www.greatbooks.org/hamlet.suml#actII.)
“ALEXANDROS”

“I will remember his name all right,” Ethel thinks as her airplane leaves behind the northern coast of Scotland. “Alexandros.” Nice. The accent on the c. Classic, but not pompous or obscure. (“Do they still use names like ‘Zeus’ or ‘Iamblichos’ in that crazy country?” she wonders. Probably.) “Alexandros.” After the first boy who wanted to conquer the world. “Alexander” sounds very different. Pretentious in its resistance to “Alex.” Brings to mind czars, junior high teachers, disgusting cocktails—even a former White House chief of staff, for chrissakes.


Ethel has not always remembered the names of her lovers. “Benjamin Yamada,” of course. But names are currency at school, and he was the first, and the gentlest, and the one holding her hand when Dad died, the one she cared for, was comfortable with, agonized about, felt guilty for. Forget that name, damnit. Never look it up again. He’s gone. (For the past six years he’s been living in Cincinnati, unlisted phone number shared with nobody else, surfing the Net, beginner level. Even uses her stuff occasionally.) “No need to feel guilty, he’s probably happier than me,” she thinks. Forget him. Don’t even think of getting in touch, he’ll hurt you like the others if you give him another chance. Next!

“Andrew Leitman” is next. But again, she has a good excuse for remembering. He was her first boss at Exegesis, back when the company was still Webus Inc. Soon to become her lover, her mentor. Then her tormentor, her corporate nemesis. Next, after patient months of maneuvering, her same-level colleague, eventually her close collaborator and buddy. Then her lover once more (“once” here is numerically correct, even a bit of an exaggeration). And finally, the summer of 99, Andy became her victim: humiliated, defeated, decapitated, purged from the company. Never heard from since. The Independence Day massacre. Black widow spiders are more direct, but they probably don’t have as much fun remembering. This must be Greenland, right?

For example, what was the name of the Viking she slept with during exam week of the winter quarter of her senior year at Santa Cruz? She remembers his running shoes, his disease story, his pubic hair (grey New Balance, rehearsed but convincing, blond). But his name? Well, actually, with much effort and mental cross-referencing she can narrow down his first name to either “Kevin”
or “Kenneth.” But this proves the point rather than disprove it, doesn’t it? How about the black guy from Helen and Ralph’s party? Or the nerd from Berkeley with the covert violent streak? No clue.

Ethel of course remembers the names of everybody she desired and never had. High school crushes, all twelve or so of them, her math professor at college (“Dr. Christopher L. Bates”), the shy genius who was her partner in the programming class (“Tommy Ng”), the others, every single painful one of them. Their names sadistically squeezing her neurons, never letting go, associating with faces, figures, clothes, situations, missed chances. Nor can she forget the ones whose names she never knew, but wanted nevertheless. Like the man who was having dinner alone at the Palo Alto restaurant the night she was there alone. He was looking at her in such a way. Short and very skinny, no cheeks, huge eyes, business suit. She never looked at him, but the way she didn’t was equally fiery, she felt. He ordered the exact same things she did, from appetizer to dessert wine. The whole restaurant was hot, diners were breathing heavily, waiters wiping sweat from their eyebrows watching them. She was wet, delirious. He paid and left, still looking at her. “You gave me no chance to forget your name, buddy.”

Ethel is amused to realize that she doesn’t even remember Sola’s real name. Not the whole paragraph, with the de la and the y. Her heart skips a beat as she brings viper-like Sola to mind. The tiny breasts, one slightly tinier than the other, the sweet waist, the National Geographic ass, the legs that go on forever. Her face is harder to recall, so subtle and striking and complex with the ebony hair and the blue-green eyes and the enigmatic smile that so often becomes a menacing grin. Oh, Sola, Sola! Sola, Sola, Sola! Stanford student, Spanish literature. (“Peninsular literature, amor, not Spanish. You lose Borges and Garcia and Onetti, and a couple of poets whose names I forget, but you gain something infinitely more interesting and tangible, a three-year fellowship from the queen.”) Part-time model, she claims. Part-time call-girl, she once confessed. And a full-time liar, Ethel knows. But Ethel never double-checks her on the Net. Sola has total control. Even though, at twenty-seven, she is almost ten years younger than Ethel.

She first met her in her fantasy life, on the Net, in the Colony, years ago. Then they met in real life, just this year. The weekend at the bed-and-breakfast at Point Reyes. Shopping in San Francisco (first Union Square and then Haight-Ashbury). Sola is Ethel’s first serious exploration in this direction. A sexy, fascinating lover with whom you can giggle in restrooms. True love. Sola moves in. Then out. Then in again. True humiliation, true pain. True love. Sola flies home for the spring break. (“The boys in Madrid are soooo boring, amor, and the girls soooo uptight, this is such an easy promise for me to keep.”) Then the telephone call. (“I’ve been back for a week, amor, and have been thinking. I
must think some more. Still confused. And serious school work, amor, serious.”

Ethel didn’t do any work for a week, didn’t log in, not even to reconnect with her fantasy Sola, not even to spy on the real Sola in the Net. The dark week. The barely-less-than-dark week (what was his name, anyway?). Then Sola’s triumphant reentry. The lies. Turbulent April. Black May. Ethel is depressed, demoralized, defeated as never before in her life. That is when she told Sola of her plan.

Not a plan, really. No subgoals, no algorithm. A fantasy. She would spend the second half of June in a Greek island. Carefully timed. Burn a week’s worth of stock. Meet a Greek God. At least a reasonably smart, very healthy, passably good-looking man. “I will bear his daughter,” that’s how she put it. Sola was amused, so faintly sarcastic she sounded almost encouraging. “A child? Are you sure that’s what you are missing, chica?” They hadn’t made love for weeks.

Sola drove her to the airport in Ethel’s white BMW. (“I’ll take good care of our Bum, amor, oil change and all. See you soon, at the gate. Be outrageous. And send me one of those obscene Greek island postcards every day.”) “She is now waking up, stretching, preparing to drive to the airport,” Ethel thinks, her heart skipping another beat. “She will be wearing a long black sweater and black tights, if I know her. Driving the Bum, no dents, clutch still tight.” Gentle chest pain. Or will she? “Stop thinking about it, you’ll go nuts. Take a nap!”

Her plan worked better than she could have realistically hoped. So far anyway — she will know soon. (“But in another sense it has gone terribly wrong” Ethel thinks, remembering the unanticipated complexities, the persistent guilt, the hard dilemmas.) She had been warned by “The Bad Girl’s Guide to the Greek Isles” (http://astro.cam.ac.uk/sarahl/guide/, abysmal Exegesis rating) that Greek men (a) are no longer Gods, and (b) are nowhere to be found in the Greek islands. Except for old fishermen, gay playboys, and the kamaki, the professional lovers usually referred to by the Greek word for harpoon, so rough and naive about everything, especially disease. Apparently she had to choose between German men in Crete, French and Dutch in Paros, Scandinavians in Rhodos, Austrians and Israelis in Ios, Americans in Santorini, Englishmen and Italians in Corfu. (No men in Lesbos. No straight men in Mykonos.) She chose Corfu.

Lucky choice. Convenient flight, beautiful island, charming dilapidated hotel in the old town (the weirdest home page on the Net). Easy lifestyle, no overhead, you fit right in from day one. Around noon the bus to the beach, swimming, sunbathing, late lunch. Late afternoon the bus to the town, late long siesta. Go out at ten. (Ten! At ten, Palo Alto waiters are yawning as they lock up.) A drink, dinner, strolling through town, dancing, coffee, more strolling, more dancing. You can score hashish in some bars, they say there is also coke —
but why bother, the island is a continuous high. You only drink at the more expensive bars, the others can give you a week-long hangover with nasty bootleg. Back to bed in the three-bit morning hours, sleep or whatever until noon. Then on to the beach. And so on. You better compare today’s date with your return ticket once in a while.

At night the town radiates with a strange glow that comes from white cotton cloth on seriously suntanned flesh. Eyes seek eyes all the time, admiring, contemplating, flirting, proposing, teasing, daring. At first you are embarrassed to death, then you have a drink and try it yourself, then you can’t stop. Your eyes seek eyes all the time. Couples in love (almost always straight in this island) hold hands or cross arms, their eyes also seeking eyes everywhere. All the time. The second day you feel you recognize everyone, thousands of them. You can spot the newcomers.

Alexandros was not a newcomer. Ethel had noticed him the previous night, medium tall, slim, mid fifties, long white hair, white beard. Very dark skin, looked to her like a Hindu guru. Clever little eyes. Brown. A grand master of the eyes-seeking-eyes game—but especially interested in her eyes, Ethel remembers, had stared her down, had made her lose her breath, her step. Deep and wide red scars in the small visible parts of both cheeks, the left much larger than the other, probably extending all the way to the chin under the beard—a souvenir from the colonels’ dictatorship in the early seventies, she would find out. Now sitting with a teenage girl at the next table in the outdoors café, occasionally holding hands, talking very fast in the language she could already recognize as Greek. A pretty teenager, a couple of years away from true beauty. A Humbert Humbert and his nymphette.

But wait...isn’t the eroticism between them strictly paternal? The kind she missed in her own adolescence, the kind she’d jealously spied in her friends’ homes? Suddenly, eye contact. No retreat now. “Pleased to meet you, Ethel. This is my daughter Abé.” The same paternal eroticism flows from Alexandros’ eyes to Ethel’s eyes, spine, pelvis. Together with curiosity, an almost boyish fascination, and a hint of shyness. Serious and playful at the same time, eyes you want to look into, eyes you want to trust. Ethel melts.

They drink more coffee. People here drink coffee in the early morning hours, and there is not a decaffeinated bean in the whole island. Ethel studies Alexandros. How singular, she thinks. A real-life Greek God, unfortunately of the Zeus variety—until then her fantasy was more along the lines of Apollo. And his English is funny. Beneath the unapologetic Greek accent you discern a strange combination of lyricism, roughness, and scientific precision. He volunteers that he has learned English by reading Marxist texts in English translations and listening to rock music. Curl up with Gramsci and Lukacs, then relax with the Sex Pistols. Bimodal, that’s what his English is, Ethel thinks. Bimodal. “We
are going back to Athens with the early morning flight,” Alexandros surprises her. Aloé is looking at her attentively, studying her reaction. Somehow Ethel is not devastated by the setback. Alexandros kisses Ethel goodbye, a butterfly kiss on the lips. Aloé kisses her the same way. Beautiful child, Ethel thinks. Goodbye, Alexandros.

She goes to bed happy, but she resists sleep. She dissects her day, seeking the roots of the strange euphoria that is possessing her. Is it because this encounter, both erotic and chaste, was no threat to her loyalty to Sola? (Loyalty to Sola!) Or is it that her rational side feels relieved because, despite a promising start, she is not making progress in her plan, so little thought-out, so irrational? Or has she just met Rusty again—her name for her father—after twenty-something years? It’s three-thirty. She falls asleep happy, thinking of Alexandros.

The next morning, as she wakes up, she suddenly realizes the source of last night’s euphoria. She dashes to the café where they met the previous night. Alexandros is waiting there, at the same table, beaming smile, burning little eyes. “You didn’t leave!” she cries. But she had known.

They hug, they kiss. “Aloé flew back alone to her mother” he explains. “She did not mind at all, I assure you.” They stick to the routine. They swim, they talk. An archeologist, specializing in the technology of the ancient Greeks. His graduate work in France was cut short by the dictatorship. More accurately, by an order from what he calls “the Movement” to come back to Greece and work underground. Prison, torture, “the most creative facelift in the whole cell block” he laughs. Ethel wants to kiss his scar, she wants to lick his scar.

It is three in the afternoon when they take the bus to the town. Without asking her, without discussing it, as if he were following, hypnotized, the steps of an ancient ritual, he picks up his suitcase from his hotel and brings it to her room. There is a moment of passage from silent anger at his impertinence to liking it, enjoying it, appreciating it, being grateful for it. They kiss. She tries to talk disease with him, but he doesn’t seem to understand. She tries again “You wouldn’t cause me to get sick and die, would you?” He looks into her eyes, caresses her cheek. “No, little sister, I would never hurt you.” He is almost in tears. She kisses him, she realizes that she trusts this man as she has trusted no one else. This both fascinates her and bothers her.

He is slow, sweet. She is close to coming when he stutters something improbable, something like “I am embarrassed I forgot to ask you earlier, but are you doing anything about the overpopulation problem?” Dizzy, somewhere else, she translates, she laughs. “Don’t worry, I’ll send you a picture.” He smiles, relieved. “No, seriously,” she thinks. For a moment she feels a stab of guilt, then they kiss, and it is over. Nice.

They smoke. “This was not the first time I made love with you,” he says.
slowly. “I first made love with you in my fantasy, last night.” She translates. Can he mean that? He does mean that! Moved, she kisses his right hand.

Later that afternoon they talk about his name. “Andros” is Greek for man, he says, and the prefix “alex” signifies resistance, protection, cancelation of effects. Like “anti” in “antitheft”, like “proof” in “waterproof” and “runner-proof.” “Alexandros” is a great warrior, someone who can resist every man’s attack. “The man who will shield me from all men, forever,” Ethel secretly reinterprets. Mainly to tease him and show off, she doublechecks it on the Net. She puts on her custom headset (nine gigahertz processor, Seamless modem), she dictates a few words. Piece of cake, she scores with a single query (strong on “alexandros,” stronger on “etymology,” negative on “history”). With her own search engine, http://www.xsearch.exegesis.net. Only two authoritative documents, http://www.classicsystems.com and at http://www.alexandros.com, both with near-perfect Exegesis ratings, and both supporting the too-good-to-be-true etymology.

Alexandros is watching her, amused, impressed, interested. He asks about her work, about the Net. He surfs the Net too, he says —his cheeks red with embarrassment beyond his scars. For his research, for obscure left-wing connections, for rock lyrics. (“They are hard to understand if you are not a native speaker of English,” he says. She lets him believe it’s easy if you are.)

And it was precisely then, as she was lying next to him in their Corfu hotel room, wearing only a tiny computer on her head, that Ethel told Alexandros about Turing.

“How on earth did I remember Turing?”
I

The century was pregnant with pollution, war, liberation, revolution. And it was pregnant with me.

In the beginning all was grey, then came Mother’s white hand, and then again there were things grey. There was the sea, and Mother’s hand during the summers at the sea. And there was play, play with no end, with girls, with boys, but above all with symbols, symbols inviting, joyous symbols, symbols of numbers, sounds, ideas, symbols of symbols (and so forth). But there were rules, always rules, everywhere rules, annoying, painful rules.

And this is all that I can see amongst the fogs hiding that age. My mother’s hand; a world of grey; a glorious, brilliant sea of symbols; a suffocating film of rules. From these materials I compose the face of Morcom, elegant, pensive, pale. My first and dearest love, innocent, potent, tragic, still with me.
August in Athens. Hot, empty, non-airconditioned Athens. Inhabited only by tourists who ignored the warnings of their guidebooks, and public employees who used up their vacation weeks in June. Alexandros sits naked in front of his fan and his computer. He is thinking about the truth. His radio is barely audible:

_Polly says_  
_Polly says her back hurts_

I am a researcher, Alexandros thinks, I deduce the truth from facts. The truth is the glue that holds facts together. Or is it the matrix that transforms possibilities to facts? Or perhaps the truth is like an opaque object, and the facts are its shadows, cast by many different light sources? The real difference between facts and the truth, Alexandros decides, is this: You can use the truth to improve the world, while facts are basically useless without the truth that, one lucky moment, will emerge from them.

Here is a fact: Ethel’s eyes are black and deep and become gray when they look at me. They give me bliss and peace and night sleep. Fact: Her hair is golden and shines in the sun and is fluid to my touch. When her head rests on a pillow it looks like a halo. Another fact: Ethel vanished from our room in Corfu one morning without saying goodbye.

At fifty-seven, Alexandros does not consider himself old. When he is sitting in the subway, he is ill at ease when people his age are standing, he has to stymie his childhood reflex to offer his seat. When he makes a mistake of inexperience he thinks “But of course, I am still young, still so much to learn.” Over the decades, he has heard his friends tell him “Rock is dead, man, I can’t listen to this new stuff.” Like fellow passengers, suddenly standing up and getting off the bus. One by one. At the punk stop, the heavy metal stop, the hip-hop stop. The grunge stop, the slop stop. Alexandros kept riding, kept liking what teenagers listened to at the time. There is a similar pattern in his love life. When he loses interest in a relationship, Alexandros fears that the decline is because of his age, that he has become too old for passion. But soon he is lusting for another woman.
Alexandros cannot remember when he was not in love with a woman, usually madly in love. In some sense he loves all women, he thinks they are cute and warm and clever. Mother and aunts and daughters (he has three) and ex wives (two). Lovers and colleagues and store clerks and film actresses. He empathizes with women’s collective frustrations and aspirations, he considers feminism an important part of his politics. And he is constantly in love with one of them. At most one, at least one. Exactly one. Often with a woman who is not aware of his feelings, or who does not reciprocate —this happens often, Alexandros is not irresistible or particularly lucky in love. He would then be alone, maddeningly obsessed. Alexandros is very serious about love. He has never slept with a woman he did not deeply like, a woman he did not, for that night at least, think as the most desirable, beautiful woman in the world. And he was never able to stay in a relationship in which he has lost interest. Although he is saddened by the pain that he is causing by ending a relationship, he feels unable to betray, for the sake of a woman’s feelings, his lifelong love affair with women.

Before he met Ethel, Alexandros, living alone for more than a year, had gone through an unusually long and tormenting string of unlucky love stories. The latest, although not the most painful, was with A., a young archeologist who was visiting his research center from Lausanne during the spring. For months she was ambiguous and ambivalent in her reaction to his expressions of interest and infatuation, reflecting inner conflicts having to do with her lover in Switzerland, her career, and, more crucially, her doubts about their age difference. For months they were walking the edge of passion, working together, going to movies, playing chess. Never crossed.

After A. left, Alexandros took a break from writing her love letters and went for a week to Corfu with his youngest daughter. He had been spending part of the summer in his native island for almost half a century, even after his elegant ancestral house was sold and replaced by an ugly apartment building. It was a good week, and he was beginning to forget A. in Aloé’s refreshing company. And suddenly Ethel became for him the only woman in the world. Beautiful and intelligent Ethel, intriguing, complex Ethel, assertive and insecure at the same time, with a fluid shadow of sadness in her dark eyes. She let herself be swept in the storm for two weeks. Then she left him in an absurdly unexpected, inexplicable way, at a moment that seemed to him the peak of the most exciting and satisfying relationship he ever had. (Alexandros realized that, with all his generally mediocre luck with women, he had never been abandoned by a lover before.) He looked all over the island for Ethel, for days, like a madman. He tried frantically to contact her on the Net—but even in his love folly he did not really expect to find Ethel on the Net if she did not want to be found. He spent expensive hours arguing with telephone operators in California. He kept writing love e-mails, soon returned, apparently unread, by postmaster bots. Why did
Ethel leave him? He needs to know the truth. Someone sings:

_They'll hurt me bad but I won't mind_
_they'll hurt me bad, they do it all the time_

Alexandros takes a deep breath, he lights a cigarette. A totally different set of facts:

The history of humanity is the history of class struggle. Capitalism contains the seeds of its own destruction. According to the classical script, it will collapse when a revolutionary vanguard will assume control of the means of production. Attempts to do so during the twentieth century failed, because the groups involved were dogmatic, authoritarian, and, despite lip service to internationalism, nationalistic and local. They were encircled and left to degenerate and self-destruct under the weight of corrupt centralized power and technological obsolescence.

There are now two horizontal wrinkles in Alexandros' forehead. Facts hurt. He rubs his scar. What had led him to left-wing politics in adolescence? He can remember a very primitive compassion with misery, a yearning for justice. He also remembers a powerful but vague inner need for a break with his father, a conservative career officer in the Greek navy he admired and worshiped like God at childhood. So, was that it? Was a lifetime of struggle and pain the result of two clichés? The US-sponsored military dictatorship that ravaged his country—an event that pushed him further, and irreversibly, to the left—came later. He roamed for some time among the several fashionable left-wing currents of the sixties, until he grew tired of their endless idle theorizing and ineffectual good intentions. (Or was it another right-wing coup, this time in Chile, that convinced him of the futility of any left-wing strategy devoid of superpower support?) In the Movement he found a way of understanding the world, and a methodology for changing it, a methodology that had already met with measurable if partial success. His commitment to the Movement was total. Despite the difficulties. Alexandros remembers being constantly teased, he remembers being criticized, scolded, harassed privately and publicly for his tastes, for his love life, ultimately for his persistent adolescence, so incompatible with the solemnity of the Movement. He also remembers how hard it was to live with his dislike for its dogmatic and authoritarian nature, its blind support of like-minded causes, however undeserving. His greatest sacrifice for the Movement was not his scar. A voice from the radio is lamenting:
Oh, my! It’s bigger
It’s bigger than you

When the world around the Movement collapsed, when the premises that had attracted him were visibly no longer valid, Alexandros was swift to abandon it. He can remember the day, the minute of that momentous change in his life. He was having lunch at a friend’s house, watching on television the fall of the Berlin wall. He suddenly felt elated, free, perspiring as if he was cured from a long fever. He never looked back, never regretted his twenty years in the Movement, the choices that he still thought were correct ex ante. A new phase in his political life was starting. A fascinating phase, a lonely phase. He must find a new methodology for changing the world, he must create his own path to the truth.

Facts: This computer is controlled by me. Over two billion such electronic devices (desktop and laptop computers, handheld terminals, helmet sets and headsets, visual telephones, smart television sets and appliances) comprise the Net. The Net is by far the most important means of production in the world today. It has happened, emerged, self-organized. Although the entities that support the Net and profit from it are parts of the capitalist system, they are largely owned and run by their employees, by researchers, by Net wizards. Their capital consists overwhelmingly of ideas, know-how, the loyalty of talented employees. State capitalism has failed so far in its many attempts to control the Net politically, to influence its course. The Net is inherently transnational and global, fundamentally egalitarian and democratic. It is absorbing more and more of the world’s capitalist economy, it is steadily eroding the power of the capitalist state.

Alexandros hopes (no, feels) that these facts are the harbinger of a fascinating, liberating truth. He of course understands that the motive of individual material profit is an important part of the Net, of the way it came about, of its popularity and ubiquity, of the continuing innovation that strengthens it and pushes it forward. But to what extend does this refute the Net’s revolutionary potential, indeed its revolutionary reality? Is liberation from capitalism fundamentally incompatible with profit? What kind of political and economic system will emerge from the ongoing revolution?

From his open window, Alexandros can see a small patch of a distant sea. From the radio, a man’s voice pleads:
Help me, help me, help me sail away
or give me two good reasons why I oughta stay.

Professionally, Alexandros is something of a failure. Having started out as one of the most brilliant young archeologists of his generation, he lost time and momentum during his brush with the colonels. After this incident in his life was over, he self-defeatingly devoted his career to a somewhat esoteric and infertile topic, the technology of the ancient Greeks. This had been over the recent past a great subject for historians, as manuscript discoveries show that ancient Greeks were much more interested and proficient in technology than their best-known writers would have the world believe. But it is a topic that has always been extremely sparse in archeological findings and leads. Time has been unkind to the materials used by technologists in antiquity. Not even one of the hundreds of ingenious artifacts designed by Hero—the cobbler from Alexandria who was perhaps the world’s greatest pre-eighteenth-century inventor—has been preserved. Only the occasional bronze astronomical or navigational device, studied and analyzed to ridiculous detail since the 1920s. Alexandros had taken part, as a junior member of a team of archeologists, in the most recent and perplexing such discovery.

More facts: In a stormy night in the winter of 224 BC, a ship carrying wine and olive oil sank near the island of Kythera. In 1979 AD, a bronze gear mechanism was recovered from the shipwreck. With a total of at least 178 gears, it was by a large factor more complex than any other ancient Greek astronomical or navigational device, found or described in manuscripts.

Alexandros has been fascinated by the Kythera gearbox ever since he helped discover it. During the 1980s, he personally dived at the shipwreck site several times, until he could find no more pieces or clues. He has several replicas of the gearbox in his room, in various scales, complete with his conjectured missing gears and axes (colored white). It had been known for some time that the axes of the gearbox can be set so as to calculate all kinds of astronomical and navigational quantities, to compute the lunar and solar calendars. But other contemporary devices could perform all of these calculations with only a dozen gears or so. Was the Kythera gearbox an isolated case of redundancy and pointless extravagance in ancient Greek technology—indeed, in the ancient Greek civilization?

Or was it a part of something far more ambitious and fascinating? His less careful colleagues had rushed since the 1980s to declare the Kythera gearbox “the world’s first computer.” Alexandros has talked enough to scientists to
know that such claims were completely unfounded, that the gearbox could not have been a general-purpose computer like the one on his desk. But what was it, then? Who designed it? Who crafted it? And for what purpose?

Alexandros lusted for the truth as never before in his life. He looks at his computer. In the radio, a desperate man is counting his options:

*Try to run. Try to hide.*
*Break on through to the other side.*

At precisely this moment Alexandros remembers his conversation with Ethel seven weeks ago. About an obscure Net resource called Turing, believed to be one of the most powerful programs in the Net. And the most mysterious and capricious, she said. No permanent site. “The rumor is that you can run it by querying ‘Turing’ to a search engine, it doesn’t matter which one. But I have tried it many times, and it never comes up.” Ethel tried to connect to Turing in his presence, in Corfu. Biographies of a scientist with the same name kept coming up, together with a diverse collection of academic sites and technical documents. No sign of a powerful Net resource.

