

Name:
Tutor:
Student ID:
Signature:

CSE 8B
Final Exam
Fall 2015

You can rip off the last page and use as scratch paper.

Page 2 _____

Page 3 _____

Page 4 _____

Page 5 _____

Page 6 _____

Page 7 _____

Page 8 _____

Page 9 _____

Page 10 _____

Page 11 _____

Page 12 _____ 0 _____

Page 13 _____

Page 14 _____

Total _____

1) Write the output.

<pre>int i = 8; while (i >= 3) { if (i % 2 == 0) { i++; } else { i -= 3; } System.out.println("i = " + i); }</pre>	<u>Output</u>
--	---------------

2) Circle the valid Java identifiers.

Ican't wait4 x-mas break
8B_final Integer st0ck1ng\$ MyCSE8B

3) Write the output.

<pre>public class Question3 { public static void main(String[] args) { int[] arr = { 7, 12, 10, 23, 3, 23, 19, 7, 14, 3 }; int x = 15, y = 0; for (int i = 0; i < arr.length; i++) { int n = arr[i]; if (n < x) { x = n; y = i; } } System.out.println(x + ", " + y); } }</pre>	<u>Output</u>
--	---------------

What is printed if the line *if (n < x)* was changed to *if (n <= x)*?
What is printed if the line *if (n < x)* was changed to *if (n >= x)*?
What is printed if the line *if (n < x)* was changed to *if (n > x)*?
What is printed if the line *if (n < x)* was changed to *if (n == x)*?
What is printed if the line *if (n < x)* was changed to *if (n != x)*?

4) Write the output.

<pre>recurse(6); public void recurse(int x) { if (x < 1) System.out.println("complete"); else { System.out.println(x); recurse(x - 1); } }</pre>	<u>Output</u>
--	---------------

<pre>recurse(6); public void recurse(int x) { if (x > 1) System.out.println("complete"); else { System.out.println(x); recurse(x - 1); } }</pre>	<u>Output</u>
--	---------------

<pre>recurse(0); public void recurse(int x) { if (x >= 8) System.out.println("complete"); else { System.out.println(x); recurse(x + 2); System.out.println(x); } }</pre>	<u>Output</u>
--	---------------

<pre> recurse(-4, 1); public void recurse(int x, int y) { if (x >= 0) System.out.println("complete"); else { System.out.println(x*y); if (y >= 1) y = -1; else y++; recurse(x+1, y); } } </pre>	<u>Output</u>
<pre> recurse(1, 4); public void recurse(int x, int y) { if (x >= 5) System.out.println("complete"); else { recurse(x, y+1); } } </pre>	<u>Output</u>
<pre> recurse(0, -1); public static void recurse(int x, int y) { if (x+y > 4) System.out.println("complete"); else { if (x<y) { System.out.println(x); x++; } else { System.out.println(y); y++; } recurse(x, y); } } </pre>	<u>Output</u>

- 5) You are given two threads, one programmed to continuously print “a” and the other programmed to continuously print “b”, where the second is started before the first. What is the expected output to the terminal?
- The two threads will run one after the other. “b” will print first, followed by a continuous pattern of “abababab...”.
 - Not enough is known about what is going on behind the scenes. There is no predictable pattern and it is unknown what is printed first.
 - Though the second thread was started first, the process of starting the multithreading process causes it to delay for a single clock cycle. Thus “a” prints first, followed by “babababa...”.
 - The process of starting the multithreading process causes a delay for the first thread to start, but by an unknown amount of time, depending on the system. “a” prints first, and continues to print for an unknown amount of time, followed by “babababa...”.
 - None of the above
- 6) When the method start() is called on a thread, e.g. myThread.start(), which method in myThread eventually gets executed?
- 7) Fill in the blanks to finish the abstract class Abstract, the interface Interface, and the class Final. Read the comments and method calls in Final to figure out whether methods and variables are public or private, as well as method return types.

```
public abstract class Abstract {
    private int number;

    // returns class variable "number"
    _____ getNumber() {
        return number;
    }
}
```

```
public interface Interface {
    public abstract double method();
}
```

```
public class Final _____ Abstract _____ Interface {
    public static void main(String[] args) {
        int a = getNumber();
        double b = method();
    }

    _____ method() {...}
}
```

