Midterm Exam 2, Mar 10, 2000
CSE 105 – Winter ’00

Problem 0 (1 point): PRINT your name: DANIELE MICCIANCIO

Problem 1 (33 point): Let $L$ be the language over the alphabet $\Sigma = \{a,b\}$ of all strings $w \in \Sigma^*$ such that $w$ contains an equal number of $a$’s and $b$’s. Give the formal description of a Turing Machine accepting $L$.

Solution: The language $L$ is decided by the Turing Machine

$$M = (Q, \Sigma, \Gamma, \delta, \text{start}, \text{accept}, \text{reject})$$

defined as follows: the set of states is $Q = \{\text{start}, 1, 2, 3, \text{accept}, \text{reject}\}$, the input alphabet is $\Sigma = \{a,b\}$, the tape alphabet is $\Gamma = \{a, b, \$, x\}$, and the transition function $\delta$ is defined by the following state diagram:

![State Diagram](image)

Figure 1: State diagram
Problem 2 (33 point): Let $L$ be a language over some alphabet $\Sigma$, and let $L' = \{w\#w | w \in L\}$ be the language of all words over $\Sigma \cup \{\#\}$ obtained repeating any word from $L$ twice (with a $\#$ sign in between the two copies). Prove that if $L$ is decidable, then also $L'$ is decidable.

Solution: Let $M$ be a TM that decides $L$. (Such a TM exists by definition of decidable language.) Then the following TM decides $L'$:

On input $x$:

1. First check that the input string has the right form $x = w\#w$. This can be done using the TM $M_1$ from section 3.1 in the book (also described in class).
2. Erase the second copy of the string, overwriting all symbols from the $\#$ and after with blanks.
3. At this point the tape contains the string $w$. Run $M$ on $w$ to decide if $w \in L$. If $M$ accepts, then accept, and if $M$ rejects then reject.

Another possible solution to this problem based on the closure properties of decidable languages is the following: Since $L$ is decidable, then also $L_1 = L \cdot \{\#\}$ is decidable (where $\cdot$ is the concatenation operation). Moreover as proved in class (or section 3.1 in the book) the language $L_2 = \{w\# | w \in \Sigma^*\}$ is decidable. Using the closure of decidable languages under intersection we get that $L' = L_1 \cap L_2$ is decidable.

Problem 3 (33 point): Let $L$ be the following language

$L = \{(G, E) \mid G$ is a CFG and $E$ is a regular expression such that $L(G) \subseteq L(E)\}$.

Prove that $L$ is decidable.

Solution: We describe a TM that decide the language $L$. The idea is the following. We first notice that $L(G) \subseteq L(E)$ is equivalent to $L(G) \subseteq L(E) = \emptyset$. Then we recall that since the complement of a regular language is regular, and the intersection of a context free language with a regular language is context free, then $L = L(G) \cap \overline{L(E)}$ is also context free. Moreover, a context free grammar generating the language $L$ can be computed from the original CFG $G$ and regular expression $E$. Once we compute a CFG $G'$ for $L$, we can check whether $L = \emptyset$ using the decider for $E_{CFG}$. Details follow:

On input $(G, E)$:

1. First check that $G$ is a valid description of a grammar, and $E$ is a valid description of a regular expression.
2. Compute a CFG $G'$ generating the language $L = L(G) \cap \overline{L(E)}$ using the procedures from the book.
3. Run the decider $R$ studied in class (or from Theorem 4.7 in the book) on input $G'$ to decide if $G' \in E_{CFG}$. If $R$ accepts then accept, and if $R$ reject then reject.