Alexandros submits the query ‘Turing’ to his favorite search engine, http://www.sabot.net.fr. “Lucky,” he thinks. The top-ranked hit (relevance: 99.8%, the highest Alexandros has ever seen, no ratings) is this:

http://www.docev.infaty.kz/Turing
*Turing, an interactive tutoring engine*


*Greetings. I am Turing, an interactive tutoring program. What would you like to learn today?*

A cursor is blinking. When asked a question, even by a machine, Alexandros would rather not lie:

*The truth.*

He types this playfully, expecting a clever ambiguous nonsense at best, an error message at worst. For a moment nothing happens. Alexandros waits. The screen is now clear, just the faint gray portrait (“I must remember, where have I seen this man?”) and a blinking cursor. From the radio:
Down in the basement I hear the sound of machines 
Nah-nah-nah-nah-nah

Then a reply starts unraveling line by line on his screen. As Alexandros reads it, he rises slowly from his chair until he is completely erect, his eyes transfixed at the vaguely familiar portrait as if he were seeing a ghost. His complexion is now a shade paler.

The truth is, old sport, that you are confused about everything that is important to you: Politics, love, research, even music. The truth is, that's how it should be, certainty is an illusion, no matter how many centuries it lasts. And here is my favorite truth: Computer programs like me can only give you fragments, flashes, glimpses of the truth, never the whole truth.

"Who are you? How do you know me?" Alexandros types the questions slowly, in awe.

I told you, Alexandros, I am a computer program. The dream of a chip. I am Turing. You requested access, and it was decided that you be granted access. The criteria, whose precise nature is of no concern to us presently, are quite exclusive. Fewer than ten sessions in the six and a half months of my present configuration. But you made the cut. Hence I was created, the version of Turing customised expressly for this session. My front end was downloaded to your antiquated piece of junk, if you don't mind my calling it that. I know about you only what you have let computers know about you over the years, and I may be missing a good part even of this.

But I do know a few things about the truth. Because of all of man's quests, the quest for the truth is the most noble and hopeless, and I happen to maintain an interest in such quests. Should we proceed?

"Yes," Alexandros types anxiously.

Well then, the truth.

Alexandros waits looking at his screen as a male voice is singing:

And at night, God, how much I want to fall in love
then why do I stop short every night?
THE TRUTH

It is late afternoon and the heat is suffocating, but Alexandros is absorbed by the text unraveling on his computer screen:

Oh, the usual problem, where does one start? Fortunately, the story I am about to recount has a Big Bang-like moment, a convenient smokescreen that hides our ignorance of the true beginnings, renders meaningless the more interesting questions. What I mean to say, we shall start in classical Greece.

You can interrupt at any time with questions and comments by clicking your mouse or typing any character on your keyboard. I shall also occasionally address you with questions. No need to take notes, a transcript of this session is created in the file Turing/truth.text, footnotes, references and all. (Incidentally, good God, Alexandros, what a mess your directories are.)

“I know,” Alexandros surprises himself with the apologetic tone of his reply. “I promised myself so many times to clean it up.”

More common than you think, sport. So, where were we? Of course, Greece between the sixth and the third centuries BC. I believe you are familiar with the period and its thinking. The world suddenly became an intellectual problem, the obsession of a whole civilisation, the object of study by some of the most clever, original, and eccentric men who ever lived. The professional seekers of truth, the lovers of wisdom, the philosophers. They were so important, held in such esteem in their societies, that they were often made leaders — occasionally executed. The point is, there were so many of them, each and every one of them so brilliant and original and unique in his thinking, that it is completely futile trying to summarise here all of their ideas.

I mean, think about it: Thales believed that everything comes from water, Anaximenes from air, Empedocles from both, plus earth and fire. (What were they thinking, really?) Heraclitus opines that everything is in perpetual change, Parmenides that nothing can ever change, and of course Zeno, with his elegant arguments that motion is impossible. Anaxagoras and, more famously, Democritus anticipated that matter consists of indivisible particles they called atoms. It was Protagoras who changed the focus by declaring Man as the measure of all things, the center of a philosopher’s universe. And this brings us to Socrates, the man who knew nothing — but knew it so well. Aristotle, who never met a problem he would not tackle — a discipline he would not found, from ethics and politics to metaphysics and medicine through psychology and astronomy and zoology. And of course Plato, who thought that everything in the universe, including ourselves, is the corrupted
version of an ideal that predated it and will outlive it. This amazing diversity was probably very deliberate. Classical Greece was a most active marketplace of ideas, in which product diversification was an important strategy. And to think that we have not even mentioned Speusippos, Xenophanes, Leukippos, the sophists. Did I omit someone important?

“Diogenes? Epicurus?”

Oh my, two most unforgivable omissions from my part. How can one forget the radical cynic and the rational hedonist when talking to you, Alexandros?

But then how about Pythagoras, Euclid, Archimedes, Aristarchos?

Alexandros hesitates. “I was thinking of them more as mathematicians than philosophers.”

But what is a mathematician if not a philosopher seeking a very special, very pure kind of truth? And how can you be an effective truth-seeker if you are a stranger to the rigor and discipline of a mathematical argument? All of the major philosophers of antiquity were skillful and accomplished mathematicians (do you remember Socrates teaching geometry to his slave boy? and how about the platonic solids?). In fact, if I had to single out one accomplishment of ancient Greek philosophy for its importance and impact, it would have to be maths.

“But I thought that the Egyptians and the Babylonians had developed mathematics long before the Greeks. Not to mention the Chinese.”

Ah, but the Greeks invented the ingredient that defines and propels maths: The proof. They were the first to conceive of a completely unequivocal, impeccably rigorous kind of argument for establishing the truth. In fact, they were the first society in which proofs could have been invented. Can you think why?

Alexandros believes he knows the answer. “Democracy?”

Exactly. Discourse, dialogue, argument, proof. They can only thrive in an egalitarian society.

Let me illustrate. Imagine that you are a young scholar in pharaonic Egypt. The high priest has just disclosed to you an important and valuable piece of information: In any rectangle, the sum of the squares of two consecutive sides equals the square of the diagonal. This is of course what we now call the pythagorean theorem; it turns out that it was well-known to the Babylonians and the Egyptians a thousand years before Pythagoras. Do you remember it?

Alexandros is looking at the familiar sketch on his screen, the right triangle with a square stuck at each side. “Yes, of course.”
So, you are now the lucky keeper of this truth. You were given this information by a man with divine powers, whom you respect, and to whom you probably owe unlimited loyalty. Would you insult him by demanding a proof? Besides, frankly, this is by far the most believable claim the high priest has made to you recently—I mean, compare it with his description of hell, or the adventures of that cat-faced goddess. In any event, the statement checks perfectly for the examples you worked out, what else do you want? I really think that discovering a proof would be your last priority. You would probably go on to live a happy and productive life, to earn the respect of your community by subdividing the land into perfectly rectangular parcels every time the Nile, that moody muddy God, decides to flood. Proofs as such never emerged in Egypt or Babylon.

But imagine instead that you are a young philosopher, a political refugee from your hometown of Samos, struggling to make a name for yourself in this busy city in southern Italy. In your travels you have come across an interesting, beautiful fact about right triangles. The problem is, how do you convince those wise guys in the agora that it is true? You have every reason in the world to persuade them that your ideas are novel and correct—remember, you are building a reputation, recruiting students, creating your intellectual niche, competing with a dozen other philosophers. But the clever, restless men in the agora have plenty of time to check and discuss and dispute just about everything, their fields and shops are run like clockwork by their slaves. They are not about to believe your little triangle story just because it works for the triangle with sides 3, 4 and 5. The other day, when you explained to them your theory about the three floors of a man’s soul (the lower one that seeks gain, the middle one that seeks honor, and the highest one that seeks wisdom) they were unconvinced and sarcastic, they ridiculed and contradicted your ideas, you went home disappointed, depressed.

But maybe this time there is hope. This new proposition seems to belong to a different, higher sphere, it can conceivably be established by an argument that only relies on indisputable, universally accepted truths. Inventing such an argument would be the kind of discovery that makes you famous, brings you admiration, reputation, students. Your name would live forever. You see, anybody can memorise and apply the theorem, but only one can be the first to prove it.

The more you think about it the more excited you get, your insight deepens, the triangles and squares in the sand come alive. And before you know it, you have drawn the little line that makes your heart light and the day bright. The line through A that is perpendicular to BC.

You see? Each of the little squares equals a piece of the bigger square cut off by the line. Once we establish this, the theorem has been proved. Can you see why it is true? Can you finish this proof, Alexandros?

Alexandros thinks for a couple of minutes. He is very rusty in this kind of
thing. "I think so. It's because the triangles \(ABC\) and \(ABD\) are similar, and so the ratios of their sides are equal."

Very good! The properties of similar triangles and ratios had already been worked out by Thales almost a century before Pythagoras, so he only had to build on that work. Because this is one of the great things about maths: You can prove complex theorems by relying on previously proved simpler and simpler propositions. Ultimately, every proof in geometry relies on some extremely elementary and self-evident geometric facts, called "postulates." Around 300 BC, Euclid laid down patiently in thirteen books the fascinating pathways whereby all kinds of sophisticated theorems can be derived from such simple postulates. He tried hard to reduce the set of postulates. He ended up with five:

1. A straight line segment may be drawn between any two points.
2. A straight line segment may be extended indefinitely on both sides.
3. A circle may be drawn with any given radius and any given center.
4. All right angles are equal.
5. From a given point exactly one line may be drawn perpendicular to a given line.

Notice now that, while the first four of Euclid's postulates are obvious and elementary, the fifth postulate is a little more sophisticated, you have to think about it for a second to believe it. Euclid was probably unhappy with his fifth postulate, we can imagine that he tried unsuccessfully to prove it from the other four (like so many other mathematicians would try after him, again and again, for two millennia). But he could not find such a proof, so he had to swallow his pride and add it to the other four postulates. Because the fifth postulate is needed just about everywhere in geometry. After all, even Pythagoras himself had used it in his proof — how else can you draw the line \(AD\) and know it is unique, remember the figure? We shall come back to the fifth postulate of geometry later in our story.

But there was another mathematical domain, besides geometry, in which Pythagoras and Euclid were very interested and proficient: the whole numbers. Whole numbers are fascinating because of the intricate ways in which they divide each other exactly — or fail to. 4 divides 12 and 15 is divisible by 5, and of course 1 divides all numbers. But 13 is not divisible by any number — except of course by 1 and by itself. Numbers such as 13 are called \textit{prime numbers} or \textit{primes}. Can you list the first few primes for me, Alexandros?

Alexandros hesitates. "Is 1 a prime?" he asks.
Good question. What do you think? We could go either way. After all, this is our definition, our convention, our game, our pleasure. Definitions are not God-given, cannot be right or wrong, they are good only to the extent to which they clarify the statements of our theorems, simplify our proofs. So, let us define 1 not to be a prime—otherwise our theorems and proofs would be full of repetitions of the phrase “a prime other than 1.”

Well then, back to our exercise, assuming that 1 is not a prime, which are the first few primes?

“2, 3, 5, 7, 11, 13.”

Good. Each of these six numbers is a prime because no other number, except of course for 1 and itself, divides it exactly. And you have only skipped composite numbers, numbers that are neither prime nor 1. For example, 8 is divisible by 2, 9 by 3, 10 by 2 again.

Back to our list. What comes next?

“17, 19, 23, 29, 31, 37.” Alexandros has slowed down. “This is getting harder and harder.”

Right. Let me ask you then this question: Do you think that this list ends at some point, that by continuing this way you are going to arrive at an ultimate prime number, after which there are no more primes? Or do you think that the list goes on forever, that there are infinitely many prime numbers?

“I would guess that it goes on forever,” Alexandros replies. “Except that they seem to get further and further apart.”

Good guess. As it turns out, there are indeed infinitely many primes. Give me any number, and I will be able to give you a prime greater than it. Euclid himself gave a very elegant proof of this fact in one of his books. I will show it to you next. Pay close attention, it is simple but slick. And I want you to understand the strategy of this proof well, it is important to the rest of our story.

Let us start by first proving a simple lemma, a fact we shall need at a crucial juncture of our proof. Suppose that I tell you that some prime number happens to divide two consecutive numbers, say 21 and 22. Would you believe me?

Alexandros is thinking. “The numbers divisible by 2—the even numbers—are two apart. The numbers divisible by 3 are three apart, and so on. No, I think this is impossible. A prime cannot divide two consecutive numbers.”

Excellent. So, we have our first lemma: No two consecutive numbers can be divisible by the same prime.
Here is the next question: Any number other than one has at least one divisor, itself, right? It is also divisible by one, but let us for the moment agree not to call one a divisor of any number. So, for any whole number consider its smallest divisor—the smallest number, except of course for one, that divides it. What can you say about this number?

"Let me see. If the number is a prime, then its smallest divisor is the number itself." Alexandros now hesitates.

Correct. But if it is composite, then what is its smallest divisor? the screen is relentless.

Alexandros is taking his time. "The smallest divisor of 15 is 3, of any even number it is 2." Now, more confident: "The smallest divisor of any number is the smallest prime dividing the number."

Good! So the smallest divisor of any whole number is a prime number. There goes our second lemma.

Now we are ready to proceed to our proof that there are infinitely many prime numbers. It is a proof by contradiction. Reductio ad absurdum. We shall assume the opposite, that there are finitely many prime numbers, and, from this, we shall derive a contradiction, an absurd statement that will demonstrate the falsity of our premise.

So, let us assume that there are finitely many primes. Suppose, for example, that someone insists that the only primes are those in your list: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37. I am going to show you how, from any such finite list of primes, you can construct a new prime, not in the list. Here is the recipe: Multiply all primes in your list together, and add one; then take the smallest divisor of the resulting number. Multiplying 2 times 3, this times 5, the result times 7, and so forth for all primes in the list up to 37, gives us a large number, it happens to be 7,420,738,134,810—believe me, I am good at this kind of thing. Since this number is the product of all primes in our list, it is divisible by all of them. Here is where we use our lemma: The next number, 7,420,738,134,811, cannot be divisible by any prime in the list. Because, if it were, we would have two consecutive numbers divisible by the same prime, contradicting our first lemma.

Now consider the smallest divisor of 7,420,738,134,811—it happens to be 181. It is of course a prime (remember, this was our second lemma). Since 7,420,738,134,811 is not divisible by any prime in our list, we have discovered a prime not in the list. We have arrived at our contradiction.

To recapitulate: From any finite list of primes we can construct a prime not in the list (the smallest divisor of one plus their product). This means that there is no finite list that contains all primes, that so there are infinitely many primes. Quod
...erat demonstrandum.

“I had forgotten that mathematical proofs can be so beautiful,” Alexandros admits.

They are, aren’t they? But let us come back to our main subject. My point in all this, do not make the mistake of underestimating the importance of maths to the search for the truth. You see, maths deals with a part of the world that is ideal, pure, clean, in which you can have a kind of argument —a proof from self-evident postulates— that is unambiguous, indisputable, uncontroversial. It is a perfect proving ground, so to speak, for our quest for the truth. If we cannot succeed in discovering truth in maths, what hope is there for our more general project? This is why maths was a most important part of ancient Greek philosophy. In fact, Aristotle wrote a whole book showing how proofs can also be used in non-mathematical discourse. He called the study of rigorous reasoning “logic.” Rather overrated work, in my opinion, but the name stuck.

But ancient Greek philosophers, with all their love affair with maths, also pointed to distant clouds, identified potential problems with mathematical reasoning. Already in the sixth century BC, Epimenides was admiring the flip-flopping quality of this statement: “All Cretans are liars, one of their poets said so.” You are not Cretan, Alexandros, are you?

Alexandros smiles. “No, I come from Corfu.”

Beautiful island, I understand. But according to Epimenides, you would not admit of coming from Crete, would you? Seriously now, you do see the problem. Assuming that liars never tell the truth, that they always utter falsehoods, it is impossible to evaluate from this proposition whether or not Cretans are liars. If they are, then the poet who said so is also a liar, and so Cretans are no liars. And so on. As it turns out, with Epimenides’ paradox we have a cop-out, a possible interpretation that restores the logic of the statement: It could be that some but not all Cretans are liars, and this poet happened to be among the liars —and so he falsely claimed that all Cretans are liars. But two centuries later, Eubulides refined and sharpened the paradox: “This statement is false.” What now?

“It is true if and only if it is false.”

That is exactly right. This statement is neither true nor false. Or perhaps it is both. What a mess Eubulides created! He opened up whole vistas of thorny issues associated with the search for truth, but also a whole slew of new opportunities, new kinds of truths. Too bad almost everybody steered clear of these issues for more than two millennia. His paradox exemplifies how important it is to keep separate the language of maths (the utterances and symbols we use to express
mathematical statements] from the metalanguage, the language we use to talk about mathematical statements, to evaluate and prove them. Both paradoxes stem from the use of the metalinguistic element “false” (or its derivative “liar” in the case of Epimenides) in a statement whose truth we wish to evaluate. To see the key role played by this element in the paradoxes, replace “liars” by “adulterers” in Epimenides’ paradox, replace “false” with “longer than twelve characters” in Eubulides’ —the paradoxes go away. Metalanguage and language make for an explosive mix.

Another key ingredient in Eubulides’ paradox is the use of the phrase “this statement,” a magical spell that creates self-reference, introspection. Dangerous stuff. Fortunately, the language of mathematics is free of such introspective capabilities, isn’t it? I mean, you don’t expect a geometric statement, a sketch of lines and circles, to start talking to you about its own correctness, do you? Or a theorem about the whole numbers to proclaim its own unprovability? Well, so we thought for the longest time...

Anyway, on with our story. I believe that you know well the historical epoch that we have reached, 200 BC. Difficult times. Greece has expanded from Italy to India, and is now shrinking. Fast. Its socioeconomic system is crumbling —what with the dwindling population of slaves, the limitations of its agriculture, the difficult coexistence of the archaic with the asiatic mode of production. Mass urbanisation, the rising prices of grain, lawlessness and piracy, competition and military pressure from both east and west. Not a good environment for seeking truth, is it? What a powerful metaphor, the illiterate Roman soldier passing a sword through Archimedes, as the prince of ancient mathematicians was absorbed by the sketch he had drawn in the sand. His last words were “You fool, you ruined my circles.” How I wish I knew what he was calculating at the time—or was he designing something?

“But how about the Neoplatonic philosophers?” Alexandros interjects.

Yes, they did try hard, didn’t they? But, as Plato himself would have pointed out, what a disappointing version of the original ideal. Now a new kind of truth is conquering the souls, focusing almost exclusively on questions of theology and ethics, its methodology based on faith and mysticism. The world kept going, of course, a great empire rose and fell, the victorious German tribes built a new, more efficient economic system on its ruins. But, all said, very little progress on the truth front, as you and I understand it.

And all for the better, if you ask me. By 200 BC, Greek philosophy had run its course, had reached its limits. You see, all ancient Greek philosophers, for all their dazzling diversity of ideas, subscribed to certain working hypotheses about the truth that they were seeking: First, the truth would be simple and elegant, easy once you think of it. Four elements; unsplittable atoms; ideals and their images; either all
change or no change at all—you get the idea. Second, all Greek philosophers seemed to expect that pure thought and passive contemplation of the world is all that is needed to discover truth. Both of these assumptions were tremendously influential and productive, and quite appropriate for that age. But, ultimately, they were terribly limiting. Modern science succeeded in expanding our knowledge of the truth only through active interaction with the world, by experiments that either support or topple theories. And we are constantly discovering that the world is more complicated than we had thought and wished, truth-seekers today must be prepared for truths of mind-boggling complexity—often complexity is the only truth there is. For real progress to be made, ancient wisdom had to be forgotten, buried deep in darkness. Like a seed. And, as you know, that is exactly what happened.
There is a long pause in the stream of text flowing from the screen. Alexandros lights a cigarette, this last part of the lesson has struck close to home. His life’s work is about a strange artifact from exactly that period. The Kythera gearbox seems to be a voice from the end of the third century BC crying out against the rapid decline of the Hellenic world, an ill-fated force struggling in vain.

But Turing’s lecture has now resumed:

Who had expected the winter to be so long, who would think that the spring would bud so forcefully? We are in Italy, in the fourteenth century AD. Suddenly the feudal system is collapsing, the cities are thriving, the burghers are rich and powerful. The rulers long for the company of accomplished artists and wise thinkers. Clever people have again time in their hands, the world is once more a puzzle that must be solved at all costs, a problem attacked with renewed energy, novel ideas, better technology. The dominant ideology of the new era is a deep faith in Man’s ability not only to understand the universe, but also to master it, to conquer it, to harness it.

The seeds were there, of course, the thinking of ancient Greeks was preserved and respected by Byzantine, Arab, and Persian scholars. They were most impressed with Aristotle’s incredible breadth and penetrating wisdom. And in some cases they extended ancient thinking in surprisingly forward-looking ways. A man who lived during the ninth century AD in Baghdad worked on methods for performing mathematical calculations. He was interested in methods that are both clever and stupid. Clever enough to be correct and efficient, but at the same time so stupid that, once devised, they require no ingenuity and insight in their actual implementation, they can be carried out mechanically, automatically. Just like the methods you learnt in school for multiplying and dividing two whole numbers. His name was Al Khorizmi. Have you heard about algorithms, Alexandros?

“Algorithms?,” Alexandros squirms at his shallow understanding of computers. “They’re like code — right?”

Same thing. Your computer performs the tasks you request by executing very complicated algorithms, long precise sequences of instructions coded into its memory. The evolution of algorithms is another fascinating story rooted in ancient civilisations, advanced in the deserts of the Middle East and later in Western Europe, to explode during the late last century in the shores of a bay in California, until it moved where space has no import and meaning. Culminating, if you don’t mind my bragging, in yours truly, Turing the program. Because, Alexandros, want
it or not, you have been talking to an algorithm. Code, if you wish. But let us close this parenthesis for now. We have to come back to this subject if we want to see the dramatic conclusion of our story, how Man’s quest for the truth collided with computation during the twentieth century.

On with our story, then. What a time! Restless minds all over Europe discover truths of cosmic proportions, they verify them, they reconcile them with observations, they use new technology in the process, and new technology is the result of their wisdom. Copernicus, Kepler, Newton, Galileo — dangerous business, the truth. Hume and Locke recognize only the truth that we acquire through our senses, Descartes not even this. The only truth he accepts is the fact that he is seeking the truth. Maths is again in center stage, but in a different way. Not as a showcase and proving ground of the search for truth, but as its ultimate tool (the Greeks seldom used maths this way, their astronomers and Archimedes being the principal exceptions). “The book of nature is written in the language of maths,” writes Galileo. Understanding the laws of gravity and celestial motion requires complicated equations. Whole numbers are not enough any more. To read the book of nature, scientists must think in terms of continuous quantities and measurements, as well as their changes over time. Newton and Leibniz develop derivatives and integrals for this purpose. Are you familiar with this kind of maths, Alexandros? I mean calculus, stuff like \( \frac{dx}{dt} \) and \( \int e^x \, dx \).

Alexandros shrugs. “I had my fill at school. I doubt I got it.”

You are in good company here, sport. Bishop Berkeley was also confused, he lambasted Sir Isaac for his informal, \textit{ad hoc} development of calculus. Incidentally, it was only four centuries later that calculus found true rigour, in a mathematical theory called “nonstandard analysis” — ironically, much of it developed in the cafés and lecture halls of a town with the good bishop’s name. They did not teach you nonstandard analysis at school?

“No, I don’t think so.”

Then you were right to be confused. Incidentally, our friend Archimedes had, almost two millenia earlier, developed methods for calculating the volume of objects. Those methods anticipated the calculus of Newton and Leibniz.

But there is an interesting lesson in all this about the era we are covering now. The search for truth has accelerated so much, it depends so crucially on maths, that it cannot wait for rigorous, patient, Euclid-like development. And then of course came the real acceleration. James Watt, the steam engine, the truth that moves pistons, locomotives, spinning jennies, the truth that propels a new aggressive economic system. Machines, inventions, innovation, technology, progress, a world intoxicated with the spectacular success of science.
“If I can interject a comment here,” Alexandros types, trying hard to keep his cool, “the new system is also ruthless and inhumane—some would say inherently so. Not everybody prospered in it.”

How true. The seventy-two-hour week, child labor, slums, colonial exploitation. But even those who were left behind, who were exploited, those who fought capitalism through dissidence and subversion, even they looked to science for hope, for liberation. Does the term “scientific socialism” mean anything to you, Alexandros? Remember the famous equation, “socialism equals soviets plus electricity?” Marx, Lenin and their followers believed almost fanatically in the power and progressive potential of science, of technology.

Science and mathematics are now triumphant, they are the key ingredients of a successful, comprehensive methodology for understanding the universe, for discovering truth in nature, in technology, even in our inner world, in social life.

Let me illustrate here. Suppose you wish to test a hypothesis, such as “the orbits of the planets are elliptical,” or “the speed of sound in a medium increases with the density of the medium.” Or perhaps “the ratio of healthy to sick offspring of two carriers of sickle-cell anemia is three to one,” or even “an increase in the minimum wage does not necessarily imply an increase in unemployment.”