8) Given the following definitions:

```
public interface Countable {  
    public abstract String count(int i);  
}
```

```
class Counter1 implements Countable {  
    private int count;  
  
    public Counter1() {  
        this.count = 1;  
    }  
  
    public String count(int i) {  
        return this.count + ", " + i;  
    }  
  
    public String count(String s) {  
        return this.count + ", " + s;  
    }  
}
```

```
class Counter2 implements Countable {  
    private int count;  
  
    public Counter2() {  
        this.count = 2;  
    }  
  
    public String count(int i) {  
        return this.count + ", " + i;  
    }  
  
    public String count() {  
        if (count < 2) {  
            return "2";  
        } else {  
            return "two";  
        }  
    }  
}
```

And the following variable definitions:

```
Counter1 c1 = new Counter1();  
Counter2 c2 = new Counter2();  
Countable c;
```

What gets printed with the following statements (each statement is executed in the order it appears). If there is a compile time error, write "Error" and assume that line is commented out when run.

```
System.out.println(c1.count());  
System.out.println(c1.count(3));  
System.out.println(c1.count("three"));  
System.out.println(c2.count());  
System.out.println(c2.count(3));  
System.out.println(c2.count("three"));  
c = new Counter1();  
System.out.println(c.count());  
System.out.println(c.count(3));  
System.out.println(c.count("three"));  
c = c2;  
System.out.println(c.count());  
System.out.println(c.count(3));  
System.out.println(c.count("three"));
```

```
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____  
_____
```

9) Write the Big O complexity, in terms of n.

<pre>for (int a = n/2; a > 0; a--) { System.out.println(a); }</pre>	<u>Complexity</u>
<pre>for (int i = n; i > n; i--) { for (int j = 100; j > 20; j--) { System.out.println(i + j); } }</pre>	<u>Complexity</u>
<pre>for (int i = 0; i < n; i += 2) { while (i < n/3) { System.out.println("p"); } }</pre>	<u>Complexity</u>
<pre>for (int i = n; i != 0; i /= 2) { System.out.println(i * 2); }</pre>	<u>Complexity</u>
<pre>for (int i = 0; i < (n - n/2); i++) { for (int j = n; j < (n + n/2); j++) { System.out.println(i + j); } }</pre>	<u>Complexity</u>
<pre>for (int i = 0; i*2 < n+25; i += 4) { for (int j = n; j > 10 + n; j--) { System.out.println(j); } }</pre>	<u>Complexity</u>

10) Given the following filepaths, answer the following questions.

`/home/jundt/cs8b/homework/HW4.java`

`/home/jundt/cs8b/something.jar`

Provided you are in `/home/jundt/cs8b`, what command do you use to change your working directory to `/home/jundt/cs8b/homework`?

Once you are in the `/home/jundt/cs8b/homework` directory, which command would you use to show the contents of the directory?

You see that there is a file named `HW4.java`. How would you open this file in vim? How would you close it once it is open in vim?

Let's say you wanted to make a backup of `HW4.java`. What command would you use to make a copy of `HW4.java` in `/home/jundt/cs8b`?

Imagine that `HW4.java` requires `something.jar` to be in its classpath during compilation. How would you compile it?

How would you run `HW4.java`?

11) Write the output.

```
char[][] array = {{'l', 'g', 'f', 'h'},
                  {'8', 'b', 'g', 'g'}};
alakazam(array[0], array[1][2]);

for (int i = 0; i < array.length; i++)
    for (char j : array[i])
        System.out.println(j);

public void alakazam(char[] array, char mod) {
    for (int i = 0; i < array.length; i--) {
        if (array[i] == mod) {
            char temp = array[i];
            array[i] = array[i-1];
            array[i-1] = temp;
            mod++; // Goes to next char
        }        // Ex. 'a'++ turns to b
    }
}
```

Output

12) Write the methods that run, in order.

<pre>method1(); // throws Exception try { method2(); method3(); // throws Exception } catch (Exception e) { method4(); method5(); }</pre>	<u>Methods</u>
<pre>try { method1(); // throws Exception method2(); } catch (Exception e) { method3(); return; } finally { method4(); return method5(); }</pre>	<u>Methods</u>
<pre>try { method1(); method2(); // throws Exception method3(); } catch (Exception e) { } finally { method4(); } method5();</pre>	<u>Methods</u>

13) Select the best data structure for the given situation.

array ArrayList HashMap HashSet

Saving a fixed amount of integers in a specific order: _____

Matching a bar code number with an item: _____

Storing a large amount of objects, so you can find one in O(1) time later: _____

14) Place the following variables under the right location in memory based on where the content is stored. Remember the difference between a pointer that references the content versus the actual content itself.

```
static int a = 0;           String[] nam = {};           boolean pass = "false";

static float f = 1.0;       float frac = 2.0;

Integer i = new Integer(42);      double[] array = {1, 2, 3};

ArrayList<int> dynamicList = new ArrayList<int>();
```