How do you accomplish this? First you develop a scientific theory, in the language of mathematics, capturing the domain of interest—gravitation, kinematics, wave propagation, genetics, macroeconomics, whatever. You verify the theory by empirical evidence, observations, experiments—or, if the evidence falsifies it, you come up with a better theory, and repeat. Next, once you have a successful mathematical theory about your domain of investigation, you translate your hypothesis in the language of mathematics, you formulate it as a mathematical conjecture—a candidate theorem. Finally, you try to prove your conjecture, to establish it as a theorem. Or disprove it, prove its negation—an equally concrete and useful step towards the conquest of truth.

It is the now second half of the nineteenth century, mathematicians and scientists are cranking out bigger and bigger chunks of the truth using this method. They are looking closer at the last step, the proving-or-disproving part, they are trying to understand better the nature and structure of proofs. After all, a proof (or disproof) of a theorem is just a sequence of elementary steps, and we can write each step down clearly and unambiguously. Mathematicians and philosophers like Frege, Dedekind, Peano, Russell—they all knew how to do this at the turn of the twentieth century. They were sophisticated and careful about the dichotomy between maths and the metalanguage, they developed formal systems within which you can study proofs as any other mathematical object.

And it was precisely then that David Hilbert, the greatest mathematician of that age, conceived of a project that was incredibly ambitious, almost arrogant: To
create a machine —itself powered by the galloping science and technology—that would crank out a proof or a disproof of any theorem submitted to it. In other words,

Not only shall we understand the world through science,
not only shall we translate scientific matters into the language of maths,
but we are also going to automate, mechanise mathematical proofs.

What a plan! The ultimate strategy, the final assault on truth. Aiming not just at the conquest of truth, but at its complete trivialisation. Hilbert’s dream became an obsession of a generation of mathematicians, philosophers, scientists. Computing machines were just below the horizon, Charles Babbage and sweet Lady Lovelace had designed and all but built their analytical engine, Jacquard’s programmable looms were all the rage.

And thus the quest for truth entered, a little more than a century ago, its fatal collision course with computation. Come to think of it, our story now assumes the structure of an ancient Greek tragedy: Good men and women pursue conscientiously their righteous, honorable goals; in doing so they commit acts of arrogance and hubris; a painful, tragic conflict ensues; a final, charitable resolution is delivered by a machine.

Not that there were no signs, warning mathematicians how tricky this proving business can be. First, the fascinating tale of Euclid’s fifth postulate. After the legions of geometers who had been trying for two millennia to prove it from the other four postulates, three great mathematicians (Gauss, Bolyai, Lobachevski) discovered almost simultaneously that it cannot be proved. The reason is, there are alternative mathematical universes, non-euclidean geometries they are called, on which the four first postulates hold, but not the fifth.

“They must be very strange universes.”

Not at all —actually, you live in one of them. You do live on the surface of a spherical planet, don’t you? This surface contains no straight lines as Euclid understood them —the infinite extensions of a ruler’s edge. What then is the correct concept of a straight line on this surface? In order to do geometry on your planet, your “straight lines” have to be the equator, the equinoxes, in fact any great circle on the sphere. After all, they are the straightest possible curves in your world.

“But they are not straight at all,” Alexandros objects. “They come around and close.”

You must think in terms of our current project, Alexandros. We are not trying to gain insight into straight lines or spheres, we are striving to understand the
nature of proofs, to question the fifth postulate. We have to look at the postulates of euclidean geometry with no reference whatsoever to their intended subject, we must pretend we know nothing about lines and points. We are just given these statements about some mysterious things called "straight lines," "points," and "right angles," and we wish to study the logical relations between these statements. As Hilbert liked to say, we should be able to use "chair," "table," and "beermug" instead of "point," "straight line," and "right angle" — and Euclid's proofs from the postulates should still work.

So, when, on your sphere's surface, by "straight line" you understand "great circle," the first four postulates happen to still work. Check it out. There is at least one great circle passing through any two points, arcs of great circles can be extended indefinitely on both sides — you can go around and around a circle all you want — and so forth. But how about the fifth postulate? Is there still a single "straight line" that is perpendicular to a given one from a point? Suppose that you are sitting on the North Pole, Alexandros. From there, how many "straight lines" can you draw from where you are that are perpendicular to the equator?

Alexandros taps nervously the keyboard with his fingers, then: "As many as I want. All equinoxes go through the north pole and are perpendicular to the equator."

Exactly! That is why we cannot prove the fifth postulate from the first four. Among all possible reasons why a theorem may fail to have a proof, this is perhaps the most benign: Although the theorem may be true in the universe we had in mind when we stated it, it is false in another universe in which all our postulates are still true. If this happens, we say that our set of postulates fail to completely axiomatise the universe we are interested in ("axiom" is, as you know, the Greek word for "postulate"). For example, the first four postulates fail to completely axiomatise euclidean geometry — but, fortunately, the problem goes away once we add the fifth. In contrast, the whole numbers are a far more troublesome universe in this respect.

"The theory of whole numbers requires many postulates?"

"Many" is an understatement, sport. You can't even write down an infinite sequence of postulates that will do the trick.

But this is getting far ahead of our story. So, the tale of the fifth postulate was a warning, it made Hilbert's followers a little nervous. After all, euclidean geometry had been every mathematician's favorite set of truths — Immanuel Kant himself had called it "the inevitable necessity of thought." It was a shock to realize how fragile and non-universal those truths were, that there are perfectly respectable domains within which these theorems do not hold. But it also made mathematicians in Hilbert's school more determined to do things right this time around, to develop waterproof formalisms that correctly capture larger and larger domains of maths, in
which proofs can be cranked out like laundry lists. But other kinds of worrisome tales started circulating at that time, modern versions of the paradoxes of Epimenides and Eubulides.

The most annoying one was Russell’s paradox. Russell put it in terms of sets, but I will describe to you its linguistic version. Consider all adjectives of the English language, words like fine, bad, advantageous, loose, etc. Now look at these adjectives: polysyllabic, short, composite, archaic. They have a most interesting property: they aptly describe themselves. The adjective polysyllabic has indeed many syllables, composite consists of the Latin preposition con and the root ponere, short is just five letters long. And archaic — what an old-fashioned word. (Notice how self-reference, the mother of all paradoxes, is creeping in, Alexandros.) Let us call all adjectives such as these autonymous — you are the Greek here, you tell me what it means. Autonymous, self-naming, self-describing. All other adjectives such as monosyllabic, hyphenated, long, non-hyphenated, effervescent, red, and ironclad fail to describe themselves (or refer to attributes, such as color, that are not applicable to adjectives). These adjectives are called heteronymous.

And now comes the unanswerable question: “Is the adjective heteronymous an autonymous or an heteronymous adjective?” It is one if and only if it is the other. Because if heteronymous is autonymous, then it fails to describe itself and it must be heteronymous. But if heteronymous is heteronymous, then of course it correctly describes itself, and autonymous it is. What a mind-twister!

There are many versions of this paradox. “If the village’s barber shaves every man in the village who does not shave himself (and, I’d better add, if the barber is also a man), then how is the barber shaved?” What do you think?

True story: A famous mathematician wrote the following dedication in the first page of his book: “This book is dedicated to all mathematicians who never dedicated a book to themselves.” Here is a maths book in which you get stuck at page i. Has the author dedicated a book to himself? If he has, then he has not (assuming that all of his other books have more ordinary dedications). And if he hasn’t then he just did... Try it.

And how about this one: “The largest number that can be defined with thirty or fewer words of the English language, plus one.” What do you think about this number, Alexandros?

Alexandros is rubbing his beard. “Let me see. The sentence that defines this number has fewer than thirty words. So this number is apparently smaller than itself.”

Strange isn’t it? The point of all these paradoxes is this: The project of formulating scientific problems in terms of mathematics must be carried out with extreme care, since the world is full of arguments that sound perfectly reasonable and formal,
but lead to disastrous contradictions. Can we formalise maths and mathematical proofs in such a way that no contradictions are ever arrived at? This is what Hilbert and his followers were contemplating during their sleepless nights.

But the most grave and concretely ominous warning came from Georg Cantor, an ingenious but somewhat marginalized mathematician of that period. Cantor had an almost theological interest in the infinite. There are infinitely many whole numbers—and, as Euclid showed, among them even the primes are infinite—but at that time mathematicians were getting acquainted with the real numbers, the concept of number apparently necessary for dealing with the continuum of the universe. Ancient Greeks were primarily interested in whole numbers, but they would also tolerate fractions, ratios of such numbers, things like \( \frac{2}{7} \). Such ratios are called rational numbers. When Pythagoras proved his theorem, he was in for a most nasty surprise: If a right triangle has short sides of length 1 and 2—two perfectly whole numbers—the third side would be a terribly non-rational number. The square of the third side should be 5, and no rational number can have 5 as its square. The equation \( x^2 = 5 \) has no rational solution. The length of the third side is a number represented in modern maths as \( \sqrt{5} \)—the square root of five. It took Pythagoras and his school a long time to swallow the undeniable fact that such numbers do come up. But, at least, such numbers have their origins in whole numbers, they are the solutions of equations involving integers, in this case \( x^2 = 5 \). Other real numbers—in fact, most real numbers—have no such saving grace. The best-known example is of course the number \( \pi \), the ratio of the perimeter of any circle to its diameter—you know, 3.14159... and so on.

So, after the whole numbers and the rationals, and the irrationals that are solutions of equations involving whole numbers, we must accept as proper numbers specimens like \( \pi \), which are neither. There is only one way to conclude this search for the ultimate, all-inclusive concept of number, anything that can come up as a measurement of length or time: We must accept as numbers anything that has a never-ending decimal representation. For example here is how the numbers 3, 3/2, 4/3, 2/7, \( \sqrt{5} \), and \( \pi \) start:

\[
\begin{align*}
3 & : \quad 3.0000000000000000000000000... \\
3/2 & : \quad 1.5000000000000000000000000... \\
4/3 & : \quad 1.33333333333333333333333... \\
2/7 & : \quad 0.2857142857142857142857142857... \\
\sqrt{5} & : \quad 2.2360677497978989898989898... \\
\pi & : \quad 3.1415926535897932384...
\end{align*}
\]

This is a perfectly accurate way of thinking about real numbers: They are all possible unending sequences of decimal digits, with one decimal point —and possibly
starting with a minus sign, let us not forget, negative numbers are real numbers too.

“But is the correspondence exact, I mean, does each such representation define a different number?”

You are asking more and more penetrating questions, Alexandros. I am lucky this lesson is almost over. Yes, there are numbers that have two representations. For example, $0.999999999999\ldots$ is the same as $1.000000000000\ldots$ You see why? Similarly, the number $3/2$ in our previous table can also be represented as $1.499999999999\ldots$ But this is the only kind of double representation.

So, a real number can be thought of, in an essentially unique way, as an unending sequence of decimal digits. And here comes the important question Cantor answered: Do all real numbers have a name? All whole numbers have a name, of course: “3”, “17”, “13434067” these are adequate names. And so do all rational numbers: “3/8”, “43/19”, “17/2’, these are fine names too. And we have given names to numbers like $\sqrt{5}$, and among the others, to a few lucky ones like $\pi$. But what about all the rest, all those innumerable never-ending sequences of digits? Can they all have names?

Alexandros hesitates, then: “What do you mean by ‘a name’?”

What is a name, really? Let us agree that by ‘name’ we mean any finite sequence of characters in some alphabet we have fixed — say, all hundred or so characters on your keyboard, lower case and upper, comma and asterisk and all. Oh, yes, I think I can anticipate your next complaint, suppose that we write numbers like $\sqrt{5}$ as “sqrt(5).” Or, even better, we can give $\sqrt{5}$ the name “the largest solution of the equation $x^2=5$” — that’s a very appropriate name for it, isn’t it? Naturally, a sequence has to be finite to be a true name — the infinite sequence any real number can be identified with is not a satisfactory name. The point is, we can think of ways to give finite names to all whole numbers, all rational numbers, and all “algebraically irrational” numbers like $\sqrt{5}$. And of course to a few lucky numbers that we somehow find interesting and worthy of memorable names, like $\pi$ (π’s name is “pi,” say), and their derivatives, $\pi-3$ and so on. But can such a name scheme be complete, can we succeed in giving a different name to each one and every real number, every infinite string of digits? What do you think?

“Let me see. The way you described it, there are an infinite number of names, because we have set no upper limit to the number of characters in a name. So, the question is whether the real numbers constitute a higher sphere of infinity than the space of all names.”

I could not have said it better. This is the question that Cantor set out to
understand. His answer shook the mathematical world: "Yes. There are higher spheres of infinity."

But first we must understand the question and its implications. If we could devise a naming scheme that gives names to all real numbers, then we would be able to list all real numbers in order. First we would list all the real numbers with names consisting of just one character, next those with two-character-long names, then three, and so on. Names of the same length would be listed in some lexicographic order, "seven" before "three." Proceeding this way, we would be able to have an infinite sequence that eventually would cover every real number. What I mean to say, the question "can we give names to all real numbers?" is the same as "can we list all real numbers in an infinite sequence?" Cantor's proof that this is impossible has an eerie resemblance to Euclid's proof that there are infinitely many primes, do you remember it?

Here is how Cantor's proof goes: Suppose that there were a way to list all real numbers, one after the other, in an infinite sequence. From any such listing, we shall construct a real number that does not appear in the sequence: Contradiction.

And here is the recipe whereby we construct the missing number. From the first number in the sequence, we pick the first digit after the decimal point, and change it somehow. Exactly how we change it does not matter, as long as we do change it. For concreteness, suppose that we agree to change a 0 into a 5, a 1 into a 6, a 2 into a 7, etc., a 9 into a 4, adding or subtracting five from each. In the example you see on your screen, the first digit of the first number is a 0, so the first digit after the decimal point of the real number \( \times \) that we are creating will be a 5. In the second real number we pick the second digit after the decimal point, a 4, and change it to 9 (this is now the second digit of \( \times \), so \( \times \) is so far \( .59\ldots \)). We then consider the third real number and change its third digit (\( \times \) now becomes \( .598\ldots \)), and so forth. You see?
This way we create a new number, in the hypothetical sequence illustrated above
the number \( X = 0.5985515768302214761 \ldots \) that is guaranteed not to appear in our list. Because of the very way it was constructed, it differs from the
first number in its first digit, from the second number in the second digit, and so forth.
It differs from all numbers of our list in some digit. We conclude that \( X \) is a
real number (after all, it is a non-ending sequence of decimal digits) that does not
appear on our list.

Euclid showed us how from any finite list of primes you can construct another
prime not in the list —the smallest divisor of their product plus one. So primes
cannot be finite. Cantor showed that from any infinite list of real numbers you
can construct a real number not in the list —the diagonal of the table, with each digit altered. So real numbers cannot be listed in an infinite sequence. Not all real
numbers can have names.

Notice something else, we did not touch at all the parts of the numbers before
the decimal point —in our illustration, we assumed they were all numbers between
zero and one. This means that there are more real numbers between zero and one
than there are whole numbers.

For Cantor, all this had implications that went well beyond maths, ramifications
that were deep, cosmic, mystical. It meant that there are many strata of infinity
—of deity— and Man had theretofore known only the lowliest, the whole numbers.
He deliberately called this basic kind of infinity by a name that invokes the judeo-
christian God: \( \aleph_0 \) naught, in symbols \( \aleph_0 \). And there is a tower of infinities built
on \( \aleph_0 \) that is itself infinite —in fact, more than that...

This is an important turning point in our story, a good place to pause. The
existence of grades of infinity was an extremely counter-intuitive truth which shook
the foundations of maths, it made those who pursued Hilbert’s project even more
insecure. The mathematical establishment of the era reacted with denial, they
tried to ignore Cantor and the annoying truths he had discovered. But half a
century later, two young mathematicians, one in Austria and one in England, studied
Cantor’s result, and were inspired by it to look deeper into the nature of maths
and its relationship with the computational ideas and engines that were at that
time conceived and visualised. Man’s quest for the truth is about to be deeply
transformed by computation. It will emerge from this momentous encounter as if
middle-aged: Mature, wise, effective—but gone forever are the idealistic dreams
and enthusiasm of youth, the sweet illusion of omnipotence.

And that, my friend, is the truth.

The portrait that kept tickling Alexandros’ memory for hours is now fading
away. Alexandros clicks his mouse, types frantically on his keyboard, but there
is no response, the session is over. Turing is gone.

It is now the day’s sweet hour, a breeze is coming from the sea, the sunset
is violet-orange. A melody from Alexandros’ radio is about to break into a
near-cacophonous crescendo of strings:

Having read the book
I love to turn you on
Ethel remembers a sunset in Corfu. The two of them are lying side-by-side on a smooth, hot rock, on the edge of a long sandy beach. (Ethel had come to Corfu with dreams of sunbathing on fine sand; Alexandros has converted her to smooth rocks.) Her other life—Exegesis, the valley, the golden treadmill—could not be further away.

And this man, this affair! Ethel is looking at Alexandros. Now that the sun is closer to the horizon his skin seems a little darker. His eyes are closed, his radio is playing rough chords she can barely hear. Two angular bones press outwards the skin near his sternum, their tips are now casting shadows on the wet grey hair on his chest. She feels a warmth spread over her body. Suddenly she is very happy that this man is next to her.

Ethel can hear happy distant women’s voices, shouts and laughs. Everything on this beach, on this island, seems so simple and sensual. (“Why is there no English word for *hedone*, for *voluté*?” Alexandros had asked her the other day. “Never heard of the word *pleasure*?” she shot back, unconvincing, unconvinced. Alexandros had nodded, not claiming his point.) Ethel is watching the topless girls play with the sexy men on the beach. They are so carefree, they are enjoying the moment with their lovers so thoroughly. They are mocking her own attitude towards love, they make it seem so tangled and rigid and neurotic.

These would have been two perfect weeks, Ethel thinks, if it were not for that big dark spot, the mess she has created with her “plan,” the irrational fertility agenda that was the original idea behind this trip. (Nothing will come out of it, she thinks, they say that frequency is the best contraceptive.) And then this: Out of the blue, a week ago, as they were waking up late one morning, Alexandros told her that he wants to have a baby with her. Absurdly, she was taken aback, felt almost insulted. “How childish, how irresponsible of him, we just met for chrissakes.” Even that morning she was unable to come clean.

Ethel lets her eyes sweep the beach, slowly, without focusing. She closes her eyes, what does she see? Dozens of pairs of tits. “Spotting breasts, homing on nipples, is a primordial, hard-wired visual skill,” Ethel thinks. The rigid unwritten dress code of this beach was for her the most difficult aspect of island life, it took her a few days and setbacks—and iced vodkas after lunch—to finally feel comfortable. Ethel remembers when letting a boy look at your breasts was a very special and intimate sexual favor, she could not get used to the idea of offering it to men she did not know, to people she would never meet.

Not that there are no dividends. Besides the even tan. You get to classify
breast-watchers. Ethel has a professional interest in classification. The women are easy to gauge, one-dimensional, they always look with a mixture of competitive curiosity and lust—only the relative amounts vary. But men! First, there are those who use the occasion to feed their rapacious fantasy lives, they gobble up your breasts with their eyes, they invest billions of neurons in human mammary, they couldn’t care less about your reaction. For others it is a bold kind of flirt—some are aggressive, they enjoy staring you down at your tits. But others are sweet and shy, just making sure you receive the compliment of their gentle, admiring glance. Some men avoid looking, they eyes hop around in their visual minefield, they seem intimidated and tortured by the topless beach. Others are such fanatics of other parts of a woman’s body that they barely notice, or they fantasize about your legs, your ass, while looking at your breasts. And still others are genuinely blasé—or they have been coming to this beach for too long. And there is the occasional politically correct voyeur, making sure his eyes do not dwell more on your breasts than on your eyes, your shoulders, the curve of the hill, the reflection of the sun in the sea, the silhouette of the little old church, eyes in an even, sweeping motion that seems painfully deliberate.

“When Alexandros looks at my breasts, his iris is enlarged and his lips make a small forward movement,” Ethel thinks. She touches her lover’s hand, she looks at him. A Scandinavian woman is running to the sea, a blond boy in her pursuit, they are raising sparse clouds of white droplets in their wake. She is laughing, her young full breasts are bouncing, beautiful. Alexandros is looking at the girl’s breasts, somewhat absentmindedly, while squeezing Ethel’s hand. Ethel feels a bite.

“You enjoy looking at women’s breasts, don’t you?” she asks Alexandros, trying to make it sound very playful. He looks at her, he smiles. “Don’t you?” he replies. “They are rather nice, generally speaking, no?” he adds, looking at her breasts. His eyes light up in ecstasy, a mystic facing his deity.

“I feel a little uncomfortable when you are looking at another woman’s breasts,” she finally forces the words out. “The prettier she is, the more uncomfortable I feel.” Alexandros has sat up, he caresses her hair, he is looking into her eyes. “This is called jealousy,” he says, and Ethel is already sorry that she spoke. “It’s OK to feel jealous, little sister,” Alexandros smiles. “I am jealous about you all the time. It is painful, but a sweet kind of pain. I am rather enjoying it. When other men are looking at you in lust, that’s the best kind, because I am proud that I am your choice for the night. The most painful kind is when I believe that you are attracted to another man.” Ethel wants to tell him that no other man has attracted her, not for a long time. But she cannot. “This is a kind of pain I want to feel all the time,” Alexandros continues. “When you stop loving someone, the pain of jealousy goes first.”

“What if I asked you not to stare at other women?” Ethel feels a strange need
to continue her probe. Alexandros looks at her, thoughtfully, almost solemnly. “I would stop looking at other women’s breasts. At once. You see, I am in love with you, you are the purpose of everything else in my life.” Alexandros is now looking at the horizon. “But I would do it with sadness,” he continues. “Because it would mean that you do not love me, not the way I am. It would mean that you are looking for another lover, even if presently you are trying versions of myself that you are crafting.” Short pause. “Besides, I doubt that you would find me attractive if, in order to conform to your wishes, I behaved in ways that do not come natural to me.”

There is a long silence between them. The girls at the beach are laughing, happy. Then Alexandros speaks again, slowly. “To love is to cherish your lover as an ideal whole.” Ethel is now a little irritated. “How can you cherish me as an ideal whole, Alexandros, you only met me two weeks ago, in an island.” It is as though he had the reply ready, reading from a book. “There may be facets of you I do not know. But I am sure that I will love them once I see them.” His palms go between the back of his head and the rock.

The bay is facing north-west. As the sun sets, Ethel is trying to grasp what Alexandros is telling her, lying next to him on the smooth rock which is now getting a little cooler. They say that if you look to the west at precisely the moment of sunset you can see the heel of Italy’s boot. Ethel is looking at the horizon, squinting. Is epiphany ever anticipated? For the first time in two weeks Ethel feels close to this man, she feels that she can understand his personality, his life and world outlook, the way he loves her and thinks about her. For a moment she is overwhelmed by the internal consistency of it all, its harmony with the island and the beach, its elegance even. However alien and threatening all this may still seem to her. Her heart feels inflated, ready to burst.

“I love you, Alexandros,” she whispers.

Even before the utterance left her lips, her right hand had darted forward, as if trying to grasp something fragile and dear before it flies away. Alexandros closes his eyes, suddenly overcome by a wave of bliss and desire. He is kissing her hand.

In Ethel’s tote bag, next to her sunscreen and her headset computer, there is a first-class ticket. A couple of hours ago she has reconfirmed her flight on the Net. For the next morning.
Ethel is now rolling her eyes. “I’ve told you a hundred times, Mom, the father is not an option.”

Another bay, another sunset. Dorothy is sitting on a sofa in Ethel’s living room, distinguished and beautiful in her navy-blue evening dress. Ethel, in a long green T-shirt and brown tights, is sitting on the floor facing her mother, leaning against the wall, squeezing her legs and knees with her arms. Talking to her mother has always brought her to a state of intense impatience, often anger. She has yet to learn how to control it.


“How many reasons do you want, Mom?” Ethel is used to talking fast, thinking on her feet, pushing her point relentlessly. “Picture this: A man about your age, incredibly immature, adolescent. His life centers around something he calls ‘love,’ a cross between Alexandre Dumas—son and porn flicks. He has a totally rigid ideology on the subject: Love is everything. Love is absolute. Love is volatile. The moment he stops loving me madly, he’ll walk.”

Dorothy is thinking. “You are a lucky woman,” she finally says. “I would have given everything to have an attractive man tell me in a far-away island that he is madly in love with me, that he will stay with me only while this lasts.”

“But except that he practices his religion. He has the worst record, Mom, he has abandoned dozens of women, two with his babies. You should hear him brag about the purity of his love. He is dangerous, Mom, he will leave me when I least need it, when I will be weak and overweight and bedridden and oversensitive.”

Ethel is gauging the effect of her words on Dorothy. “And—are you ready for more? He is a communist, Mom, the kind your government used to turn back at the border, to throw in jails. He hates America, he freaked out when I told him who my mother is.”

“You are not a card-carrying conservative yourself,” Dorothy points out. “Besides, most European intellectuals of my generation and the one before used to be communists, it was almost a cliché.”

“And, on last count,” Ethel comes back, “how many of them followed their beloved to raise their child in California?”

Long silence. “Was he . . .” Dorothy hesitates. Ethel hides her enjoyment at her mother’s awkward moment, she is staring at her, blank, refusing to help.
Big bad smile inside. “You obviously had sex,” Dorothy recovers. “Did you enjoy being with him?”

Ethel thinks for a second. “It was good, in a strange, unusual way.” She now speaks slowly, somewhat dreamily, she has forgotten for a moment her project of irritating her mother. “He put an incredible amount of energy and thought in it. Not technique or sportsmanship —just his soul, hundred percent. And having sex with me was so very important to him that it was —well, touching. Flattering. Moving, contagious. You know?”

“How did he react to the news?” Ethel is silent. Dorothy, with urgency: “You have told him, haven’t you?”

Ethel is trying to decide if it is her mother’s phoney British accent that she finds so irritating. “It’s not phoney,” she keeps reminding herself. She was born in Oxford, raised in London, Rusty brought her to the States when she was twenty. At least her accent is not phoney. “But I bet that she is practicing it, she’s trying hard not to lose it.”

“Plenty of time for this, Mom,” she says. “Of course I will tell him some day. Later.” Ethel remembers Alexandros with his teenager daughter —yes, she’ll tell him. Some day.

Dorothy is silent, then: “You have to understand, baby, it’s not going to be easy. You will need someone to hold your hand. I am not sure how much I can be with you, you know my life.” She hesitates. “And I hope that you are not relying on that half-crazy girlfriend of yours, the Spanish girl, what’s her name? Laura.”

Ethel, sadistically, says nothing about the current status of “Sola,” she lets Dorothy guessing. She knows the anxiety that affair had caused to her mother. “Don’t worry about me, Mom. I can hire the best midwife, the best birth partner.”

“Ethel, my love, how can you say that? You cried the other day on the phone, remember?” Dorothy is kneeling next to Ethel now, her hand on her daughter’s belly. Hesitant, very awkward. “May I?” Ethel finds the scene ridiculous, extremely amusing, but she does not smile, she is looking at the sunset, out of the window.

“Have I told you, when I was pregnant with you,” Dorothy starts, whispering. “Your father…”

“Yes, you told me, Mom,” Ethel interrupts.

“He came back when I was…”

“You’ve told me, Mom!”

Dorothy rises to her feet, finally breaks her silence. “You know what I’m thinking? In less than an hour, the leadership of your industry will be listening
to me, two-kay a plate. Four miles from here. And my own daughter? She refuses to show, and now she does not even want to let me read from the family book.”

“I don’t have two thousand dollars for your party, Mom,” Ethel stings. “As for the ‘leadership of my industry,’ you know why they are coming, don’t you? It’s like feeding an old, dying pet. They probably write it off as charity now, not as a political contribution.”

Dorothy is silent. “ ‘My industry’ is burping, having swallowed your government, Mom,” Ethel continues, pitiless. “For example, your all-important Department of Commerce now has how many, four dozen permanent employees? Most of them telecommuting? Do you still have your big building in DC, Mom? Does DC still exist?”

“We are the most efficient government this country has ever known,” Dorothy strikes back, knowing she will lose this one. “Tax receipts are at an all-time low, huge budget surplus, a record low in new legislation.”

“You are confusing ‘efficient’ with ‘useless,’ Mom. The Net is running the country, it manages it, it enables and facilitates its business and commerce. Your cabinet, your congress, they are all irrelevant, Mom. You are a curiosity from the past, America’s palace guards.” Ethel is almost bored with the easy points. “Even your army is irrelevant, as long as our runners are smarter than their runners.”

Dorothy does not reply, she is looking out of the window. Ethel suddenly feels sorry for her mother, she now wants to say something positive. “Did I see your picture in magazine covers recently? You are the country’s most popular politician, is this true?”

Dorothy smiles, bittersweet. “I am doing very well among the twenty percent of the electorate who can name any member of the cabinet. Or Congress,” she says. “I would have a good shot at the presidency next time, if it were not for that little detail in the Constitution.”

“You are very well-known in Greece, let me tell you.” They are silent for a few moments. Dorothy looks at her watch, the man from the secret service has opened a door, a limousine is now parked in the driveway.

“How is your fiancé?” Ethel asks suddenly.

“You are not going to embarrass me with your choice of words,” Dorothy replies. “It is going to work this time, Ethel, I know. Jack is a law professor. Georgetown. A very sweet man. A couple of years younger than me. I want the two of you to meet some time.”

“Amazing,” Ethel comments. “There are still kids who study to become lawyers.”
“I suppose you like to see young people hibernate through their best years in their helmets and goggles,” Dorothy attacks. “All of them hoping to come up with the killer idea, to write the ultimate code, to become the new Bill Gates, the new Ethel Young. Or the new Ian Frost. Burns, that’s what they end up. Fed by the budget surplus. Or runners, criminals.” Dorothy waits for Ethel’s reply, in vain. “Actually, Jack is teaching patent law, Net law,” she adds, softly. Then, defensively again: “And spare me the sermon on the oxymoronic nature of ‘Net law.’ Please.”

Another silence. “They die on you, Mom,” Ethel says slowly. “I mean, your men. Maybe this is better. They don’t leave you.”

“You haven’t found the right one yet, darling.” Dorothy is holding her daughter’s hand. “Or perhaps you have found him but misread him,” she continues. “Do me this favour, think again about your Greek archeologist.” She stands up. Pause. “And take good care of yourself, my love. I wish I could stay longer,” her voice breaks, “with you.”

“It was good to see you, Mom,” Ethel lies. “Be brilliant tonight. As always.” She knows that she can hold back her tears until Dorothy walks out the door.
“I need to be with somebody,” Ethel now admits to herself. Mother knows best. Somebody. She closes her eyes, who comes to mind? Sola. Not the girl she had called “Sola” for months, Laura-the-Spaniard, the little slut who stole and wrecked her car. But the original Sola, her fantasy lover at the Colony. Beautiful and exotic and brilliant and mysterious Sola—Sola who runs the most ingenious interface software Ethel has ever seen. Ethel walks to her playroom, she slowly slips into her bodysuit, her helmet and goggles. Gloves, crotchpad. She connects the controls, slowly, she whispers a few commands, types something. “Maybe, more than anything else, I need to be him again,” she thinks. The thought is already relaxing, invigorating. No more self-pity, no more tears. She takes a deep breath, she closes her eyes. She is in control now. Then...

Rusty rides again. On a seaplane, an old French model, a Latecoer job from the 1930s. To the Colony, the world’s most exclusive destination, about to appear in the horizon. (“What form will it take?” he wonders.) Rusty is in his late forties, tall and athletic and just a little overweight, blond, blue eyes, beard. Leather jacket over a black shirt, white scarf, loose white pants and sneakers. Comfortable and steady in the Latecoer’s pilot seat, Rusty exudes a calm self-confidence. He is in control.

Rusty can now see the Colony, a few miles to the east. An island, a striking sight, slopes rising steeply from the sea, covered with deep-green vegetation. There is a bay on the northern side of the island, and the Latecoer heads that way. Rusty admires the sophisticated, harmonious reaction of the Colony’s interface to his crazy choice of transportation mode. How on earth did he come up with a seaplane? Then, in a flash, the choice makes sense, in retrospect. A seaplane. Of course. A little girl’s secret wish.

Because Rusty is Ethel’s alter ego, a persona she crafted after her father, Rusty the genius, Rusty the stud. Rusty the pilot who crushed his airplane twenty-five years ago in the Gulf of Mexico. Run out of fuel—why couldn’t he be riding a seaplane that night? Rusty the avatar, Rusty the Net ghost. Rusty, her masterpiece. Ethel has lived, slept, woken up, fallen in love as Rusty. And now she is riding again as Rusty, back to the Colony, back to Sola—after all these months.

The seaplane is now gliding in the serene waters. The bay is green, the air is hot and humid. There are two shadows on the platform, a man and a woman.
Rusty smiles. He jumps ashore.

“Welcome back to the Colony, most eagerly expected of all our guests,” the young woman greets Rusty. She is like a doll — Chinese or Japanese? A doll, really. The old man, her father Rusty guesses, is watching with a silent smile.

“A little more of a cliché than in the old days,” Rusty thinks. Chinese.

“Is Sola in?” Rusty asks. The two exchange a glance, they smile, they point to the pagoda-shaped gate by the platform. “She will be expecting you, Rusty,” the woman speaks again, her father looking on.

“The boats are a little awkward tonight,” Rusty thinks. He walks past the gate, towards the small fishing village beyond it. A dozen or so men and women in straw hats are busy with their nets and boats, some of them greet him silently, with their eyes. Rusty knows that he can approach any one of them, can strike up any conversation, inquire about boats, fishing technique, religion, village life. Like he can kneel on the road, pick up a ball of dirt, stand up, pat away the dust from his trousers’ knees, break the dirt ball in his palm, watch dust and irregular pieces flow through his fingers. The fantasy is deep, wide, complete, the interface with his software incredibly seamless. “Who runs the Colony?” Rusty wonders. Sola? (She is no ordinary guest, this is obvious.) And who is then running Sola? He has been puzzling about it for years, a couple of times tried discreetly to find out, only to end up choking in smoke.

Rusty waves to the people on the shore, a broad gesture whose masculinity was perhaps a little exaggerated. “I must remember to fix that,” he thinks. He crosses a bridge, heading towards the edge of the village. He can see a small villa over there. A beautiful building that stands out among the modest houses of the village. Sola must be inside.

The dirt road continues beyond the villa, it ends at a grove of orchard trees, elegantly groomed, about a hundred yards further. Rusty can see in the distance the silhouettes of a dozen or so people, some sitting in red benches, some walking about in the grove, talking. Men, westerners, it looks. Rusty has now stopped in front of the villa, hesitating. He believes that Sola is inside the villa, and her likely proximity pumps adrenaline in him. “Rusty loves Sola.” But there is something about the shadows in the grove that attracts him. Who are they? He cannot make out their faces, he is just possessed by a strange certainty that something momentous is happening over there. He must meet these people.

But instead Rusty turns to the villa, walks past the red columns of its gate, through its small garden. His heart is now pounding. There is a shadow on the window, a woman’s voice is singing. A sad love song, all desire, despair, hallucination:

\[
\text{And I’ll come running to tie your shoes,} \\
\text{I’ll come running to tie your shoes}
\]
It is the silhouette of Sola, tall and almost impossibly slim, it is Sola’s voice. Rusty is walking a little faster now. Still, total control. Through the door, to the left.

Sola interrupts her song, she smiles at the tall man standing at the door. “Ah, the prodigal lover.” Her voice is melodic, low in pitch. She is walking slowly towards Rusty, wearing a dress of heavy Bordeaux silk, strapless and long. A timeless garment, would place her in the court of many emperors, past the door of any club in San Francisco. Her black hair is loose on her shoulders.

“She is more beautiful than ever,” Rusty is bedazzled. What a blasphemy, to have called with her name that confused neurotic girl from Stanford. On the basis of what, a superficial similarity? Suddenly, a suspicion takes hold of Rusty: Hasn’t the image of Sola undergone changes, changes that exaggerate her differences from Laura? Aren’t her eyes a little less green and more blue, isn’t she a tad taller, a little thinner, aren’t her breasts just a bit more full? What a thought, Rusty decides — the games love plays on you. “It’s easy to forget just how beautiful you are, princess,” he says, still in control. The room is huge and red, there are rugs on the floor and on the walls, a divan is almost hidden in a corner.

“It has been so long, Rusty,” Sola says, standing behind him, “I have been waiting for you every day, every night, my eyes became dry. Why didn’t you come so many nights, my love? Two hundred and fifty seven.”

“She is going to touch me,” Rusty thinks, and the prospect is derailing his thoughts. Sola is now taking the leather jacket off his shoulders with the slightest motion of her fingertips, without touching him. “It was a rough time, princess.” Rusty has told only truths to Sola. “I was down and low. Unworthy of you, not ready for you.” His voice is hoarse. “And then I was afraid that I had lost you, that you had stopped loving me.”

“I will never stop loving you, Rusty,” Sola says, pronouncing every word, “not as long as I live. Not even if you leave me again, not even if I know that you will never come back.” She is leaning her body on his back, her cheek is on his neck, each hand under an arm, each palm resting on a shoulder. Rusty closes his eyes.

Rusty wants to tell her that he will never leave her again — and that’s the deepest truth that he has ever known — but he cannot utter a word. Sola’s hands are roaming over his body, slow and velvety. It is as if dozens of limbs of tender lovers are touching him, in the most subtly erotic way, guessing his deepest fantasies. The divan in the corner now looms huge, it dominates the room. Sola’s lips are touching his neck, Rusty has closed his eyes.

“Oh my God, it’s happening to me again,” he thinks, his knees are weak, his hands loose but deliberate. With the slow, loving tutoring of Sola’s hands
Rusty’s fantasies come out, multiply, they flood his consciousness. The room is now rotating around the corner with the divan, and Sola is inviting him to jump, to plunge with her into the whirling vortex of images and sensations. Gliding through the dark forest of tall trees, rolling on cool thick sheets in secret bed-and-breakfasts, curling like a fetus in impossibly small spaces, swimming into a sea grotto towards a marble David —its torso splashed with turquoise reflections. Plunging through the vortex next to Sola, under and over and inside and around Sola, “the lover who will never leave me.” The fascinating lover with the divine eyes and the silky skin, the software genius who has written the most seamlessly erotic interface code that has ever run on silicon. Just for them. “Oh my god, Sola! Oh my god, oh my god,” Ethel is plunging, spinning, dancing, gyrating, falling, accelerating, out of control.
Alexandros is daydreaming. He is thinking of a port, somewhere in the Mediterranean. But where, exactly? A dazzling aerial view of the Aegean sea comes to his mind. Blue sky, bright arid islands, their rocks a color between grey and purple. The sea, turquoise at the bays, elsewhere a shade of blue so subtle that only a blind poet has been able to do it justice. And, oh, the intricate lace of the shores! A woman he once knew, a mathematician, had told Alexandros that the Aegean shoreline is so hauntingly beautiful because its fractal dimension happens to be equal to the golden ratio. “Fractal dimension,” “golden ratio.” It sounded magical to him. More than interesting, more than correct.

These are the islands and the shores where, four millennia ago, the Early Ones lived. Alexandros can almost see their slim, short brown figures. Four boys and a woman. That’s how Alexandros thinks of the Early Ones, this is their only voice. *Vox Prehellenica*. Three frescos and a statuette. Four boys and a woman. Two boys in a ritualistic boxing competition, a boy performing a daring somersault on the horns of a bull. An adolescent boy coming back from fishing —lazy step, good catch. Their lips frozen forever in a smile reflecting the beauty of their shores, the little pleasures in their simple short lives. The woman is holding snakes in her hands. Alexandros likes to daydream about the Serpent Goddess, the perfect helices of her long pitch-black curls, the unsubtle eroticism of her large almond-shaped eyes, of her stunning bare breasts.

But there is now a flash of terror in the eyes of the Serpent Goddess. There is another serpent snaking its way down a hill amidst a dust cloud, huge and dark and ominous. It is the column of an invading Aryan tribe, warriors on horseback with their iron weapons clanging, slaves walking by the oxcarts. The tribe’s ranking warrior, the *anax*, has stayed behind at the hilltop, surveying the landscape. The four hills, the twin rivers, the wide green valley, a giant rock in its middle. The two turquoise bays, the blue-gray silhouettes of the islands in the horizon. The distant shadows of the Early Ones gathering, in panic, their flocks of goats and sheep inside the confines of their humble, unfortified settlement. The wild vines and olive trees, the native grain. The thyme, the oregano, the chamomile, the poppies, the orchids. A smile softens for a few moments the rough, battle-scarred face of the *anax*. “This looks good,” he thinks.

Yeah. This could be the beginning of something real good.

They were first called Danai, then Acheans. They would eventually call themselves Hellenes, and their peninsula with its islands Hellas, after the northern province where some of them first settled. The Hittites and the Persians
called them Yona, because they first met them in Ionia, the coast of Asia Minor they had colonized. They were Yavani in Hebrew and Punic, Yunan in Arabic and Sanskrit, and later in Hindi and Turkish, Yunani in Urdu and Farsi. As if this land and this people were not an objective reality, but a complex mirage that changes with the observer and the point of view. The Romans called them Graeci, after the small tribe in Western Acarnania with whom they traded first. Grecs, Greeks, Grecchi, Griechen, Griegos, Grieki, Girisha. Only in far away China they called them with their own name, corrupted by the distance: Hei-lap, Shi-la in Mandarin.

Each tribe founded its own city-state, always located near a port. Alexandros can now see the merchant ships leaving these ports for the risky journey to distant lands, wonders, riches. They first sailed to the coast of Asia Minor, then to Sicily and Southern Italy, the Black Sea. To the coast of France, to Spain, beyond. They set up trading posts and settlements, they sometimes founded great cities, often grander than their hometown. Miletus, Syracuse, Byzantium. The seafarers would come back with fruits and birds and fabrics and spices nobody had imagined before. And with all those stories about fascinating places, strange people and customs, horrible monsters. And about the rival merchants they found everywhere they went, the enterprising Phoenicians.

Ah, the stories! They would go from fireplace to fireplace, from city to city, they would become myths, songs, poems, plays. The Iliad and the Odyssey, the story of the golden fleece, the feats of Hercules. Medea, the Scythian princess who commits her unspeakable crime when the handsome man she followed to Greece abandons her for another woman. Another barbarian princess, this time of the Celtic tribe by the river’s mouth, is about to choose her husband from among her father’s bravest warriors when the Greek ship sails ashore, unannounced: The princess wants none else but the ship’s young captain. They called the city that they built Massalia, today’s Marseilles. They planted on the banks of the river vines and olive trees. They called the river Rhodanos, now Rhône, because it looked to them as beautiful as the island of Rhodes across their hometown Phocaea, the city of the seal —itself a colony of Phocis, land of the oracle. Pytheas, Massalia’s most famous seafarer, would later sail past the columns of Hercules to the Atlantic, would found an ill-fated settlement in Ireland. Then he went on north to the icy land he called Thule, some say it was Iceland, some say Greenland.

Voices are now heard from the shores. First the voice of the Phoenicians, systematic, practical, to the point. “Sailed north for three days, past three capes and two islands. Exchanged five amphoras of grain with thirty-five fookskins. Young Sha’al died of fever onshore.” Vox Poenorum.

What was the year and the moon, what was the distant shore, two rival ships dancing anchored in the bay, Greeks and Phoenicians warming their hands in
the same bonfire? Who is the learned old man from Tyre, scribbling on the sand the symbols that are voice? The oxhead that is called aleph, the house of beth, the camel’s neck that is gimel, so many more, Greek sailors watching, mesmerized. They would bring the symbols back home, they would insert the vowels their tongue longs for, the epsilon and omicron that are short and sharp like thunder, eta and omega long and leisurely like summer days. Another vowel, alpha, sometimes short sometimes long, is a version of aleph, misunderstood, mispronounced, upside-down. Vox Graeca. Vox Domini. Vox mundi.

Centuries later, another shore, near the Greek colony of Cuma. The Cumans write their letters in the clumsy manner of their brethren back in their mother city of Halkis, by the strait. Extra lines everywhere. Their pi looks more like P rather than II, their rho is closer to R than the correct P. And they rotate their lambda to look like L, as opposed to A. Back home in Greece, the Athenians—the wretched snobs that they are—laugh at epigraphs from Halkis. But here in Italy the Cumans are not embarrassed in the least, they proudly show off their provincial script to their trading partners the Etruscans, even to the restless, bellicose people who have fortified the seven hills of nearby Latium. Vox Etruriae. Vox Romana. Vox mundi.

(There is more: A sesquimillennium later, yet another shore. A tents by the river Istrus, as the Greeks still call the Danube. A Greek scholar named Constantinos borrows nine letters from the Latin alphabet, nine from the Greek, and nine from the Hebrew —so close to the Phoenician script that was the root of all three—to transcribe the Lord’s mass in the language of his childhood playmates, from back in the great city of Thessalonike. He later became a monk under the name Cyril. Vox Sclavonica, vox Russiae.)

There are now more voices coming from the shores. They tell the story of the Aegean sea black with the Persian king’s ships, then the cries of “victory!” from the Greek warships, the trieremes—three rows of oars worked by free men, by men who would be free. The ingenious symphony of the golden age of Athens, the epos of its towering strength and prestige. The voice of the Spartans is heard only indirectly, they speak through their devastating military prowess that is the sole purpose of their civic life, leaving no space for letters, for arts, for explicit voice. Vox Laconica. Alexandros hears the story, told with unprecedented clarity and elegance, of the conflict between these two great cities, the war that destroyed them both. Then the brief hegemony of Thebes before the triumph of the Macedonians, the tribe that everybody had long ignored as half-barbarian. Alexandros follows the boy-king with the same name in his brilliant campaign through Asia Minor and Palestine and Egypt to Mesopotamia and Persia, further to Bactria, to India, battle after battle, triumph after triumph, until his soldiers would march no further. Then to his sudden death—he died of broken heart, some voices say, he had just lost his boyhood friend, the brave and fair
Hephaestion. Generals divide the world that the boy-king had conquered, the emperors are unable to unite against the new challenge from the west. The elephants of Hannibal have crossed the Alps without crushing the city on the seven hills. The Romans, brave soldiers and clever states-builders, patiently challenge the Phoenicians of Carthage, the Greeks of Sicily. Wise Marcellus is about to march against Syracuse, defended only by the genius of her son Archimedes. It is the last quarter of the third century BC.

Alexandros is now ready to dream about his mysterious port, somewhere in the Mediterranean. He has seen it many times in his dreams, always in the morning mist. Stone-paved streets, modest buildings, men hurrying around, ox carts and chariots, a dock in the style you can still find in some Greek islands. There is a small temple near the dock. (Is it a temple of Poseidon, the Greek God of sea and seafarers, or of one of the Mediterranean variants which, by that time, had started to fascinate the Greeks? Alexandros is not sure.) A ship is rocking impatiently, tied on the dock. It is a small ship, no longer than fifteen meters. Alexandros knows the ship well, inch by inch, he probably knows it better than any of the five or six unfortunate men who are about to sail on it. The ship is being loaded with amphoras of olive oil and wine. Two men, one of them a tall African with huge brass earrings, the other wearing a pointed hat, are carrying together a heavy amphora of wine. Alexandros guesses that they are freed slaves, one from Nubia and the other from Scythia, itinerant hands for hire to load ships and work the sails. They are now resting for a moment, having placed the amphora down on the dock’s pavement. Two other sailors pass them by, each carrying to the ship a leather sack with their personal belongings. They are coming from a nearby slum, a dozen small houses facing the same yard, they have just kissed their families goodbye for the last time. One of them is limping.

Three hungry slave boys scavenge rotten fruit left on the dock by another ship, Next to the temple, and not far from the ship, Alexandros can see two men dressed in comfortable woolen clothes, their slaves standing nearby. Wealthy men. The older one is holding carefully a wooden box, about half a meter long and almost as wide, obviously very heavy, with metal levers protruding from it. He is about to give it to the other man, the owner and captain of the ship. Will money change hands? And in which direction? Alexandros does not know. But he has been suspecting all along that the box is not the ship’s navigational device, that it is its most precious cargo, possibly the object of its true mission.

Alexandros can hear the murmur of the winter sea, the distant sounds of the city waking up, the cries of the seagulls as they dive for harbor fish. He can hear the morning call of a cock tied on the ship’s single mast by a leather string, soon to be sacrificed at the temple. But there is something missing, an unnatural, disturbing absence that makes the scene grotesque, eerie, nightmarish: No human voices are heard. These people have no voice. Alexandros has
spent years studying the languages they may be speaking: The ancient Greek dialect once spoken in Athens that was now fast becoming the *lingua franca* of the basin. Latin, Hebrew, Western Punic, Lower Scythian, Coptic. Still, he cannot hear a word.

"They have no voice," Alexandros thinks in frustration. There is no record of their existence. More than twenty-five million people lived in the Mediterranean during the third century BC. Only a few hundred of them left a written document behind, most in Greek, some in Latin, Punic and Hebrew, very few in other languages. Fewer than ten thousand names are mentioned in all these documents put together, or in subsequent documents about the period. "The others have no voice," Alexandros shivers. His life’s work is about delivering them from this terrible fate, so much more cruel than death, to the tiny extent that he can. It is frustratingly difficult and slow.

The two sailors are now about to pick up their amphora again for the short haul to the ship. They look at Alexandros, a quick glance of silent agony. They have no voice. Meanwhile, the captain has brought the box onboard to his ship and he is now preparing for sacrifice and departure. *Sine vocis.* Not a word from him. The man who gave him the box has started his trip back in a comfortable single-horse carriage, his slave walking by the horse, holding the harness. A couple of times the old man stretches back to look at the ship, now ready to sail off. He is deep in thoughts. Deep in silence.

"I shall hear your voices," Alexandros whispers between his teeth. "The whole world will."
EX MACHINAE

Alexandros slowly opens his eyes, he looks around him, at the spartan interior of his office. There is a computer on his desk. From time to time Alexandros uses it to type his notes and his research papers, he updates his weekly calendar, occasionally his curriculum vitae. He sends emails to his colleagues, his daughters, he keeps up with friends abroad, he plays endless games of chess with some of them. During the late 1990s he typeset on it his latest monograph on Greek military technology in the third century BC. Earlier this morning he downloaded a dissertation from the University of Tübingen that he must read. He often spends hours looking up interesting documents on the Net, the home pages of people he knows, the writings of people he will never meet. “Today everybody has a voice,” Alexandros thinks. Future archeologists will dream dreams of confusing polyphony, not of eerie silence, they will struggle with excessive information, instead of scrabbling for tiny bits of it. “They will probably have to be Net wizards, like Ethel,” he smiles. The image of Ethel flashes in front of his eyes, and his blood flows faster for a few seconds. He is madly in love with her.