<u>Static</u>	<u>Stack</u>	<u>Heap</u>

15) Given the following method:

```
public int find(int n1, int n2)
{
    if (n1 < n2)
        return n1;
    else
        return n2;
}
```

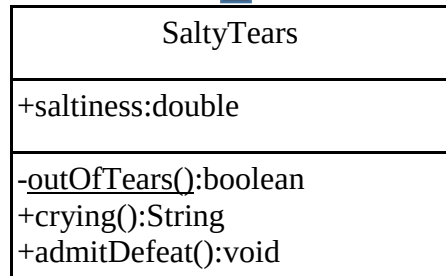
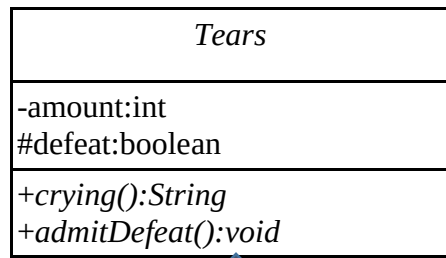
Fill in the blanks with the correct integer that passes the test.

<pre>@Test public void test1() { int n1 = 99, n2 = 66; assertEquals(find(n1, n2), ____); }</pre>	<pre>@Test public void test2() { int n1 = 50, n2 = 50; assertEquals(find(n1, n2), ____); }</pre>
---	---

Circle the correct answer.

- Unit tests are designed to test the functionality of the entire project. (TRUE / FALSE)
- In test-driven development, when should unit tests be written in relation to the actual code? (BEFORE / AFTER)

16) Given the following UML diagrams, reconstruct the classes in code. If a method is not abstract, put ellipses in the method body. (Ex. public void method() {...})



17) Given the following class definitions, write the output.

```
class ClassA {
    public void method1() {
        System.out.println("ClassA 1");
    }

    public void method2() {
        System.out.println("ClassA 2");
    }
}

class ClassB extends ClassA {
    public void method1() {
        super.method1();
        System.out.println("ClassB 1");
    }

    public void method3() {
        super.method2();
        System.out.println("ClassB 3");
        method1();
    }
}

class ClassC extends ClassA {
    public void method1() {
        super.method2();
        System.out.println("ClassC 1");
    }

    public void method2() {
        System.out.println("ClassC 2");
    }
}

class ClassD extends ClassC {
    public void method2() {
        System.out.println("ClassD 2");
    }

    public void method3() {
        super.method1();
        System.out.println("ClassD 3");
    }
}
```

<pre> public class Question18B { public static void main(String[] args) { ClassA ref1; ref1 = new ClassD(); ref1.method1(); System.out.println("-----"); ref1.method2(); System.out.println("-----"); ((ClassD) ref1).method3(); } } </pre>	<u>Output</u>
<pre> public class Question18B { public static void main(String[] args) { ClassA ref1; ref1 = new ClassB(); ref1.method1(); System.out.println("-----"); ref1.method2(); System.out.println("-----"); ((ClassB) ref1).method3(); } } </pre>	<u>Output</u>

18) Select the valid constructors for the following classes.

a) BufferedInputStream

1. BufferedInputStream(RandomAccessFile ram)
2. BufferedInputStream(FileInputStream fis)
3. BufferedInputStream(FileOutputStream fos)

b) PrintWriter

1. PrintWriter(File source)
2. PrintWriter(FileOutputStream fos)
3. PrintWriter(Scanner scnr)

19) Fill in the blanks with one of the following words.

implements, compilation, default, extends, runtime, package, stream, abstract, text editor, IDE, GUI, UML, data structure, thread, polymorphism

- A compiler error occurs when you try to compile it. The programmer is notified of the error during at/during _____.
- While both are useful tools for programmers, a(n) _____ will usually have more features than a(n) _____, such as debug mode.
- If a program is expected to take a long time, it can be useful to distribute the workload onto more than one _____.
- Buttons, file choosers, and text fields are some components that a(n) _____ typically carries.
- After you are done opening a File object, you should remember to close the file _____.
- Common _____ (s) include arrays, lists, hashmaps, and sets.
- The Java keyword which denotes inheritance of interface is _____, the Java keyword which denotes inheritance of abstract classes is _____.

Scratch Paper