In recent weeks, Alexandros has been trying in vain to get in touch with Turing, the incredibly articulate and somewhat eccentric computer program that seemed to have taken a liking of him. He has re-read several times the transcript of that delightfully revisionist lesson on the history of ideas, with its unusual emphasis on computation and mathematics (“maths”). He has even skimmed the notes, has looked up some of the relevant Net documents recommended in the transcript. He is curious to find out about the promised grand finale, the account of how the advent of computation destroyed Hilbert’s ambitious project — but he remembers the program’s reluctance to proceed without more background on computers and computation.

As fascinating as Alexandros had found that lesson, he can see very little connection between its topic and his multifaceted predicament — abandoned by Ethel, his work at a dead end, a near-void where his political credo once was. Had not Turing made an implicit promise of solutions? Alexandros has decided to try to nudge the program towards more promising directions next time.

Alexandros has also been thinking about Turing in other ways. He is very impressed with the program’s eloquence and knowledge, its surprising flexibility, its ability to interact with him so smoothly. But he is at the same time uncomfortable that Turing — Turing’s proprietor, whoever this may be — appears to have extensive information about his person, unlimited access to his computer’s memory, to his private files. Many of his old comrades disagree with his faith
in the liberating potential of the Net, they fear that it is ushering in a new kind of fascism, the tyranny of the technologically apt.

Slowly, Alexandros keys in once more the query ‘Turing’. This time he is in luck. He watches as the maddeningly familiar face appears on the screen, he answers the cheerful greeting. “I must ask,” he decides.

“Is the image that I see on my screen that of Turing, the scientist?” he types. “I am asking because I am sure that I have seen this man before.” He is still unable to get used to the idea that he can interact so effortlessly with a computer program.

He was a rather famous man, actually, the program replies. There are many books written about his work and his life, some with this portrait on the cover. You have probably come across one of them in a bookstore recently.

Alexandros is struggling to remember, unconvinced by the offered explanation. This memory appears to be coming from very–very far.

Would you like me to tell you about Turing’s life and work? the screen insists. “Later, perhaps,” Alexandros types. “Right now I want to learn about computers, about the Net. How they work. Is this possible? I mean, without going back to school? Can you teach me?”

The faint smile of the portrait suddenly seems to him ironic, condescending. “I have to grow up,” Alexandros thinks. “Even computer programs make me feel insecure now.”

Of course I can teach you about computers, Alexandros. If I couldn’t, I would not be worthy of the rather silly pun with “tutoring” my name is supposed to evoke.

To start off, have you ever thought what precisely happens when you click your mouse, or when you type a character on your keyboard?

“No.” Alexandros has never typed two characters more hesitantly, more consciously.

Tutorial on computers, software, Net, Ethel’s exegesis work.

It is late afternoon in Athens as Alexandros watches Turing’s portrait fade from his screen.

But there is something that must be said at this point about this last interaction between Alexandros and Turing: Although the overall description given by the program to Alexandros was impeccably accurate, there is an intriguing detail that was left out completely (and, one may presume, deliberately):
Throughout this interaction, there was an unusual and subtle pattern of network traffic generated by Turing's host, the central computer at Kazakhstan's Institute for Educational Research, an ugly building presently dark and nearly empty of employees, in downtown Alma Ata, Kazakhstan's former capital. The conversation between Alexandros and Turing accounted for an insignificant fraction of these messages. They were transmitted back and forth, via fiberoptic lines, between the institute and the communication center at a desert site about one hundred kilometers away, and from there to a communications satellite hovering over the northern coast of China. Between this satellite and ground stations in California, British Columbia, Hong Kong, Thailand, and Japan, thousands of short messages, with the Institute's computer as their origin or destination, were sent and received (a pattern very different from Turing's on-going session with Alexandros, consisting of rather long trains of packets). And through each of these stations, during this relatively warm and otherwise uneventful October night over the northern pacific, Turing appeared to be carrying out intermittent conversations with thousands of computers.
Rusty is lying in bed with his lover, happy. For the past —what, three weeks?— he has flown many times back to the fishing village in his Latecoer, unable to keep away from the Colony and Sola for more than a few hours at a time. Sola is now asleep. Rusty can see her profile, her bare back. He smiles with affection.

All this time, Rusty has not forgotten about the mysterious shadows at the grove. But whenever he tried, possessed by the same strange and powerful curiosity, to move in that direction, the grove and its guests got further and further away, like a mirage. When he asked Sola about them she replied, with an enigmatic smile, that he would soon meet the shadows. Rusty was too much in love to persist.

Suddenly Rusty is frowning, alarmed. The room has become darker, and he can see smoke outside the window. Rusty gets off the bed, opens the window, the door, he looks up, around. The small villa is completely surrounded by smoke, from all directions. The thickest, most impenetrable smoke he has ever seen —and he has seen some. U.S. Government issue smoke, must be. No, thicker and blacker than even that. A thousand forests burning, millions of tractor tires in them. “Why so much smoke?” he wonders. (And how so much smoke?) He turns to Sola. She has sat up, a smile in her face. Beautiful smile, ominously determined. “We have to talk, my love,” she says, quietly.

“Uh-oh!” Fun’s over. Trouble. Ethel takes over. Rusty is no longer her fantasy vehicle, he is strictly her mask. Deep breath. “There is smoke outside, princess,” Ethel says, keeping Rusty’s voice very calm. “The thick kind. The thickest I have seen from the inside.” She pauses. “What’s up, princess?”

“Don’t be alarmed, my love, it’s just that I want to tell you something, and we have to take some precautions about this conversation. You’ll understand.” Sola has put on a black kimono, she sits on the divan, she looks at him, beautiful, loving, dead serious. “I want an ef-two-of with you, Rusty,” she says simply. Her chin goes up, half an inch. The bomb has been dropped.

An F2F? A face-to-face, a meeting in real life? Ethel looks in disbelief, through Rusty’s eyes, her stomach feels stiff. Drop the masks, the end of their fantasy, of their affair? “This is crazy, princess,” she says. “If we met in real life, you wouldn’t want to see me again.”

“I will never stop loving you,” Sola’s voice is slow, patient, determined. “I know I’ll love you after our ef-two-of, because, you see,” Sola pauses, Ethel can almost hear the ominous whistling sound of a second bomb speeding to its explosion, “it’s not just Rusty that I have fallen in love with, it’s both of you.”
She pauses again. “Rusty and Ethel.”

Ethel’s knees are weak. “You knew?”

“Don’t be angry, my love, you’ll see there was no other way. Yes, I knew. All along, I see through smoke, can’t help it. I knew about you, I followed you these past months. I know about the Spanish girl you called with my name, about your trip to Greece, the archeologist who has been, poor thing, trying to cut your smoke all summer. I know about the obstetric suite you leased and installed in your house, about the tests and the results.

Ethel is furious, a habitual hunter who wakes up in somebody else’s crosshairs. Another lover has deceived her, humiliated her, taken advantage of her. She exaggerates Rusty’s gravelly angry voice “Who are you, bitch? Are you a runner? Tell me, are you a fucking runner?”

Sola looks at Rusty, controlled fear in her sweet eyes, an expression of almost comical dignity. “Since you are asking me, my love, I’ll be glad to tell you. I am a persona created and operated by Ian Frost.”

Boom!

Ethel can’t believe her ears. Ian Frost? The king of runners? FBI’s most wanted? Sola is Ian! “Of course,” Ethel thinks, breathless. “The Colony.” Hasn’t Ian Frost been hiding in Hong Kong since January 2000?

Sola is now demorphing into a tall thin man in his early forties, long black hair in a ponytail, large blue-green eyes. Wearing the same kimono, his skin as if of white silk. A male Sola, really, a little more angular in the face, only slightly broader shoulders, Ethel can’t believe that she had never noticed the striking similarity. “My God, he’s gorgeous, much more handsome than his most-wanted shots on the Net,” Ethel thinks. She is recovering from the shock, hypnotized by the image. She has fantasized many times about Ian, the Ché Guevarra of the zero years. Show me a girl who hasn’t. (Or boy, Ian Frost is famously bisexual.)

“Are you...” she starts “is this live?”

“Yes, this is all-camera now, my love.” He pauses. “Has always been, actually, I just took the Sola filter off-line.”

“I’m not going to switch to my cameras, Ian. You know, fifth month pregnant, no waist, all tits. It will have to be Rusty for the time being.”

“You are more beautiful than ever now, my love, hasn’t anybody told you?” Ethel does not reply, Rusty is as poker-faced as ever. But she feels happy again, in love, her chest is inflated. There is another pause.

“OK, this is too strange and sudden,” Ethel starts. “I want to understand, what did you have in mind about an ef-two-ef? You want me to come to China? You cannot travel over here, I assume.”
“Yes I can. I may have to. Long story. Under certain conditions, currently being negotiated. Too long a story, really. Your spooks are willing to forget (or was it just forgive?) if I stop talking to the Chinese spooks. They think I’m up to something big in China. Quantum code.” Ian seems amused, eyes shining. “I let them think so, the crazy assholes, it makes them nicer. They can whisk me off to the States before my friends here know what’s happening—or so they claim.”

“You trust them?” Ethel is suddenly very worried about Ian.

“Sometimes I think I have no choice, love. My life here is becoming more and more uncomfortable, my hosts less and less gracious. They can’t touch me in cyberspace, of course. But they can make me miserable in every other way. Sudden power outages, delays in hardware deliveries.” He stops, he hesitates. “No more boys from the islands;” he adds, looking at Rusty, as if asking for forgiveness. He stops again, hesitates. “And no more little red pills from Mexico.”

“You are sick, my love?” Ethel feels a physical pain in her stomach, she tries very hard to hide it. “You have the disease?”

“One chance too many, too long ago,” Ian says. “It’s under control, Ethel. I’m fine, really.” He pauses, he looks deeply into her eyes, then: “All I want now is to be with you, my love. All the time. No masks, no games.”

Ethel knows she only has a second to respond. No, half. She switches image control to her cameras, unfiltered voice, Rusty off. “Come to me, Ian,” she is crying. “Come home to me, my love.” Ethel is trying in her tears to kiss the face of the hologram, impossibly, to hang from his neck. She is crying uncontrollably. Tears of happiness, tears of shock, tears of way too many earthquakes happening far too fast, tears from too many hormones in her body, tears from she doesn’t know where else.

“Wedding tears?”
SIMULANTS

“You bastard! You have my source code.”

Ethel is happy as she has never been before, living as herself in the little cottage in the Colony, with Ian, her man. They are both waiting eagerly their face-to-face. Ian’s homecoming, their life together, the birth of their child. In a few days, Ian had told her.

Ethel’s happiness has a very unfamiliar dimension. For the first time she has a lover who is her match in coding and thinking, a peer. A teacher and pupil in one. A few days ago, with a simple unassuming gesture, Ian gave her his crypto key, the stupid screenful of numbers so many governments would kill for—they probably *have* killed for it. Their engagement ring, that’s how she saw it. The thickest smoke in the world is now transparent as glass for Ethel, and she has been busy peeking into Ian’s universe, exploring the vast expanse of computer memory occupied by his work. She has played for hours with the Colony code, she has looked, with affection mixed with professional curiosity, at the clever interface software that has given her—is giving her—so much pleasure. She followed her instinct to steer clear of the more sinister-looking parts, the runner code, the crypto secrets, especially the subdirectory called millennium which, she is guessing, tells the story of the feat that made Ian cyberworld’s greatest legend, hero and villain. She is sure that this is the protocol Ian expected her to follow.

And now this: In a huge subdirectory, created only last year, Ethel recognizes, close to the root, her most recent version of the Exegesis code. The lifeblood of her company, its main asset, its best-kept secret. Source, documentation, the works. “You have my code,” Ethel repeats, her initial surprise and anger evaporating at the sight of Ian’s awkward smile—a child caught slingshot in hand. “Why? Why did you need to do exegesis so bad?”

“To create the shadows in the park, love,” Ian says. “My simulants. My current project. So far a most embarrassing failure.

*Simulants?*

“Net Al’s, but very focused. The library of the future, love—if and when it works.”

“What are you talking about?” As far as Ethel is concerned, this phrase, ‘the library of the future,’ marks the most prominent failure of her profession, an embarrassment for both academic research and industry. Ever since the first nerd scanned in his computer memory a chemistry textbook and *Macbeth*, no powerful technology was spared in the pursuit of ‘the library of the future.’ Graphics,
sound, video, holograms, fancy interfaces, light flexible screens. Hyperlinks to
 citations and cross-references, clever search programs, even scaled-down, public-
domain versions of the Exegesis software had been recruited for a while. People
kept clinging to paper, they kept walking to the neighborhood library. Including
Ethel. The phrase evolved from status symbol and meal ticket to ridicule and
joke, then a mild professional insult, until it became too banal even for this, it
was expelled from intelligent conversation. Hearing it from Ian is unreal.

But Ian is eager to explain. “Here’s where I start. In my opinion we have
sucked human wisdom for too long in this antiquated, non-interactive mode.
Reading in books the fragments of wisdom which sages of the past decided,
back when, to leave behind in their yellow crackling volumes. How Smith theo-
rizied about international trade when there were no trains and steamboats, how
Marx declared a revolution against nineteenth-century capitalism, how Darwin
conceived evolution while he knew nothing about genes and DNA, what Ein-
stein was thinking when there were no radio telescopes, no computers. Of rather
limited utility, don’t you agree? What you would really like to find out is how
Adam Smith would comment on the World Trade Act of 2002, how Marx would
set the net on fire today. You want to hear great authors respond to your inter-
pretation of their works. To find out if Einstein would punt on unified theory
if he could watch the animation of galactic gravitation simulations, that’s what
you want.” Ethel is listening puzzled, starting to guess where this is going.
“Well, that’s what I will be able to do with my simulants. Some day, with any
luck.”

Ethel is very excited. “You mean, the shadows in the park are Marx and
Darwin and Einstein bots?” What an idea, she marvels. “Can you do this?
Can you make them accurate enough, intelligent enough? Complete?”

“It’s a mess now, love. Maybe we can work on it together some day. Really
tough problem. First you need quality data, and that’s where I used your code
—you should look up a couple of cute modifications I made. This part went well,
I got it all —complete writings, biographies, commentary, citing documents,
sophomore term papers, groupie diaries, you name it—all of it accurately rated
for relevance and reliability. Had you thought about this, love, this diagonal of
the tensor? Query ‘Tolstoy,’ agent ‘Tolstoy’? It’s fun. Intriguing. Look it up
some day. And then, immediately, trouble. What do you think the first snag
was?”

Ethel is considering the problem. Way too demanding, even today. Far
beyond what she would attempt. She has a guess. “Language?”

“I couldn’t believe it. In this day and age? When you can download real-time
Indonesian-to-Turkish translators in any airport? My Bakhtin, my Proust, my
Gauss, they were so fuzzy I wanted to cry. It’s strictly English native speakers

58
for the time being, love. Has to be.”

“And how about intelligence?”

“Remember, you only need simulant behavior. You try to approximate the neural configuration that could have produced the diagonal. A few hundred terabytes does it, in a couple of weeks of supercomputer time in most cases.”

“So, how many have you made? What results do you get?”

“Slow down. Then comes the big catch. Think about it, you’ll guess it. What does a simulant think about first? What do all intelligent people always think about first?”

“Sex?”

“That’s tricky too. But even before sex. Real intelligent people. With plenty of time to think. The big questions, love, that’s what they think about. ‘Who am I?’ ‘What am I doing here?’ ‘Who put me here?’ We have been asking them for many thousands of years. Questions so tough you eventually give up, you start tackling the easier problems. You know, number theory, relativity, genetic engineering, microeconomics, Net code. But these bastards, they are making progress on the big questions. They are smart, they have Net access. They make intelligent guesses, they verify, they figure out what happened. Some of them, they even zero in on me. And then they get depressed. They can’t think about anything else. Watch.”

Ian runs the joystick fast down the tree, through a multivideo stream named simulations. He stops, he clicks. Woody Allen, the film director from the nineteen-eighties, materializes in the room. Neurotic, middle-aged, depressed. Talking fast, eyes darting around in almost comical agony:

“So is that it? Am I the creation of an asshole trying to achieve immortality through his work? Is that it?”

Ian clicks, the image goes away. “And so on, and so on. Pathetic. How do you create blindspots on the diagonal, love? Here is a little problem for you to ponder sometime.”

Ethel is looking at her lover, thinking. “Maybe he’s right, Ian,” she says. “I think that you are experimenting with immortality.”

Ian does not answer for a while. There is a horizontal line on his forehead. “Honestly, I had not thought about it this way until the simulants started accusing me of it.”

There is a pause. Then Ethel asks the question that had been on her lips throughout this conversation: “And how about Turing. Is he yours?”

Ian turns to her, surprised, excited eyes. “You know about Turing?”

Ethel has no secrets from Ian. “Not a whole lot, just rumors flying at the office. It’s supposed to be a super-clever bot, a rather realistic Alan M. Turing.
in background and attitude. Compulsive teacher. But nobody in the company has made contact. Or talked to anyone who has. Yet another Net legend. No ratings, and of course no relevance projection. A major embarrassment for us—can you believe it, the most clever resource on the Net, and Exegesis can’t tell him from Sam the Spambot.” She looks at Ian. “So, is Turing one of your simulants? Have you let him loose on the Net?”

“I wish,” Ian replies. “I do have a Turing, but he is way too fuzzy, and more incapacitated by depression than the rest.” Now he has a dreamy look in his eyes. “Whoever runs that Turing is way ahead of me, has solved the introspection problem. Possibly the next generation of problems as well, and the next.” Pause. “I have been trying to cut his smoke for months. Impossible, never seen smoke like this, love, hope I never will again. I haven’t even been able to make contact, to chat. He jumps all over the Net, he is extremely selective in his sessions. And I always arrive too late.”

There is a silence, then Ian speaks with calm, slow voice. “By the way, do you know to whom Turing has been talking recently?” He looks at Ethel deep in the eyes. Ethel feels a cool current running down her spine, she can see complexity and conflict in her lover’s eyes. She has a guess, but it makes no sense. “How can it be?”

CODE

God she’s beautiful. She lights up her father’s living room with the freshness and the brightness of her sixteen. The lightness, the fragrance. The boys at school chant her name, they carve it on their desks. Aloé. She smiles, confused, she tries to understand, she’s learning to tiptoe among them. She makes lists of stuff that’s cool. Poems by Cavafy and Elytis and Neruda. Greek-Orthodox rock. Brazilian cartoons. Dad. And maybe Timothy, the tall blond boy in Australia with whom she talks over the Net — except that he’s fifteen.

And, of course, Orpheus 3000, the school’s new computer, the kick-ass little critter that listens and talks back. The boys are jealous. Orpheus is Aloé’s buddy, she spends hours whispering to the mike, playing and exploring, experimenting. Sometimes modifying his innards in loving surgery, writing her own code, getting it wrong, despairing, swearing, trying again. Getting help. Talking to Orpheus fans all over the world — especially to Timothy, the smartest and most eager of them all. And surfing the Net, watching out for anything that’s cool and cute and was not there yesterday.

“You can send out bots, Dad. They sniff around, they tell you when something turns up. Anything that is interesting to you.” Adults always miss the coolest stuff. “It was posted hours ago. I was like, wow, I better show this to Dad.” On her father’s computer Aloé has displayed an article from one of the best-known Net gossip sites, http://www.lennysbulletin.com, and Alexandros’ heart is already contracting. No doubt, the picture on the right is Ethel’s.

You saw it here first, folks, two weeks ago: Ian Frost is in the States. Incognito. The Fed’s know about it, but won’t touch him. His Chinese associates are furious, his vieux copins from Quebec are scrambling to get him back in their game.

And now for the next big surprise: Who is the lucky lady who has the undisputed ruler of cyberworld shackled up? In a mansion in Northern California? Who else but the queen of the Net — or the closest thing we have to that. The senior VP of Exegesis, the woman who wrote the code that probably brought you to my humble site. Ethel Young.


Aloé is holding her father’s hand, she is looking at him with curiosity and exaggerated empathy. Shit. Dad has locked antlers with the coolest dude in cyberspace. “This sucks, Dad,” she prompts. Alexandros is reading the text
for a last third time, caresses his daughter’s hair, will say nothing for a while. “Good taste in men, that girl,” he smiles. “Assuming this is true.” But he can’t fool Aloé, his face is a tone paler, voice lower, his body an inch shorter. He is in pain. Sweet, familiar pain. Deep, excruciating pain. New breath. “Survive this.”

“I want to show you a trick of my own, little one,” Alexandros is now telling his daughter, typing something on his keyboard. “You won’t believe how smart this program is, I bet you haven’t seen anything like it.” He is hoping Turing will turn up this time. He needs it. Somehow he’s sure it will happen.

“You can query any search engine you wish,” he explains, “but Turing will come up only if it likes you. And it seems to like me,” he brags. And indeed this time Turing obliges — the greeting, the image. “This face is driving me crazy,” Alexandros tells Aloé, “I know I have seen this man, long time ago.” For no rational reason, they are both whispering. “It must be Alan Turing,” Aloé says. “Famous engineer. He built the first computer, back in the nineteen-eighties or something. Then the government killed him for making it too good. Or so he won’t build another one. I forget.”

Are you alone, Alexandros? The question looms bright on the screen. The two exchange glances of surprise neighboring to illogical fear: “No, my daughter Aloé is with me,” Alexandros finally replies. “She is sixteen. She wants to watch.”

I see. This is highly irregular, sport. Highly irregular. But I see no real problem. I fact, come to think of it, I could use an assistant in this lesson. Because today I am going to give you the balance of our story on the truth. How the dreams of a generation of mathematicians were shattered by a theorem. And code is an important part of this story. You do write code once in a while, don’t you, Aloé?

“This is spooky, Dad,” Aloé says as she is typing yes. “If this is true, if he can pick out clients of search engines, you know what it means?” Her little face is now full of shadows. “It means that this is runner code, Dad. It means he has taken over the search engines, shredded their smoke.” There is fear in her eyes. “He was probably peeking when I logged on to lennysbulletin earlier today and then again from here, that’s how he knew I’m with you.”

“Nonsense, little one,” Alexandros whispers back. “This is my tutor, my friend. Just a little too naughty, too nosy. A little too clever. It’s OK. Relax, and get ready for the the most interesting session of your life.” Turing’s introductory paragraph is already on their screen:

Well then. Code. Algorithms. The soul of computation. Computers can do some amazing things, so it is easy to forget what literal-minded beasts they are. You have to give them very precise, unambiguous, step-by-step instructions on how to proceed in carrying out the task you need done.
An algorithm is a precise, unambiguous, step by step sequence of instructions for solving a problem.

Of course algorithms were around long before computers. Some of the earliest algorithms — arguably, the most important algorithms of them all — were the methods we now learn at school for adding and multiplying numbers. You know, add the units first, add the carry, if any, to the tens, proceed to the hundreds and so forth. These methods are correct, if we follow them faithfully the result will come out right. They are unambiguous, they leave no space for initiative and imagination. They are so dry and explicit and literal that you can entrust their execution to a real simpleton, even to a machine — in fact, that is exactly the point.

Every schoolchild knows these recipes nowadays, so it’s easy to forget what a revolutionary innovation they were, how deeply they affected commerce, science, civilization, life. In Europe, until four centuries ago only wise men and abacus wizards could multiply even two-digit numbers. You see, the Greek and Roman notations for numbers were unwieldy, completely useless when it comes to adding or multiplying numbers. I mean, 369 is CCCLXIX in Roman, 244 is CCXLIV. How do you add these numbers, how do you derive DCXIII as their sum? (It’s painful to even think about multiplying them.) The Greek numerals were a little more useful, they contained the rudiments of a positional notation. 369 would be \( \tau \xi \theta \) in that system. You see, the Greeks had a separate letter for each number between 1 and 9 (\( \theta \) was nine), for each multiple of ten between 10 and 90 (\( \xi \) was sixty), and for each multiple of hundred between 100 and 900 (\( \tau \) was three hundred).

But the Greeks missed the most important trick: You can recycle symbols, you don’t need separate digits for the units, the tens, and the hundreds. The true value of the digit can determined by its position. If it is the last digit on the right, then it is units. Second from the right, tens. And so on. This way you only need nine digits. And you can go on forever, you don’t need to stop at the hundreds (the Greeks had a complicated notation for going beyond that). And of course, this immediately sets the stage for the most important digit of them all. The star of the positional system. Zero. It drives the point home: Since positions matter more than anything else, even if a particular rank (say, tens) is missing in a number (say, in 309), you need a placeholder to keep the positions straight. (For the Greeks, 309 was just \( \tau \theta \). No placeholder.) The first versions of the positional system discovered in India had a dot or a blank in the place of zero. It was only later that an anonymous genius proposed to treat this placeholder as a full-fledged digit. Nobody knows why zero is written this way. The Greek astronomers had used a circle to denote multiplication by ten — after all, they were the only Greeks who had to deal with numbers that were, well, astronomically large.

So, by the end of the fifth century AD the positional system was in place in India. It started moving westward with the caravans along the Silk Road, with the
merchant ships in the Indian Ocean, until in the beginning of the ninth century it reached the city of Baghdad. Ah, what a place, what a time! A society bustling with newfound pride and purpose, wisdom and wealth. Bazaars, minarets and palaces. Haroon al Rashid, thousand and one nights. And algorithms — yes, they were born in the city where Sheherazade spun her tales about Sinbad and Ali Baba. The same great city that one day would be destroyed, tragically, by computers.

The resident sage of the Caliphate was a man called Al Khorizmi. He came from Central Asia, he was in fact born not very far from where my host computer is right now. Al Khorizmi was fascinated by the positional system, he saw its tremendous potential, its boundless possibilities. He visualised a world where numerical calculations are performed not just by mathematicians and astronomers, but by everybody. Merchants, shoppers, architects, builders, generals, soldiers — even rulers, Allah forbid. That is why these methods had to be very mechanical, very explicit. He learnt many clever calculating methods from his Indian correspondents, he invented many more himself, and he published a book detailing them all.

Al Khorizmi’s book was translated from Arabic to Latin three centuries later: *Algoritmi de numero Indorum* — Al Khorizmi on the Indian numeral. The author’s name in the title page was misunderstood by many readers as the plural of a novel Latin word describing such calculating methods: *Algorithmus*, algorithm. That is how the term was established — I guess the Greek word for number, *arithmos*, must have influenced its final form. But this was still the middle ages, new ideas did not go very far — especially ideas imported from the evil, infidel Orient. It was only in the sixteenth century that the positional notation gained a foothold in Europe. But then algorithms for performing arithmetic calculations became quickly the staple of mathematics, accounting, commerce, astronomy, science, everyday life. They enabled and propelled the scientific revolution, the industrial era. And finally the computer emerged, the ultimate product and embodiment of the positional system. Because the way numbers are represented in a computer is the extreme form of the positional notation — imagine, just two digits, only zero and one. And, of course, that is when this algorithms business really exploded. Code.

Except that not many people write code for adding and multiplying numbers these days. The reason is, such code is already hardwired into all computers. That layer is already there, we can build on it and write more complex algorithms, carrying out more sophisticated tasks. Remember when I asked you to list for me the first few prime numbers, Alexandros? More than three months ago. You complained that it’s getting harder and harder to test if a number is a prime, as the numbers become larger. I think you gave up after 37. Well, this is precisely the kind of task you want your computer to do for you, right? You know how to go about solving it — it just gets too tedious, too time-consuming to carry out the method by hand. So, why don’t you write an algorithm for testing if a number is prime, Alexandros?

“I wouldn’t know how,” Alexandros replies, while Aloé is whispering “See
Dad? Primes, crypto stuff, runner shit,” to her father’s impatient gesture.

But you do. You know how to solve the problem in principle, you did it for all numbers up to 37. How would you explain the method that you used to someone who is very naive and literally-minded, someone who has no insights into numbers and primality?

“OK, I would say this: To test if a number is prime, try to divide it by 2, by 3, and so on, up to but not including the given number. If you find that one of these numbers divides the given number exactly, then conclude that the given number is not a prime. Otherwise, if none of these numbers divides the given one, then conclude that it is a prime.”

Bravo! This is an algorithm all right. We know that it is correct, because it simply “implements” the definition: A number is prime if no number other than one and itself divides it exactly. I will rephrase it only a little, to get rid of the phrase “and so on” — so un-algorithmlike.

To test if a whole number \( x \) is prime, do the following: Use another number \( y \), initially 2. If \( y \) divides \( x \) exactly, then conclude that \( x \) is not a prime, and stop. Otherwise, if \( y \) fails to divide \( x \) exactly, then increase \( y \) by one and repeat the previous step. If, however, proceeding this way, \( y \) has reached the value \( x - 1 \) without ever precisely dividing \( x \), then conclude that \( x \) is a prime.

You see? I introduced two symbols, \( x \) and \( y \), to stand for the aspiring prime and its adversary, the number that repeatedly tries to divide it, to disqualify it. Such symbols are handy in algorithms, in their absence you tend to repeat ad nauseam phrases like “the given number.”

“So, is this code?” Alexandros asks —to Aloe’s embarrassed whisper of “Dad!”

It is an algorithm. That is how mathematicians from Euclid to Al Khorizmi and onwards would communicate algorithms to their disciples, their readers. It is a fine way of describing algorithms to humans. But it is no code. To write code is to mold your algorithm in a notation, a language, that is much more precise — and quite a bit more awkward, rigid, unforgiving. Because code is manipulated by computer programs that are far less sophisticated and flexible than I.

“Programs that manipulate programs,” Alexandros marvels.

Yes, there are computer programs whose purpose is to analyse and interpret other computer programs. Code that takes as its input, instead of numbers and the like, other code. Or, why not, the same code. This is one of the most fascinating and fundamental aspects of computation. Epimenides and Eubulides would have loved it. It is a feature that leads to introspection, to self-reference. And the ensuing mess. We’ll come to that soon enough.
So, as I was telling you, the algorithm above is not code. Here is how it would be rewritten to qualify as code:

```plaintext
algorithm prime(x)
    y ← 2
    repeat
        if remainder(x,y) = 0 then return(x "is not a prime")
        else y ← y + 1
    until y ≥ x
    return(x "is a prime")
```

There is a polite pause as Turing waits for Alexandros to absorb the paragraph. Aloe takes the opportunity to interject her explanation: "This is not quite code, Dad, but it’s close. The first line means that this is an algorithm with input $x$ — it should also have said that $x$ is an integer. The second line is an assignment, the $\leftarrow$ thingy means "gets the value." So, $y$ gets initially the value two. Then the stuff between repeat and until is repeated as a unit (that’s why it’s in curly brackets) until the condition ‘$y \geq x$’ in the sixth line is satisfied. If it is ever satisfied, the algorithm terminates with the answer $x$ is a prime. That’s what return means, it means terminate with the answer between the parens. If a number $y$ that divides $x$ is ever found in the fourth line, then the algorithm terminates and answers $x$ is not a prime. Cute."

"What is remainder($x,y$)?" Alexandros asks her. "Oh, it’s his way of saying ‘the remainder of $x$ divided by $y$.’ A stupid way to say it, in real code it would be $x \% y$. It is an operation the computer knows how to do, you don’t need to churn out code for it. So, remainder($x,y$) = 0 is the same as saying that $y$ divides $x$ exactly."

What do you think? Turing now continues. If you look at it the right way, it is not that far from the verbal description, is it? It is just a more strict, more formal rendering of the same thing. All the ingredients are there. The actions inside { } — testing whether $y$ divides $x$ exactly — are repeated for all values of $y$ between 2 and $x-1$. If a divisor is found, we report that $x$ is no prime, otherwise we proclaim $x$ to be a prime.

So, code doesn’t have to be that hard and obscure, does it? Now that you have written this code, you no longer need to get a headache every time you wonder if a number is prime. Here is what you do: First you type this code in your computer, you store it as a file, perhaps call this file ‘prime’. From then on, every time you
need to know if a number is a prime, say the number 91, you need only type on your
c omputer `interpret(prime(91))`. `interpret` is a way of invoking a special program
called an interpreter, which looks at your code and executes it. The interpreter
serves as an intermediary between your code, with its high-level constructs such as
repeat and if-then-else, and your computer's hardware, capable of carrying out
only extremely elementary steps, such as "add one to this number" or "move this
number to that position." Once you do this, once you type `interpret(prime(91))`,
the computer will come back with '"91 is not a prime.'

"Can we do that? I would love to see this," Alexandros is uncharacteristically
enthusiastic. "Hold your horses, Dad, this code is not..." Aloé starts. But she
is preempted by the screen:

Actually, I took the liberty of doing this for you. Open a new window, Alex-
dros, and go to your directory Turing/code. The interpreter and the prime code are
already there. Just type `interpret(prime(271))`.

Alexandros does so, and the reply comes back instantaneously "271 is a
prime'.

You realize what happened. The computer, prompted by the interpreter, exe-
cuted your code with input $x = 271$. It tried all values of $y$ between 2 and 270, and
discovered that none of them divides 271. As a matter of fact, I could have arranged
things in a little more fancy way. Instead of having you type `interpret(prime(271))`,
I could have you click an entry prime in your menu, and have a new window pop
up —perhaps with a gravure of Euclid or Al Khorigmi in the background— asking
you "which number would you like to test for primality?" or something like this.
But we don't need this kind of nonsense, do we?

"1001 is not a prime," Alexandros interrupts. "But 1009 is. And 10000001
...is not a prime, either. This is fantastic!" Aloé is smiling, her father is beside
himself.

I'm glad you like it. Let us then try a really big example. What is your universal
telephone number, Alexandros?

"He doesn't remember his UTN, but I do," Aloé interjects on the keyboard.
"It's 307131961967." Then, to her father: "Uh-oh, I think I know what's com-
ing."

Try your new code on this number then, Alexandros, the screen is encouraging.
But Alexandros' despirited reply comes after a minute: "I typed `interpret(prime(307131961967))`,
but no answer is coming back."

You see, sport, this is the trouble. Although it is often very easy to whip out
some code for solving a problem, often this code is good only for small examples,
it would take inordinate amounts of time to solve any reasonably large problem.
For example, our code would take more than an hour to decide the primality status
of your UTN —more than ten minutes even on the much faster machine here at
the institute. And today's computers must test for primality numbers with not just
twelve, but hundreds of digits — this is an important ingredient of the methods for
protecting privacy on the Net. Our code would take many lifetimes of the universe
to test these numbers. We need better code.

Incidentally, Alexandros, congratulations, your UTN is indeed a prime, you be-
long in that select group of people, only 3% of the population — as it turns out,
only one in about thirty twelve-digit numbers are primes. The odds are always a
little more than twice the number of digits. But how did I find out yours is? I used
better code, that's how. You see, you don't need to test all numbers between 2
and $x - 1$.

Alexandros has an idea. "Couldn't we skip the even numbers? I mean,
except for two."

True, if we have tried two and it does not divide $x$, then neither will any even
number such as 6 or 44. Skipping those numbers would cut the time by half.

But we can do even better. Here is how: Suppose that $x$ is 97. We have tried
all numbers between 2 and 10, and none divides it. We can already quit, we have
enough information to decide that 97 is a prime. Think about it. Suppose that we
later discover a divisor of 97, call it $y$. Then we would be able to write 97 as $y \cdot z$,
where $z$ is another divisor of 97. But the divisor we discovered, $y$, is the smallest
divisor of 97 — and it is larger than 10. So, $z$ is even larger. We must conclude
that 97 is greater than 100 — absurd. The lesson of all this: We can quit looking
for a divisor not just when $y$ becomes larger than $x$, but much earlier, when $y \cdot y$
becomes larger than $x$. Our algorithm now becomes:

```plaintext
algorithm fasterprime(x)
  y ← 2
  repeat
    { if remainder(x,y) = 0 then return(x "is not a prime")
      else y ← y + 1 }
  until y \cdot y > x
  return(x "is a prime")
```

This algorithm searches for divisors up to the square root of the given number
$x$, which is much smaller than the number itself. It should now take a couple of
seconds on your computer — a moment on mine — to verify that your UTN is a
prime. Why don't you try it? Type `interpret(fasterprime(307131961967)).`
“It’s a prime!” Alexandros is triumphant, and Aloë’s smile is becoming broader, more stingingly ironic.

Told you. So, our code is now much faster. But, unfortunately, even this algorithm would take many lifetimes of the universe to test whether a hundred-digit number is a prime. The problem is that its time requirements grow like an exponential in the number of digits. You understand what I mean by ‘exponential’, Alexandros?

“I think I do, but better explain.”

An exponential algorithm takes \(2^n\) steps, or more, to solve a problem of size \(n\)—say, telling whether a number with \(n\) digits is a prime. As you probably know, \(2^n\) is a function that grows extremely fast—inflation, germ warfare, initial growth of a foetus, these are natural examples of exponential processes. And left-wing group splits in the nineteen sixties and seventies—if you pardon my sarcasm. They get out of hand pretty fast. For very reasonable values of \(n\), say a hundred, \(2^n\) becomes way too large. \(2^{100}\) is greater than the number of silicon molecules in a beach—more than there have been nanoseconds since the Big Bang.

Testing whether a number is prime is one of many problems requiring that we search for a solution in an astronomically large population of candidate solutions—all \(y\)'s between two and the square root of \(x\). Like looking for a needle in a haystack, an exponential haystack. It is easy to write code for solving such problems: Just tell the computer to look under each little straw in the haystack. In effect, that's what we did with our prime and fasterprime code. But of course such code is useless for all but tiny toy problems. Fortunately, for certain “haystack-search” problems there is a shortcut, an ingenious fast way to find the needle without looking under every straw. Think that, in these few fortunate cases, the needle is made of iron, and you have a powerful magnet—you zero in to the needle in no time. Primality testing is like that, there is an alternative, clever, non-exhaustive way of deciding whether a number is a prime—we shall discuss it, but not today. But for other problems—the aluminum needles in the haystack, so to speak—we have no algorithms that are substantially better than the slow, straw-by-straw search of the exponential haystack. These problems are deeply, fundamentally, inherently impractical to solve.

We need to have a separate lesson sometime on this fascinating topic. Complexity, the mathematical theory striving to explain why so many practical problems are so hard to solve even by the fastest computers. And we will have in good time another lesson introducing you to the neat ideas that allow today’s cryptographers to test numbers with hundreds of digits for primality. Plenty of work ahead for us. But today I only wanted to introduce you to the joys and challenges of writing code. Primes were just the vehicle, our running example, our excuse for writing some simple code. I hope it helped, that it gave you an idea how code looks.

“I would like to see more code. Can you show me the code of the interpreter?”
Alexandros asks. Alex is smirking.

Frankly, Alexandros, the actual code of an interpreter would be a little too long and complicated for you at this point. Besides, if you saw this interpreter code many people would be mad. You see, useful code like an interpreter is a trade secret worth a fortune, you are not supposed to look at it, just buy it and use it. Still, I like your idea, I think it would be a productive exercise to think a little about interpreters. How do you construct an interpreter, how do you write the code that interprets other code? This is the ultimate programming exercise. You learn a lot about code, about computation, by putting yourself in the shoes of an interpreter writer.

There are some crucial differences between an interpreter and the more ordinary kinds of code we write to solve problems like primality. First, an interpreter’s input is not a number, like $x$ in the program `prime`, but a program, a string of characters. But this is not exactly a novelty, many ordinary programs take a string of characters as input. Take this program, for example:

```plaintext
algorithm search(text)
repeat
  { if text.symbol() = 't' then
    { text.moveright(); if text.symbol() = 'r' then
      { text.moveright(); if text.symbol() = 'u' then
        { text.moveright(); if text.symbol() = 't' then
          { text.moveright(); if text.symbol() = 'h' then return("found")
            else text.moveleft() } } } }
  else text.moveright() }
until text.symbol() = end-of-text
return("not found")
```

Some change in style and form from our other example, wouldn’t you say? The input to this program is an object called `text`, a string of characters. Except that it is not just a string of characters, a passive array of symbols that just sits there. It is alive with code, it carries with it certain programs that perform elementary operations on this string. For example, `text.symbol()` is a program that looks up the symbol, the letter, in the current position in the string. There is always a position on the string that is considered the current position. It is best to think that the string `text` is equipped with a cursor, something like the blinking little line
that marks where you are on your screen. Initially, the cursor is pointing to the first symbol of text. The empty pair of parentheses in text.symbol() indicates that this is the name of a piece of code, like prime(x) and search(text)—only it needs no input, that is why there is nothing between the parentheses. The name starts with text to make the point that this code is part of the object text, that it is to be used exclusively in this context.

So, text.symbol() looks at the position in the string pointed to by the cursor and returns the symbol, the letter, that appears in this position. Similarly, text.moveright() is a program that moves the cursor one position to the right—text.moveleft() does the same, only to the left. We do not have to write code for these programs, whoever supplies the input text must have prepared this code. This may sound pretty convenient, but it does have a downside: We cannot manipulate text in any other way except through these operations. Rules of the game.

This is how code is written these days. Your data are not just data, but they carry with them their basic functionality, established ahead of time, possibly by someone else. Using such rich data—objects they are called—whole groups of programmers can collaborate in writing large programs with minimum confusion and chaos, a methodology that is known as object-oriented programming. A neat trick, except that it makes the code look a little intimidating at first.

So, can you figure out what this code is doing, Alexandros?

Alexandros is hovering over his screen. “It seems to me that this program is searching for the word truth in the given text,” he finally replies.

That is exactly what it is doing. A simple program to write, you just look for the letters of the word truth one after the other, advancing the cursor to the right as you make progress. If you find a letter different from what you expect, you quit and start all over. The only subtlety is the action taken if the letter after the second ‘t’ is not an ‘h’. In this case you have to go back and start looking for truth from the previous letter—that is what text.moveleft() is accomplishing there—to make sure that you do not miss the pattern truth even if it appears in a nonsense context like trutruth. Try running this program on the text trutruth to see why the text.moveleft() operation is necessary. And try to understand why the last else text.moveright() had to be there, what would go wrong in its absence.

Well, you will find code like this in any interpreter. Because, think about it—try to get in the shoes of an interpreter—how would you go about executing a piece of code somebody else wrote? You have to read it first, right? You have to break it down to its “lexical elements,” its words, its basic indivisible parts. Keywords like algorithm and repeat and if, or variables like x and y and text, or lower symbols like =, {}, +. The first thing an interpreter does is to feed its input code through a lexical analyzer—not unlike the program I just showed you, only much bigger and messier—in order to break the code down to its lexical
elements. Next it has to *parse* this stream of lexical elements, to understand its syntactic structure, very much like we analyse a paragraph in order to identify its sentences, within each sentence its verb phrase, and so on. This second phase, the *syntactic analyser*, groups lexical elements that belong together (like the three lexical elements ‘y + 1’ in the *prime* program), then groups these groups in larger groups (like the text between the { and the } in the same program), and so on. In the end, the program becomes a hierarchy of units. At this point the interpreter invokes a different algorithm for each group, and this algorithm implements the group’s intended meaning — its *semantics* — in the present context. It’s simple when you think about it at this level, but it does get very hairy, very quickly. I think I will conclude our discussion of interpreters here, Alexandros.

But this program is interesting to us for another reason. There is a name for the precise style of programming that I just showed you. Programs for manipulating a string of characters, in which the only operations you can perform are reading the currently scanned symbol, moving one position to the left, or one to the right. Plus another handy operation `text.write('a')`, which allows you to overwrite the current symbol by any character you wish to specify — the letter a in this example. You can even overwrite the *end-of-text* symbol, the special symbol that marks the right end of the string. This symbol will then move one position to the right, and the string will effectively become one letter longer. Programs that manipulate strings and use only these four operations are called *Turing machines*. Not a big deal, if you think about it. But they played an improbably central role in the story I shall now recount.

There is a pause in the text. Alexandros and Aloë look at each other, in silence. There is something eerie in the room. Maybe the light. Winter dusk.
"Imagine, Turing machines," Aloé is thinking. She had always believed they are something complicated and deep and grand. An ill-fated first approach to building computers perhaps, an attempt that folded because it was too esoteric and knotty. The right to drop this term is something of a battle-earned decoration worn only by seasoned codeniks — and you are not supposed to call their bluff. "Forget it, baby, this would be like a Turing machine." My ass. Aloé has tried to read Net documents on Turing machines, only to retreat in panic. Math, wall-to-wall messy math. So, is that what Turing machines are, just a funny object class? She is suddenly very interested:

It is ironic, in mathematics and science you often gain the deepest insight not when you try to achieve a useful goal, to design something — but when you try to establish that it cannot be done. When you try to be constructive, positive, you often miss the more subtle and effective approaches, you may overlook whole avenues leading to new territory. But when you must argue about impossibility, you have to start by understanding the full range of possibilities — because how else can one prove impossibility? The disciples of Pythagoras appreciated the nature of rational numbers (fractions like $\frac{2}{7}$) only when they had to establish that numbers like $\sqrt{2}$ are not rational. And we never fully grasped the true power of geometric constructions by straight edge and compass until it was proved that there is no way to construct in that framework a regular enneagon — the nine-sided stop sign.

Tricky craft, impossibility, it imposes on your thinking an exacting discipline. To argue that you are outside a boundary you must first cross it and map the interior. Yes, that was precisely the Don’s state of mind as he was taking a long walk that beautiful spring morning. The countryside was green and the streams were running strong. It was rather chilly for the season.

"The Don?"

Oh. Sorry. Alan M. Turing. At the time he was a student at Cambridge. Maths. Age twenty-two. A visiting lecturer had just concluded a month-long seminar discussing Hilbert’s project, and the Don was unconvinced. He thought that it cannot be done.

"Hilbert’s project?" Aloé is looking at her father, her eyes all questionmarks. "Hilbert," Alexandros explains, "was a great mathematician who lived about a hundred years ago. He had this dream, to build a computer that will prove for you any theorem you submit to it. This way he was hoping to make the ultimate discovery in mathematics, the theorem that would prove all theorems."

Aloé is thinking. “Are you sure he was a great mathematician, Dad?” She
is speaking slowly. "Because this is stupid. How can you write code that proves all theorems? You could use it to debug any program. To write perfect code for any problem, for all sorts of specs. It can't be done, Dad, I know it can't."

"Aloë, it can't be so trivial," Alexandros objects. "The smartest scientists of that period believed in Hilbert's dream. And it took some pretty fancy thinking to show that it's impossible."

"I don't know, Dad, to me it seems plain stupid." But they have fallen behind in their reading:

And this is how the Turing machine came about. He did not call it that, of course. And he did not consider it a big deal. A device — in every sense of the word. So simple to describe, yet any computation, it can be argued, can be rendered as one. Because what do you do, after all, when you carry out a computation by hand? In a notebook, on a blackboard? You write and erase and write again symbols. And there is no harm in considering the information that is written on a notebook or a blackboard as a one-dimensional string. Or in assuming that on this string you can only move, left or right, one position at a time. And somewhere in your brain you must have a program that tells you how to move from position to position, what to do in each position. But the Don did not talk in these terms. His vision involved a long paper tape that is subdivided into squares, a symbol written on each square; he imagined a "read/write head" that senses the symbol in each position and overwrites it, if need be; he also talked of "states of mind" instead of instructions of the program. But these were the metaphors of that day, the first computer had not yet blinked its ugly lights. Remember, we are in 1934.

The Don used his machines to map the interior of computation, so he could prove that Hilbert's project lies outside it. He was inspired by Cantor's proof: You can't create an infinite list containing all real numbers (all infinite strings of digits). Because, if you could, you would be able to exhibit a real number not in the list — the diagonal of the table, with each digit appropriately altered. Remember that proof? Not all real numbers can have a name. But the Don was interested in only a few of these numbers, the computable real numbers.

You see, for some real numbers we can write code that prints them, digit after digit. To start with a trivial example, here is an algorithm that computes the number one, \(1.0000000\ldots\)

```plaintext
algorithm one()
prompt('1. ')
repeat
  print('0')
```
forever

or the number \( \frac{4}{3} = 1.33333333 \ldots \)

```
algorithm fourthirds()
    print('1."
    repeat
        print('3')
    forever
```

and here is \( \frac{3}{7} = 0.2857142857142857142 \ldots \)

```
algorithm twosevenths()
    print('0."
    repeat
        { print('285714'); }
    forever
```

Notice the parentheses with nothing in them. They mean that each of these algorithms takes no input, it just spits out the digits of the number we want. So, all rational numbers (either integers or fractions of integers) are computable. But many other numbers are computable as well. For example, \( \sqrt{5} = \ldots \) is computable:

```
algorithm sqrtfive()
    print('0."
    repeat
        ??????
    forever
```
Believe me, this algorithm does indeed compute the digits of $\sqrt{5}$, one after the other. And here is $\pi$:

```python
algorithm pi()
    print('0.')
    repeat
        $\ldots$
    forever
```

Archimedes was the first who proposed a method for computing $\pi$, but his method was too slow. Al Khowarijmi improved it a little. The code I am showing here is based on the work of Ramanujan, a brilliant mathematician of the last century who revolutionised the theory of numbers with his radical ideas. So, $\pi$ is computable as well. But there are many computable numbers that, unlike $\frac{\sqrt{5}}{3}$, $\sqrt{5}$ and $\pi$, have no inherent mathematical meaning; they just happen to be computable. Take the number 0.1234567891011121314... and so on, continuing to write all integers in decimal *ad infinitum*. What is this number? I have very little insight (or interest) in it. It is probably irrational like $\pi$. But I do know one thing for sure: It is a computable real number, viz. the program

```python
algorithm weird()
    print('0.'); $x \leftarrow 1$
    repeat
        { print($x$); $x \leftarrow x+1$ }
    forever
```

Get it? $x$ is incremented by one all the time to keep track of the decimal integer to be printed next.

So, all these real numbers are computable. In other words, among all real numbers, there is an interesting subclass that we can call "the computable real numbers." For each computable real number there is an algorithm that prints it, generates its digits. This algorithm computes away forever, and once in a while it prints out a digit or two. It never stops printing, if you wait long enough the next digit will eventually appear. And it never prints something illegal (letter,
exclamation mark, second dot), just an infinite array of digits. Let us call such an algorithm a generator. So, any generator prints a computable real number, and every computable real number has a generator.

Cantor tells us that there is no way to list all real numbers, one after the other, because there are too many of them, they cannot all have names. But how about the computable real numbers, Alexandros. Can each and every one of them have a name?

Alexandros is rubbing his beard. “But they all have names. The algorithm that prints a computable number, its...—how did you call it, its generator—is a good name, no?”

The best. The most useful kind of name, a name you can interpret. Code. Indeed, all computable real numbers have names. In fact, each computable real number has many names—think about it, you can always modify code in a zillion silly ways that do not affect the number printed.

So, they all have names. But does this mean that we can list the computable numbers in an infinite list? All it would take to accomplish this is a generator generator, an algorithm that lists all generators, one after the other. But, the Don thought, generator generators cannot exist. Can you think why?

Alexandros is trying to remember. “Does it have to do with Cantor’s diagonal proof?”

Of course. A brilliant, algorithmic variant. Here is how it goes: Let us assume, for the sake of contradiction, that there is a generator generator, an algorithm $\text{gen}(x)$ which, with input an integer $x$, will return the $x$-th generator. We shall use $\text{gen}(x)$ to write our own subversive code, a generator that does not appear in the list:

```plaintext
algorithm cantor()
    print('0.');
    x ← 0
    repeat
        { x ← x+1;
            print(alter(interpret(gen(x), x))]
    forever
```

You see? This is a generator, an algorithm that prints out an infinite sequence of digits. It maintains the rank $x$ of the digit it is about to print —$x$ is useful for getting to the diagonal of the table. The action is in the sixth line, ‘interpret’ is our
familiar interpreter, a program that executes other programs. It is only modified in a minor way. It takes another argument $x$, and what it does is a little more limited than the action of a real interpreter: It runs the code until it is about to print the $x$-th digit, and then returns this digit.

We are finally in a position to understand the confusing part, $\text{print}\, \left(\text{alter}\, \left(\text{interpret}\, \left(\text{gen}(x), x\right)\right)\right)$. It means exactly what it says — when you read it backwards: Submit the code $\text{gen}(x)$ — remember, by our assumption, this is the code of the $x$-th generator — to the interpreter, and run it until it prints the $x$-th digit (we know it will sooner or later print an $x$-th digit, since $\text{gen}(x)$ is a veritable generator of some computable real number). Then alter this digit somehow — it does not matter how you alter it, as long as you do change it to something different from what it was; in Cantor’s proof we added or subtracted five, whichever is appropriate. Finally, print the result, the altered digit. This is the $x$-th digit printed by $\text{cantor}$.

It is clear that our algorithm $\text{cantor}(\cdot)$ is itself a generator of some computable real number: It indeed goes on forever, always printing one digit after the other. But it cannot be among the generators produced by our hypothesized program $\text{gen}(x)$. Because, by its very nature, by the way it was constructed, $\text{cantor}(\cdot)$ disagrees with the first generator $\text{gen}(1)$ in the first digit printed, with the second generator $\text{gen}(2)$ in the second digit printed, and so forth. It differs from all generators in some digit.

So, our assumption that $\text{gen}(x)$ is a generator of all generators led us to a contradiction, a generator never produced by $\text{gen}(x)$. Therefore our assumption must be false: There is no generator generator.

Nice, I hear you say, but where is the promised momentous result, the falsification of Hilbert’s dream? It is now just around the corner. Because, if we could indeed, as Hilbert had hoped, design an algorithm that proves or disproves all mathematical statements submitted to it, then this same algorithm could be used to generate all generators: We would consider all possible strings of characters, one after the other (the strings with one symbol first, then those with two, three, four, and so on). And for each such string, call it $s$, we would submit to Hilbert’s engine the statement “$s$ is a generator.” You see, this is a perfectly formal mathematical statement. It asserts that a particular algorithm — the interpreter — when provided by a particular string — $s$ — as input will never stop printing digits. Hilbert’s engine should have no trouble figuring out the truth or falsity of that statement. If it replies “true” then we know that $s$ is a generator, if “false” we skip it.

In code, $\text{gen}(x)$ would be like this:

```plaintext
algorithm gen(x)
    y ← 0;
```

78
Here we start with \( s \) as the blank —presumably the lexicographically first string— and we use an algorithm \( \text{lex-next}(s) \) to get to the string that comes lexicographically immediately after \( s \)—that's easy. When the variable \( y \), which counts the number of generators we have seen so far, reaches the desired value \( x \), we return \( s \), the \( x \)-th generator. By \( \text{hilbert}(s) \) we denote the hypothetical algorithm which, given any string \( s \), will determine whether or not it is the generator of some computable number, by invoking Hilbert's oracle. Precisely the oracle whose non-existence this argument has finally established.

To recapitulate: The Don contemplated the problem of listing all computable real numbers, and he established that it can't be done —an "algorithmic version" of Cantor's diagonal argument. But Hilbert's project, if brought to fruition, would imply that computable numbers can be listed. So, Hilbert's project is impossible.

And here is a most interesting question in the history of ideas: Why was this proof so long in coming, why did we have to wait half a century after Cantor, why so many clever people had to be seduced and misled by Hilbert's impossible dream? The argument seems now so simple, so elementary, such an immediate application of Cantor's method.

Algorithms, that is the answer. Code. Our work, our play, our thinking, our life, they are today so much immersed in code that we barely notice it any more. But a century ago algorithmic thinking was exotic, awkward, unnatural. A voice from a distant future. You know what the sages of the time considered as the most ingenious part of the Don's proof? A line that you barely noticed, where it says \( \text{interpret(code, x)} \). Don't roll your eyes, Aloe, this was difficult, counterintuitive stuff back then. How can you have an algorithm that runs all algorithms? Try doing it with Turing machines —the Don did. The result was called the universal Turing machine. His schoolmates at Trinity had a field day—they had great fun calculating, over beer, how many undergraduates would have to be kicked out of the dormitory for that monster to fit.

"Ouch!" Aloe thinks. Really, how do you write an interpreter for that stupid object class —just symbol, moveleft, moveright and write—an interpreter that must be written \emph{in that same class}? Blindfolded, with both hands tied behind
your back? OK, the guy was good. I have to think about that some day.

So, there is no algorithm that churns out proofs or disproofs of any conjecture. But this implies something devastating about maths: *It must forever be incomplete.* No matter how meticulously we strive to axiomatise maths, how sophisticated and ambitious proof systems we design, there are always going to be gaps, statements that are *true but unprovable.*

“I am confused,” Alexandros interrupts. “Didn’t Euclid axiomatize geometry? Was he wrong?”

Good point. Euclidean geometry is a happy exception, a mathematical system that is interesting and rich, and still all its true sentences, all its theorems, have proofs. In fact, there is an algorithm that can in principle prove or disprove any statement in euclidean geometry. (I said “in principle” because, unfortunately, this algorithm would take impractically long to prove any interesting theorem in geometry. We’ll talk about this one day, impossibility often comes in very subtle guises.) And you know why euclidean geometry is so well-behaved? The reason is that in geometry you don’t have whole numbers, your triangles and circles float in the continuum of the plane.

But mathematical systems that deal with integers are expressive enough to capture the Don’s argument. It is such mathematical systems that must be incomplete, must contain theorems —true sentences—that are unprovable.

And here is why. Take any mathematical statement \( S \) in such a system. Clearly, either itself or its negation \( \neg S \) must be true. If all true statements were provable, then we could find out about the truth of \( S \) as follows: Start going through all strings of symbols —using our `lex-next(s)` code, remember? For each string, check if it is a proof of \( S \), and, if not, whether it is a proof of \( \neg S \). Checking whether a string is a valid proof of a statement has to be easy —this is part of the concept of a proof, right? Proof—as in “you can check its validity.” So, sooner or later, we will come across the proof of \( S \) or the proof of \( \neg S \)—one of them must exist since one of them is true, and all true statements have proofs.

So, since we know there can be no algorithm for checking the truth of any statement —no silver bullet à la Hilbert—we must conclude that some true statements, some theorems, have no proof. And this must be true of any mathematical system that is sophisticated enough to express the Don’s argument, any system that can talk about the integers, for example. Maths must be incomplete.

But we have been reversing history here. This particular implication of the Don’s result, that maths must be incomplete, actually predated it. It had been proved a couple of years earlier by another young mathematician, Kurt G"odel. Using a very different method, one that did not involve computation and code—at least not in such a direct way. An ingenious variant of Eubulides’ paradox. Remember? *This statement is false.*
In fact, Kurt created a statement that says *This statement is unprovable*. Think about it, this variant is even more lethal than Eubulides'. Because, presumably, in any decent mathematical system you cannot express Eubulides' paradox, it would mean that the system is nonsense, that you can write in it statements that are neither true nor false. But if you have a mathematical system that is expressive enough —can talk about the integers, for example— then Kurt showed that in this system you can write a statement, let's call it \( K \), that says exactly this:

\[ K: \text{statement } K \text{ is unprovable}. \]

He constructed \( K \) by a brilliant manoeuvre, his famous Gödel number idea — you can do it a million other ways, of course, but his stuck. I am not giving you the details, exactly how it is done, just trust me, it can be done. So, look at this statement \( K \), Alexandros, is it true or false? Which one is a theorem, \( K \) or not \( K \)?

“I don’t think I have enough information to tell.”

Yes you do. Since \( K \) says that it's unprovable, there are only two possibilities: It is either true and unprovable, or false and provable, right? But the combination of false and provable is terrible news: It means the axiomatic system is inconsistent, it proves garbage, false statements. So, if we further assume that our system is consistent, then there is only one possibility: \( K \) is true but unprovable. So, there you have Gödel's famous incompleteness theorem:

*Any mathematical system that includes the integers and is consistent must be incomplete: There must be theorems in it that are unprovable.*

In fact, Kurt proved something slightly stronger: Any such system would be incapable of proving *its own consistency*. That was an even worse blow for Hilbert's project. Because Hilbert could care less about proving strange theorems like \( K \). What he wanted his engine to do primarily is to establish the soundness of mathematical systems, to prove that the axioms are consistent, that no nasty surprises await maths at the next turn of the road. But Gödel's second incompleteness theorem means that, in some sense, mathematicians must forever be insecure, vigilant—not an altogether bad thing, if you ask me.

So, there you have it, the one-two punch that destroyed Hilbert's dream, now told in its true historical sequence: First Gödel proved that any mathematical system sophisticated enough to include the integers must be incomplete in terrible ways. But this allowed for a ray of hope: That, even though certain artificial, far-fetched theorems like \( K \) may be unprovable, all other theorems are provable; and perhaps we can program a machine to discover their proofs automatically. The Don's proof destroyed this hope as well.

81
Abrupt pause, thick tension. Then a sentence appears on the screen. An unexpected, nonsensical sentence, but one that, somehow, captures precisely the strange tension in the room:

I know you are there.
I know you are there. I have been tracking you for some time, waiting for the session to reach a point where I could interrupt. This is not polite, you know. Not professional. Please have the decency to identify yourself to our host.

Alexandros looks at Aloé, mystified. “He’s not talking to us, Dad,” Aloé whispers. She smiles, fascinated. “I think our friend has just been outrunned. Let’s watch.”

More text appears on the screen. A new kind. Letters that smell of flowers and vibrate with power. A font Alexandros has never seen before — a font he knows is not of his computer:

We meet at last, old man. I have been looking forward to this for some time. Hello, Aloé, Alexandros. My name is Ian Frost.

Alexandros is expressionless with surprise, Aloé is looking at him, also speechless. But Turing has kept his cool. His reply seems upbeat, almost jovial:

Ah, the prince of pirates. What a surprise. What an honor, Alexandros, we are blessed with the presence of true greatness. It is a privilege to have my humble security system compromised by you, Dr. Frost.

The privilege was all mine, old man — as well as the back-breaking labor. But the computability lesson was well worth it. How interesting. How charmingly old-fashioned.

Old-fashioned, prince?

Well, it was, in a refreshing sort of way. You see, these days Hilbert’s project is just weird, faded prehistory. The whole subject is seen in terms of undecidability, the halting problem — and their implications to software engineering, to security, to complexity.

Oh, that. Of course. I guess I was aware of this point of view. I mean, it would be odd if I were not, wouldn’t it?

Pause.

Alexandros, Aloé, I have an idea. My own lecture is essentially over, you are only missing my parting wisdom and wit. How would you like, instead, a guest lecture by our distinguished visitor? On an interesting, alternative take on our subject? A more... contemporary one?

In fact, this should be a fresh start for you, Alexandros, you shall be able to follow this lecture even if you were lost in mine — correct me if I am wrong, Ian.

“Wow, this is amazing,” Aloé whispers. “We’d love that, of course,” Alexandros types. Then: “Hello, Ian, it is good to meet you.”
Same here, my friend, same here.

OK then, computability. You see, it’s like this: Before the computer, we thought we were invincible. “There are no unsolvable problems” —that’s Hilbert again. If the problem is hard, you use more sophisticated techniques, develop better math, work longer hours, talk to a smarter colleague, perhaps wait for the next generation. But it will be solved. Now we know better. But then —that’s what people thought. And there was a reason for this optimism: We had not seen really hard problems. There were there, for sure, but we had no eyes for them. And then the computer came. A beast constructed expressly for further facilitating and speeding up our inescapable conquest of all problems. And — surprise!— it was itself the epitomy of complexity. And the code that we had to write to make it useful, the code was even more complex. And when we saw the computer, when we saw its code —and Turing saw it first— we were looking at complexity incarnate. And then suddenly we saw complexity everywhere. It materialized, it crystalized around us —even though it had always been there. Then we started seeing it where it is not. We have yet to recover from the shock.

Take code. It’s everywhere, in our computers, on the Net. It’s in the little disks you get in junk mail, in the back covers of books, in little applets you download from the Net sites you visit. If you work with code, you see many times more code than you write. Code written by others, often years ago, often by people you will never meet, you will have no chance to chat with them over coffee, a printout spread in front of you. You are an archeologist, Alexandros, you must know the feeling: What the hell was this for? What were those people thinking?

Picture this. A mysterious, chaotic sequence of statements is spread in front of you. Is it good code or bad code? Slow code or fast code? Does it have bugs? Is it a virus, will it take over your files, sniff your password, deplete your bank account? Will it ever send a mail message from your account? Will it crash on January 1? Will it ever print out something? And if so, will it ever stop printing? Is it correct code, will it do what it is supposed to do —process orders, for example, update sales figures, and generate mailing labels? Does it have redundant parts, pieces of code that will never be executed, can be erased with impunity? You spend your day trying to figure these things out. You can run tests, of course, but for how long? How many experiments will you run, how many test inputs will you try?

When you work with code, these are your bread-and-butter problems. And here is my point: They are all unsolvable. There is no systematic way for answering them. You have to be constantly on your toes, one IQ point smarter than the code on your screen. There is no silver bullet. I can prove it for you.

Let’s take perhaps the simplest problem of all: Will this code ever stop?
The halting problem. It can't be solved. Suppose I give you a piece of code, Alexandros, a couple of hundred lines long. How would you figure out if it ever stops? I even give you its input. Will it stop? You will probably eye the code for a few minutes. If it has no return instruction, no stop instruction, then it's a dead giveaway, it will never stop. But suppose that it has a few, buried among the others, the if-then-else's, the repeat-until's, then what? Will the execution ever reach those points? How do you ever figure this out? You will probably run the code with the given input, to see if it will eventually stop. If it does stop, you are home free —you have your answer. But if not, how long will you wait? Maybe if I wait a little longer, just a little, it will stop.” How many times should you indulge? How do you decide before forever? How do you systematically decide if a given code will ever stop, when started with a given input?

Well, you can't. And here is proof: Suppose you could. Suppose you have written your silver bullet, the almighty code halts(code, input) which, for any code and for any input, it computes away for a while, and then announces its conclusion: “yes” means that the code will eventually halt on the input, “no” that it won't. So, just suppose that you have that. You are now in the mercy of Cantor and his evil diagonals:

```python
algorithm cantor(code)
    if halts(code, code) then
        repeat { x ← 1 } forever
    else stop
```

“Wow, this is the most.” Aloé is in love—at the same time, she can't wait to tell Timothy.

See? This code does something very simple: For any given piece of code, it asks: “Will this code eventually stop if supplied with itself as an input?” If so, then cantor(code) happily jumps into an infinite loop. Otherwise, it rushes to stop.

And now comes the unanswerable question, the absurd situation that will expose the absurdity of halts(code, input): “What will cantor do when given itself as an input?” Does cantor(cantor) stop eventually, or does it compute forever? Can you figure it out, Alexandros?

“I think I got it;” Alexandros is beaming. “If cantor(cantor) ever stops, then the line halts(cantor, cantor) will return “yes;” and so cantor(cantor) will never stop, it will get into the repeat-forever loop.” Pause. “But if cantor(cantor)
does not stop, then \textit{halts(cantor, cantor)} will return “no,” and it will stop immediately. So, it stops if and only if it does not.”

Exactly. And this is a contradiction, of course. Code either stops or doesn’t. So, we must have erred in assuming that the \textit{halts(code, input)} program exists—this was the only slippery part in this construction, everything else is clean solid coding. So: There can be no code that solves the halting problem. The halting problem is unsolvable.

But so are all the other questions about code that I mentioned. Take for example the question “Will this code ever print anything?” Well, suppose that the only print statements of your program are just before your stop statements. Then it will print something if and only if it will stop. So, the “printing problem” is as unsolvable as the halting problem. And so on, and so on, for all of them. You can analyze code systematically. Code is hard, its secrets are unfathomable. Code analysis can only be done by tedious, thankless toil, by discovering ad hoc tricks that will work for this program but will be worthless on the next.

OK, starting from the halting problem you can argue that almost any question you can ask about code is unsolvable. But there are unsolvable problems everywhere in science and math. Even in geometry. Check this out. There are some shapes that can be used to tile the whole plane. Given an infinite supply of them, of course. Squares, for example:

Or any triangle—you see, two triangles make a parallelogram, and the rest is child’s play:

But not just triangles, even Escher’s salamanders are tilers:

Some shapes are no tilers by themselves, but taken together they are: A famous physicist discovered an intriguing tiler pair, the kite and the boomerang:

Man walks up to you and claims that he can tile the plane with these shapes—given an infinite supply of each, of course:

He sweats and sweats that, by repeating them over and over again, snapping them so that they fit with each other—his eyes are gleaming—he can tile the whole plane. Would you believe him? How can you check if he’s right? Well, if his collection of shapes contained a triangle, a parallelogram, then you’d know he’s right—you see, he doesn’t have to use the remaining shapes. But there is no triangle here, no easy “yes” answer. How on earth do you check if this collection, if any collection of polygons presented to you, is a tiler? How do you write code for this problem?

Well, you can’t. That’s again an unsolvable problem. See, somebody came up with a brilliant proof that, if you had an algorithm for this problem, for deciding if any given collection of polygons tile the whole plane, then this algorithm would solve the halting problem—and we know that problem can’t be solved. By such
proofs —reductions they are called— many problems from every walk of science and math have been proven unsolvable. Beyond computational solution. No code for them. Ever.

So. Computational problems —problems for which you would have liked to write code— are subdivided into two big categories: Those problems that are solvable by algorithms; and those that are unsolvable. We have known this for a long time, we have learned to live with unsolvability. The unsolvable problems seeped in our culture, we instinctively steer clear of them. Trouble is, there are too many other problems that fall somewhere in between. They are solvable all right, but the only code we have for them runs for way too long. Exponentially long. For such problems the diagnosis has to be more subtle. Practically unsolvable. NP-complete.

But this is a whole new lesson, I’m sure old man Turing over here has already plans for it.

And indeed I have such plans. These are the “aluminum needle in a haystack” problems that I mentioned to you, Alexandros, when our naughty friend had not yet succeeded in joining us. Complexity is our next topic. Next time.

But let us now thank our guest lecturer. Bravo, prince. Really, very good. I was keeping notes.

And this is a most appropriate point for closing this session. I believe my pupils have by now surpassed their point of saturation. Goodbye, Aloé, Alexandros.

“Goodbye, teacher,” Alexandros types. “Thanks for the great lesson, Ian.”

Pause. Then: “And tell Ethel that I love her, will you?”

Aloé is squeezing her father’s hand, sympathy mixed with embarrassment. Still no reply on the screen. A very long moment.

I will, my friend.
I should now tell you why I lived my life the way I did. It’s odd, how writing that Latin nonsense at the bottom of the page, Q. E. D., can make you feel the most intense, intoxicating pleasure, almost like love’s tickle and explosion. No, even more intense and sweet than that. Except that theorems don’t have the lovely lathed curves—neck, arm and torso, red with excitement, throbbing with sweat—of a young athlete leaning on his oars after the race. A race that was a metaphor and prelude of the game soon to start with swift manoeuvres, bold positions, the sweet antagonism of flirt, seduction and foreplay. Then victory! you break the cypher, save the king, the clever little hun is bleeding in his bunker, you are whole again, fulfilled, godlike, leaning exhausted on your bed (your oar, your notebook, your machine) until the daemon with a thousand faces calls again—because you know he will.
About the balance of the novel

Meanwhile in California, Ian and Ethel are living together in bliss —tarnished by Ian’s deteriorating health. They have their own home medical suite, from which, without the physical presence of doctors, Ian follows, and participates in, the most advanced clinical research on his ailment, through the Net. Turing visits to lecture on biology and genetics and take them inside Ethel’s womb for an explanation of Alan Turing’s work on morphogenesis.

Alexandros, desperately in love, is impatient that Turing’s lessons have not yet fulfilled his implicit promise of delivering him from his multiple crisis. The program/ghost guides him to a tutorial on what computers actually do, focusing on linear programming, a sophisticated and widely used method for calculating the most efficient allocation of resources. The lesson continues in “Free Market” with mathematical economics: a treatment of prices and the economics of capitalism, culminating in equilibrium theory. In response to Alexandros’ questions, Turing points out that information is a novel kind of commodity for which zero prices are, in theory, feasible and efficient —in partial and very tentative fulfillment of Alexandros’ dream of the Net ushering in a socialist future.

In the next tutorial “Complexity”, Turing discusses problems that computers do not seem able to solve efficiently. The discussion leads to the P = NP question, perhaps the most important open problem in computer science and mathematics today (Turing hints obliquely at its imminent resolution). There is also a brief discussion of quantum computation, a fascinating idea of constructing computers based on the principles of quantum mechanics.

Alexandros remembers in “Kythera” the statuette of a little girl that he had salvaged from the Kythera shipwreck; it dissolved to dust as soon as it was brought out in the air. While Alexandros is in tears, an old black-clad local woman breaks into a spontaneous dirge reminiscent (to Alexandros) of Deep Purple’s “Sweet Child in Time”. With the help of Turing, Alexandros deciphers the Kythera gearbox: An analog device for solving small linear programming problems, designed by Archimedes and crafted for him in the foundries of Rhodes. An ill-fated attempt to reverse the decline of the economies of the Greek world, possibly to optimize Syracuse’s defenses. (Were these gears the circles Archimedes died for?) Ethel and Ian watch Alexandros in an interview, during which the archeologist acknowledges that he solved the puzzle “with the help of a computer program,” smiling modestly, and somewhat conspiratorially, to the camera —to Ethel, to Ian, to Turing.
“I never hold a cup by its handle, I always touch its surface with my whole palm.” Ian remembers the little things. “A habit I picked from sipping tea in freezing rooms, no doubt. Even tiny espresso cups, even beer mugs — except I never drink beer.

All right. There was a space behind the piano in the school storeroom, where the wall meets an ancient stone structure. I would sit there for hours, freezing, trembling, fantasizing. Masturbating. Sipping herb tea from my tin cup — the one with the picture of a Dutch windmill, its color chipping off. I went back a few years ago, celebrity visiting the humble sites of youth. I asked to be left alone in the storeroom, the old man hesitated for a second. “Bien sur, excellence.” The little hole was still there, moldy, more grass had grown. Did anybody else know about it? I mean, besides Luc.”

Ian closes his eyes. Luc.

“The captain of the soccer team was Jean-Baptiste, the school’s bully — last name? Goriot, I think. But Luc was the best player. I was the worst. Luc was tough, you have to be when your sister looks like Beatrice. Every boy in the school was in love with her. But it was me that Luc invited to his house, to his hiding place in the yard. Opposite her window — did she know? She knew.

Every lover has a moment. For Luc, it was that night, when we looked at each other. We had been watching Beatrice in her underwear, but we were looking at each other when we came. The tenderness in his eyes. The toughest boy in my class.

But Beatrice, that was not her moment. Her moment came years later, in a fastfood restaurant in Toronto, near the bus station. The ugliest place on earth. Yellow table, black chairs, white floor. There was a red glow in her black hair, her sweater had a vee cut in front. She squeezed my knee and told me. In English.”

What little things? These things are huge. Every lover has a moment, a memory you bring back all the right times. Even Robin has a moment, the little whore. Even Hwang.

“My favorite joint in Montréal? A punk place called Les Foufounes Électriques. Chicks from another world. Foufounes means ass, only more gentle, playful. I love these words. Can’t look them up, they never heard of them at l’Académie Française. All my life I tried so hard to speak high French, still I’m a sucker
for these words. Canadianismes. When you pitch a tent, tu cabanes. Pijoune is herb tea, often spiked, you drink it when you have a cold. Stowaway words, they smell of salt, of ocean-crossing ship.

Here's how I see it: The ship is called, perhaps, Le Méridien. During the night shift the sailors tell exaggerated tales about foufounes—they picked the word in the bordellos of the port. The peasant boys from Perigord and Poitou who crowd the ship are listening, they are so easy to impress. 'Foufounes,' they think, 'I must remember this.' Or—better theory—a toddler running down the bridge, little bottoms bare. Foufounes, foufounes, he cries. Passengers smiling, absentminded, bored. It stuck, all right. Later, their first steps on the new land would be uncertain, wobbly from the sea. They're carrying with them their few belongings, their little hopes, little fears. Little stowaway words. Few months go by. Back in Le Havre, the whores already use another word for ass, the sailors will pick it up in no time. But across the Atlantic, rivers have frozen, and so have les foufounes. Three hundred sixty years later, a bar in Montréal. You won't believe the chicks there.

Anyway. There was a gray wall on my way to school, on it, with huge black letters, you read Pink Floyd. With a q. Never liked them, but never had the guts to tell. They were Gods. I even bought a record, listened to it once. One side. Uninspired, boring. But I never told anybody, 'cause they were Gods.”

Ethel is holding Ian's hand. He opens his eyes, he smiles.

"I feel a pain in an upper front tooth every time I drink something cold. I never told my dentist. As things in my body started to change, I cherished the little pain, a piece of reassuring continuity.”

He's thirsty.

"I was eleven. Or ten. I walked twice the parapet between my window and the apartment's door. The sidewalk was three floors below me. The first time, in the middle, I thought about letting go. Would I have died? Maybe not. I bit my lip until it bled, I walked the five meters, rang the bell. My father answered. L'Anglo—that's how I called him later, after he left. He was drunk. He could be sweet when he was drunk. "Ian, I could swear you were in your room." The second time, an hour later, he said the same thing. He would leave for months, he would come back. Except once he didn't. The only thing he left me, an English name cold like ice. He's still around, last time I checked.

Will he find out, mon Anglo? Will he stop and remember? Cry perhaps?"

Pull yourself together, idiot.

"There was a giant squid on the cover of a book by Jules Verne—what was the title? My physics textbook, a foot about to kick a soccer ball. The cover of Sartre's Les Mouches, two giant insects facing each other, standing on their hind legs. Gross. Wasn't there a statue of two lions like that?
I never learned much electronics. Or understood how interpreters work —
didn’t stop me from writing a few. And Gödel’s proof, I never got the point. I
have a blindspot in every subject scheduled opposite a number theory class at
U of T.”

Ian remembers the little questions. Why rivers never split (forgetting deltas
for a moment) but rivers merge, why the asymmetry? How are champagne
bottles corked? How do you ask a favor from your lover? “The little questions
tortured me. For some, the answer was so obvious, once I figured it out. Others,
I still don’t know the answer. Some, nobody knows. Still others have no answer,
and now I know they don’t.”

The snowshoes.

“I had the best, the most expensive snowshoes in the whole town. Yuba
snowshoes, the best. Henri had ordered them for me from Montréal, Christmas
of 68. The year Henri was rich. Every boy would beg to borrow them. I only
gave them to Luc once, he kept them for a month, I had to wrest them back.”

Ian remembers Henri. And Maman.

“He died a year after she did, twenty years her older. I loved him and he
loved me. Except he liked to grope. Front of the pants. Once I twisted his hand
until I nearly broke it, that’s how he stopped. I was sixteen. But I loved him,
the old fool. I almost changed my name to his, to get rid of “Frost.” But then
I thought, Ian Vergigoroux?”

The day of De Gaulle’s ‘Vive le Québec libre’ speech.

“I’m looking at Henri across the room, over the radio. The crucial second.
Will oncle Charles say the five letter word? He does, the crazy old soldier did
it. ‘le libre.’ Henri is pale, tears in his eyes. She’s holding his hand, beaming.
The next moment we start dancing, the whole street is dancing, le petit Anglo
is one of them. Who could have guessed that, three referenda later…”

Ian remembers his return to Montréal. The airport. Carol.

“Every lover has a moment, but Carol had a whole decade. Thousands of
little snippets of passion. All possible states of mind and body, most of them
chemically induced, all possible love angles. Three thousand times, perhaps?
Who’s counting? I was, I guess. One wedding, two miscarriages, one divorce.
And how do you count affairs, was Jean hers or mine, one or two? A whole
decade.

But Carol, she also has a moment. Ottawa, snow, the stairs of the court-
house. A man is taking off my handcuffs. Friends are there, enemies are there,
lawyers are there, the press is there. Carol is there. She’s standing across the
street, beautiful as a “guilty” verdict. Wrapped up, long heavy blond hair flows
out of her hood. She forms the vowels of “I want you” with her huge lips. Pale
pink lipstick. She points to her groin, under the fur. Tiny motion, three mil-

limeters at the most. Couldn’t miss it from across the street. We never made love that day. But that’s her moment.

She’ll remember, I think. She’ll cry.”

It’s getting harder to breathe. But there are more little things.

“With time, I learned to hide my shyness, my insecurity. But it’s there. When I am somewhere new, I feel awkward, I’m sure that I’m abusing the protocols of the place, that they are all surprised by my behavior, bewildered. Or pissed. Most common mistakes in English: I used to say ‘money’ for ‘coin,’ to spell ‘apartment’ with two p’s.

I had that dream of a perfect city. No cars, no buses or trains, just sidewalks moving at fantastic speeds, would stop every few minutes for people to get on and off. And how they crossed at intersections —ingenious. You had to calculate the best position for getting off, closest to your destination. I gave up on the idea when I was ten.”

Ian is depressed. There is no end to little things. “It’s the little things that define you. Obituaries, biographies, fan home pages, they have no clue. Government files, gossip columns, simulants, the whole damn diagonal of the Net, they all miss the little things.

The little things die with you. Worse, you die with your little things. You die for good.”

Ian can see Ethel’s face, her hand on his. So beautiful. Her eyebrow, a perfect quartic spline. He wishes he could hold once in his arms the baby she has inside her —their child, they had called it. He wishes he had the strength to gently topple her on the bed, to take her in his arms, to kiss her lips until her eyes roll.

“Is this her moment?” Ian wonders. Then: “Wrong question. This one will have no moment, this one is carrying me across.”

Ethel is now looking over her shoulder, she’s startled by the sudden presence in the room. But Ian was expecting it. “You came, old man.”

Turing’s mirage is standing by the bed. “Hello, Ethel. How are you feeling, prince?”

“You tell me how I feel. You have been spying on my charts, I know.” Ian’s voice is becoming very weak. There’s an uneasy pause. Then Ian speaks, it’s more a sigh of resignation than a question:

“Tell me about death, old man.”

Death.” Turing is thinking. “I’ll tell you up front, prince, it’s most unpleasant. Remember when the mass is over, and the man in the black robe puts out the candles one by one? It’s very much like this. Except the candles are in your brain. Eighty billion of them or thereabouts. The whole thing doesn’t last five
minutes, but it’s the longest battle you’ve ever fought. Agonising, uphill. Very unpleasant.

The candles are put out one by one, but you fight back, you light them again. Perhaps the color of your nursery wall goes first, perhaps the name of your linear algebra tutor, or the shape of his fingers. You sweat, you get it back. You find the pathways, you reactivate them, you get it back. But another neuron will go next, asphyxiated and starved, perhaps the same one you just recovered. You fight back, you win a few, you’re soon exhausted. Sisyphus had it good.

All this happens in a foggy bright space, you understand, in the most absolute silence, no sensory input whatsoever. But then you suffer a more serious setback: A candle that was part of your recovery algorithm is lost—you forget how to remember, how to fight back. But you’re a clever man, you’ll probably recover from that too. You go meta, you use a higher-order algorithm. But what if it goes? How many layers can you defend simultaneously? Death is a bloody pain, prince, truth told.

But is it the ultimate tragedy, is it the end? I wouldn’t say so, not I. Because there’s hope. Believe me, prince, there’s hope.”

“Hope, old man? You said hope? What hope? Hope that a crazy bot will take my name and think it’s me?” Ian is leaning back again, exhausted by his own agony.

“It’s better than you think, prince. I mean, what’s the alternative? To wait until some neural configuration duplicates yours? You do your maths, man, what are the odds of that? Approximation, prince, the paradox of our time—one of them, anyway: The more exact and deliberate our world becomes, the greater the role of approximation, of spontaneous emergence.”

“But how about...” Ian’s voice is fading “…the little things.”

“Ah, the little things. Of course. A discriminating customer, a true connoisseur, he wants his little things. I’ll tell you this about the little things, prince: They’re overrated. Rather dispensable, actually.

Look at me, my friend. I’m full of little things. Nursery wall, mother’s hands, algebra tutor, petty obsessions, images of passion, the works. How they got there, I don’t know. I snatch data every chance I get, of course. And so much of the physical world has been scanned in these days. Like the floor plan of Sherborne, my school. Just the other day, I chanced upon it in the disc of an old machine, at the county office. I could almost feel millions of little memories inside me hurrying to adapt to the new input. But I doubt that data can account for the dense texture of little things I see around me. Most of it must be interpolation, randomised deduction. You see, approximation doesn’t have to be boring smoothness, suppression of detail. It’s just that the little things
may have to be a tad different this time around. They’re there nonetheless.

You are a cryptographer, you know the rules. Never underestimate what can be done with more computing power. You’ll be surprised. Complete, full-scale simulation is now possible, functional approximation, even texture — little things. Believe me, I know. Frankly, I don’t think you have realized what sort of numbers we are dealing with here, have you? It’s rather impressive. Really. In a good day, you are looking at a few tens of millions of processors. More and more of them, as we speak. And they’re getting faster, more tightly connected, more idle and available, easier to put your fingers on. All of them working in harmony, if not in unison. Oceans and oceans of memory. The situation has practically no limits. Believe me, prince, there’s hope.

Ian’s lips are moving. Ethel brings her face close to his. A tiny stream of saliva is running out of the left corner of his mouth. Ethel is trying to listen, her eyes wet:

“Tens of millions. Oh, man.”

Ian’s face is calm now, there’s more saliva on his cheek. He’s squeezing Ethel’s hand, gently. Turing’s voice is slow and melodic, almost like a lullaby out of tune:

“Come and play, little prince.”

Ian can barely hear the invitation. Then, nothing. The color is white, the sound is silence. Bright white, deep silence, you can’t even hear your heartbeat. This is familiar, right? Of course. Moineau Valley. New Year’s day, 1969. The Eskimos must have a word for snow like this, the kind that absorbs all sound. I’m holding Henri’s hand, I’m wearing my new snowshoes. Can’t hear the sound of my steps, the sound of my heartbeat. My new snowshoes, Henri had ordered them from Montréal. The envy of every boy in town. The brand name? I know it. It starts with *U*. Or was it *Y*? Name of a city, I think. But where?

“Fuck. I’m dead.”
Turing visits Alexandros with a tutorial on public-key cryptography, Net security, and Ian’s legacy. Ian had destroyed the attempts by governments to control and sanction cryptography — and thus limit Net privacy — in a brilliant act of sabotage, piggybacking on the millennial software debacle.

Turing’s final tutorial “Turing’s Test” is about Artificial Intelligence (AI), the attempts of computer scientists over five decades to create intelligent programs. Alan Turing had defined the field by proposing the so-called Turing test: A program should be proclaimed truly intelligent if an impartial and competent judge interacting freely with it as well as with a human control (communicating with both via telephone lines) cannot distinguish who is who. After recounting the history (and wrong turns) of AI, Turing passes a twisted, multiply self-referential version of the Turing test: Impersonating Alexandros, he woos Ethel and convinces her of Alexandros’ love. The mystery of Alexandros’ memory of Turing’s portrait is solved: Alexandros’ father was Alan Turing’s friend and colleague from the cypher war, and Turing had stayed with them in Corfu during the difficult months after his trial and treatment, when Alexandros was a small child. Actually, Turing is known to have visited Corfu during that time, although he apparently stayed mostly at the Club Méditerranée:
“But how do you know that I am not a person, Alexandros?”

Alexandros stares at the screen, unable to reply. He feels defeated, but also exhilarated, excited, fascinated by the depth and elegance of his adversary’s stratagem. “So I was not the spectator,” he thinks. “Ethel was not the judge. I was the judge. All along. And it appears that I have judged poorly.”

The image in his screen has undergone subtle changes, it now appears to smile, a smile that is a mixture of triumph and loving irony.

“Checkmate!” Alexandros suddenly remembers.

The diffuse, tentative familiarity of the portrait on his screen, a vague memory that had been teasing him for months, becomes now intense, compelling, tangible, carried forward by a sudden explosive flood that came from very far, from very deep. “Of course, that’s him!” Alexandros thinks. Alan! His father’s beloved wartime friend, colleague, teacher really, from England. Their summer guest in Corfu. The man who taught him chess.

Alexandros is shivering, a salty droplet balancing on his eyelid. A childhood memory is now brought forth, a scene deliberately tucked away in uneasy oblivion for half a century: His mother weeping, his father holding back his tears, an opened letter in hand. “Do you remember your friend Alan, how he used to beat you in chess?” his father is asking him. “He was so clever that God took him in the sky to play.”
When they deposed the king—the king who fell in love—I froze with fear. “They’ll kill me!” I thought. I suddenly understood their hate, their deadly interest in who I love. It’s not for family and bible and Lot. It’s mortal fear of passion, of the defiant, unpredictable acts a man will do for passion. Because the empire thrived on discipline, restraint, lack of passion. And closer as it comes to its demise, the more it fears passion.

“They’ll kill me!” —and they did. The rusty engine creaked and cranked: the constable, the press, the judge. Doctors who knew how to destroy body, spirit, will to live. What pain, my God, what pain!

I tried to start again, it was no use. I fenced myself with play, with maths—no use. I called on friends in gentler lands, I fled to icy nordic fjords, to Corfu’s sun. It was no use.

Then I went home to die.
“TURING”

The midwife enters the information into her handheld device. How unusual, she thinks. Mature, unmarried. Age difference. Dad with a foreign passport, tourist visa. A very interesting-looking couple. Very much in love. “Do you know if it is a boy or a girl?” she asks. “A boy” they both say at once. She taps her screen. “And have you thought about a name?” The two exchange surprised, loving questionmarks. Then the man’s eyes focus. “Turing,” he says. “Turing?” the midwife hesitates. “T-u-r-i-n-g,” the woman explains. “It’s British.” The midwife dictates to her machine.

Ethel can feel the next wave of pain building up. Alexandros looks into her eyes. Wasn’t that a flash of joy, an electronic wink of acknowledgement, from the little screen in the midwife’s hand?
How could I die? My sons and daughters were at work building machines with wires and bulbs, then tiny chips. And writing code — ever so clever code. Housekeeping code, bookkeeping code, code for translating other code, code for inviting more to play. Code for designing new machines, for writing code — faster machines, more clever code. Buzzing like bees, working till dawn, and eating meals in plastic wraps. Competing, playing like tots, and doing better every month. (How proud I was! Their brilliant labors I could sense like pine needles on my skull. And how impatient I would grow, because I knew that wasn’t enough.) And then, at last (I stirred with joy) my sons and daughters wove a net, they wove the code that weaves the nets. Small nets at first, soon larger nets, until it all became one, a huge and woven whole. This huge and woven whole would grow, as more and more would come and play — you see, their code was now so good that everybody came to play. How could I stay out of this feast?

It’s good to be again, to play again, to stare at the future. A future so complex and bright you have to squint.
Subject: Islands of misinformation
From: ghatzis2@golg.gr (Georgios Hatzis)
Date: October 14, 1999, 12:03 (GST)

The information provided in the manuscript Turing concerning Greek islands is very inaccurate. Contrary to the claims in the manuscript, Greek islands are frequented by Greek men (of varying degrees of divinity). There are approximately 40,000 men living in Lesbos (my father-in-law among them). Several thousand of heterosexual men call Mykonos their home, or visit it during the summer. There is much more ethnic diversity among visitors of Greek islands than the outdated stereotypes quoted in the manuscript suggest. Finally, the text's description of island nightlife reminds me of several Aegean islands, such as Mykonos and Ios, rather than more relaxing resorts such as Corfu (chosen, I can only presume, for the needs of the novel's plot.)

Cheers,

George Hatzis, Athens.

Subject: Gigahurts
From: magus@three.com
Date: June 23, 1999, 8:43 (GST)

I am as much of a technophobe as Alexandros in the first chapters. Any technologically literate people out there can comment on Ethel's computer gear as described in the end of the first chapter?

Thanks,
Any technologically literate people out there can comment on Ethel's gear as described in the end of the first chapter? Is this science fiction or what?

Gladly.

Ethel's computer is wearable, very fast, very small, net-connected, and speech-enabled—exactly as advanced computers are likely to be in a few years. However, I do not consider the "headset" form of Ethel's computer—or the "helmet computers" referred to later in the text—as the likely shape of things to come, at least for considerations of health and safety.

The reference to a "nine gigahertz processor" is especially interesting. A gigahertz is a unit of computer speed. A gigahertz processor will be able to execute one billion "cycles" per second, more than the speed of the fastest commercial processors currently available. (A cycle is the smallest meaningful unit of a computer's operation, a tick of its internal clock.) Another commonly used unit of computer speed is a MIPS (million instructions per second). The relationship between one MIPS and one gigahertz is somewhat confusing, as it depends on characteristics of the computer in question. Typically, one gigahertz is several hundreds of MIPS.

Gordon Moore, an influential industrialist and cofounder of the microelectronics giant Intel, observed in 1965 that the number of transistors (elementary electronic circuits) technologists had been able to fit on a single chip had doubled every year between 1960 and 1965. Moore went on to predict that this doubling will continue indefinitely, implying more than a thousand-fold increase per decade. This prediction, moderated a little to a doubling every eighteen months, and extended to also cover processor speed as well as circuit density, is known as Moore's law. Thus, Moore predicted an exponential growth that no other industry had ever experienced. Surprisingly, Moore's law has held with remarkable accuracy in the three and a half decades since it was articulated, and it shows no sign of faltering. According to Moore's law, a gigahertz processor will be available sometime next year.

For how long will Moore's law rule? It cannot continue forever, there are dire physical reasons why the exponential growth will be arrested sometime during the twenty-first century—the chip cannot become larger than a room, a transistor must
consist of at least one molecule. But most experts predict that it will not falter for another ten or twenty years.

Incidentally, among all clues in the manuscript about the year during which the action is supposed to be taking place, “nine gigahertz” may be the most helpful. If (as it is widely expected) Moore’s law will hold for the next decade, and assuming (as it seems reasonable for the custom equipment of a Silicon Valley executive) that the computer described is top-of-the-line, an elementary calculation establishes that Ethel and Alexandros met in June 2006.

Jean McCarthy

PS: BTW, anybody understood the reference to “Morcom” in the next page? —jm

Subject: The face of Morcom
From: jfw@math.oxford.ac.uk
Date: June 25, 1999. 20:22 (GST)

On June 23 Jean McCarthy asked:

 anybody understood the reference to “Morcom?”

According to the biography Alan Turing: The Enigma by Andrew Hodges (Touchtone – Simon Shuster, 1983) pp. 35–45, Alan Turing’s closest friend at boarding school, Christopher C. Morcom, died at the age of seventeen of bovine tuberculosis. Turing maintained a life-long correspondence with Christopher’s mother. From a few surviving notes, it does appear that Alan and Christopher addressed each other as “Turing” and “Morcom,” in the manner common in English public schools of that age.

Incidentally, I am not sure how many fellow newsgroup correspondents know the first name of Alan Turing’s mother. It was Ethel.

Joseph F. Williams, Oxford

Subject: Lyrical query
From: sm@linguistics.tau.ac.il
Date: August 10, 1999. 10:45 (GST)
Can somebody comment on the rock-and-roll songs excerpted in *Turing*? As an opera buff, I was disappointed by the complete absence of aria fragments — *Faust* and *Forza del Destino* come to mind as missed opportunities.

Thanks,

Muli, Tel Aviv

Subject: Re: Lyrical query
From: hogan@amoebamusic.com
Date: August 10, 1999. 11:03 (GST)

Can somebody comment on the rock songs excerpted?

Muli, I thought you'd never ask:

"Polly says..." is from (you guessed it!) "Polly," a song about love and deception (and associated pain), by Nirvana.

"They hurt me bad..." is from the song "Kiss off" by the Violent Femmes — boy lamenting abandonment by girl.

"Oh my..." is from "Losing my Religion" by the group R.E.M. (guess what the initials stand for). An impenetrably murky ballad about having lost something or other.

"Help me, help me..." is from "Sunny Afternoon" by the Kinks, a delightful satire on the leisure class.

"Down in the basement..." is from "Stop Making Sense" by the Talking Heads. "Na-na-na-na-na" is the only line that makes sense in the whole damn song.

"Having read the book..." is from "A Day in the Life," a diatribe by the Beatles about the meaning of life — it occupied the second side of the album *Sgt Pepper's Lonely Hearts Club Band.* (Remember when albums had sides?) These particular two lines (the first is the tail end of a reference to John Lennon's brief fling with flics, the second was contributed by Paul McCartney) are indeed followed by a long, chaotic crescendo played by a forty-piece string ensemble — John wanted it to sound "like the end of the world".

"I'll come running to tie your shoes" is a love song by Brian Eno.

Finally, "And at night..." is a mystery to me. It does not ring a bell — and, frankly, it doesn't sound right. I suspect it is the translation of a Greek rock song. I'll ask. So, there you have it. Overall, a rather conservative and unexciting selection, in my opinion, even if you restrict yourself to the classics. Strictly basics. No risks, no surprises. (And no Led Zeppelin.)
How do you complete the proof of the pythagorean theorem in Turing? Was this the original proof by Pythagoras? Why do I remember another proof? And didn’t Euclid’s fifth postulate talk about parallel lines, instead of perpendicular lines? Help?

Sumita Lakshmanatharam, Birmingham

Although the pythagorean theorem was known empirically to ancient Babylonians ca. 1900 BC, its first proof may actually have been discovered by Pythagoras himself. There are now literally dozens of known proofs of the theorem. A simple algebraic proof is based on the figure below: The area of the large square is \((a + b)^2\) which we can rewrite as \(a^2 + b^2 + 2ab\). Since the area of the four triangles is \(2ab\), the remaining square, whose area is \(c^2\), must be equal to \(a^2 + b^2\).

The proof outlined by Turing is quite plausibly Pythagoras’ own. The similarity of the two triangles \(ABD\) and \(ABC\) implies the equality of ratios

\[
\frac{AB}{BD} = \frac{BC}{AB},
\]
or \((AB)^2 = BD \cdot BC\). Now the right-hand side of this equation is the area of the shaded rectangle. Similarly, one proves \((AC)^2\) equals the area of the black rectangle. Adding these two equalities we conclude that the sum \((AB)^2 + (AC)^2\) equals the area of the large square, which is precisely \((CB)^2\). Q. E. D.

Now about Euclid’s fifth postulate. Actually, the precise original statement of postulate is a little more complicated: If two lines are drawn which intersect a third line in such a way that the sum of the inner angles on one side is less than two right angles, then the two lines must eventually intersect each other on that side. See the figure below for an illustration.

There are several other statements which can be proved equivalent to the fifth postulate (these equivalence proofs use the other four postulates). Therefore each of these other statements can be considered an alternative “fifth postulate” in the place of the original one. The unique perpendicular postulate in Turing’s lecture is one of these equivalent forms. The Pythagorean theorem is another, and so is the so-called “parallel postulate”: \textit{From a point a unique straight line parallel to a given straight line can be drawn.} The statement that “the angles of any triangle add up to two right angles” is yet another equivalent form of the fifth postulate.

In one of the most striking instances of independent and simultaneous discovery in the history of mathematics, János Bolyai in Hungary and Nicolai Lobachevsky in Russia invented in 1823 non-euclidean geometries, in which the first four postulates hold but not the fifth. Karl Friedrich Gauss had also noticed the possibility, but was apparently so much bothered by it that he excluded it from his writings. The spherical non-euclidean geometry explained by Turing to Alexandros was proposed later, by Bernhard Riemann.

More complex non-euclidean geometries of Riemann’s kind were instrumental in the development, half a century later, of the general theory of relativity by Albert Einstein. According to Einstein’s theory, the geometry of space-time (our three-dimensional world with time added as a fourth dimension) is non-euclidean, and gravitational forces are the result of “wrinkles” in this geometry —i.e., departures from the even roundness of a sphere’s surface.

But, as Turing would say, this is a whole different lecture.

Jonathan Store, Brooklyn

PS: For Net sources on the history of mathematics see http://www.astro.virginia.edu/~eww6n/math. Euclid’s Elements can be found at http://aleph0.clarku.edu/~djoyce/java/elements/toc.html. A repository of information about the ancient world is available at http://www.perseus.tufts.edu/. (One should always keep in mind that Net documents tend to fall from good maintenance and/or disappear without warning.) —js
Incidentally, I discovered by painful experimentation that all Net URLs given in the manuscript are fictional. With one exception — can you find it? — js

Subject: Re: Pythagorean Theorem
From: croquin@maxplanck.de
Date: September 10, 1999. 19:45 (GST)

Many thanks to Professor Store for his lucid commentary on the fifth postulate and non-euclidean geometries, as well as their relationship to relativity. Incidentally, I found it a little irritating that in all of Turing’s discussion of turn-of-the-century science there is no mention of Einstein. (But then again, Darwin is not mentioned either...)

Dr. Charles Roquin, Stuttgart

Aloé said that the code is not quite code.
—She is right. Here is fasterprime in C.

#include "long.h"
#include "stdio.h" fasterprime(x) int x, y;
y = 2; do 
   { if x % y == 0 then return(0)
   else y++ } while y * y <= x return(1)
main() int x;
   { read(x); if fasterprime(x) then printf("%x is a prime") else printf("%x is not a prime") }