AP Computer Science
Fall 2014

Test 1

Name: _____
Mr. A. Tareen

Part I. (53 points) Solve each of the following problems. For the multiple choice problems, select the correct answer by placing an "X" in the box beside it.

(1pt) **1.** Java source code files are created and saved using what file extension?

☐ `.jav` ☐ `.jcod` ☐ `.jsrc` ☐ `.java`

1 pt

(1pt) **2.** What is the data type of the following variable: `luckyNumber = 7`

☐ `String` ☐ `int` ☐ `double` ☐ `boolean`

1 pt

(1pt) **3.** What is the data type of the following variable: `sport = "football"`

☐ `String` ☐ `int` ☐ `double` ☐ `boolean`

1 pt

(1pt) **4.** What is the data type of the following variable: `doorOpen = true`

☐ `String` ☐ `int` ☐ `double` ☐ `boolean`

1 pt

(1pt) **5.** Which of the following is *not* a correct variable name?

☐ `2bad` ☐ `zero` ☐ `lastValue` ☐ `year2000`

1 pt

(1pt) **6.** What is the value of the following expression: `(2 - 6) / 2 + 9`

☐ `7` ☐ `8` ☐ `9` ☐ `10`

1 pt

(1pt) **7.** Which of the following symbols is used in Java to represent the `OR` operator?

☐ `||` ☐ `&&` ☐ `??` ☐ `%%`

1 pt

(1pt) **8.** How many choices are possible when using a single `if-else` statement?

☐ `1` ☐ `2` ☐ `3` ☐ `4`

1 pt

(1pt) **9.** Which of the following defines the category that Java's `int`, `double` and `boolean` data types belong to?

☐ `intrinsic` ☐ `primitive` ☐ `object` ☐ `numeric`

1 pt

(1pt) **10.** Which of the following variable declarations is *not* correct?

☐ `double duty;`

☐ `String music;`

☐ `boolean lights = 12;`

☐ `int start = 34, end = 99;`

1 pt

10 pts

(2pts) **11.** Which one of the following is *not* a correct arithmetic expression?

2 pts

- ☐ `alpha`
- ☐ `(alpha / momentum) - 12.4`
- ☐ `alpha (/ momentum - 12.4)`
- ☐ `((alpha / momentum) - 12.4)`

(2pts) **12.** What does the following code fragment print to the output terminal?

2 pts

```
int sum = 14;
if (sum < 20)
{
    System.out.print("Under");
}
else
{
    System.out.print("Over");
    System.out.println(" the limit.");
}
```

- ☐ `Under`
- ☐ `Over`
- ☐ `Under the limit.`
- ☐ `Over the limit.`

(2pts) **13.** What does the following code fragment print to the output terminal?

2 pts

```
int height = 7;
if (height <= 6)
{
    System.out.print("Low bridge: ");
}
System.out.println("proceed with caution.");
```

- ☐ *Nothing will be printed*
- ☐ `Low bridge:`
- ☐ `proceed with caution.`
- ☐ `Low bridge:  proceed with caution.`

(2pts) **14.** What is the output of the following program fragment?

2 pts

```
for (int i = 0; i < 5; i++)
{
    System.out.print(i + " ");
}
System.out.println();
```

- ☐ `1 2 3 4 5`
- ☐ `0 1 2 3 4`
- ☐ `0 1 2 3 4 5`
- ☐ `i i i i i`

8 pts

(3pts) **15.** What output will be produced by the following:

```
System.out.print("\\* This is not\n a comment *\\");
```

3 pts

- ☐ `* This is not a comment *`
- ☐ `\* This is not a comment *\`
- ☐ `* This is not`
  `a comment *`
- ☐ `\\* This is not`
  `a comment *\\`
- ☐ `\* This is not`
  `a comment *\`

(3pts) **16.** What value is stored in `result` if:

```
int result = 13 - 3 * 6 / 4 % 3
```

3 pts

- ☐ `-5`
- ☐ `0`
- ☐ `13`
- ☐ `-1`
- ☐ `12`

(3pts) **17.** Suppose that base-2(binary) numbers and base-16(hexadecimal) numbers can be denoted with subscripts, as shown below:

3 pts

$$2A_{\text{hex}} = 101010_{\text{bin}}$$

Which is equal to $3D_{\text{hex}}$?

- ☐ 111101_{bin}
- ☐ 101111_{bin}
- ☐ 10011_{bin}
- ☐ 110100_{bin}
- ☐ 101101_{bin}

(3pts) **18.** Which of the following pairs of declarations will cause an error message?

3 pts

I
```
double x = 14.7;
int y = x;
```

II
```
double x = 14.7;
int y = (int) x;
```

III
```
int x = 14;
double y = x;
```

- ☐ None
- ☐ I only
- ☐ II only
- ☐ III only
- ☐ I and III only

12 pts

(4pts) **19.** Convert the following hexadecimal(base-16) numbers to decimal(base-10). Show your calculations.

4 pts

(a) (2 pts)

$2FED_{hex}$

(b) (2 pts)

$BEEF8_{hex}$

(4pts) **20.** Convert the following binary(base-2) numbers to hexadecimal(base-16).

4 pts

(a) (2 pts)

1110111010_{bin}

(b) (2 pts)

$1011011010110011111_{bin}$

(3pts) **21.** Use DeMorgan's Law to simplify the following logical statement:

```
!( !(a && !b) && (!b || c) )
```

3 pts

11 pts

(3pts) **22.** You are going shopping for meat and milk, but there is sales tax. You buy \$2.00 worth of milk, and \$4.00 worth of meat, and the sales tax rate is 3%. Write a Java program called `Groceries` that calculates the total cost of your groceries.

3 pts

(3pts) **23.** The square numbers are the integers of the form $a \times a$, for example, 9 is a square number since $3 \times 3 = 9$. Write a Java program called `Squares` that reads in an integer n from the input, and outputs all the positive square numbers less than or equal to n, one per line, in increasing order. For example, if the input is 9, then the correct output would be:

```
1
4
9
```

3 pts

6 pts

(6pts) **24.** Consider the following code listing for the video game `League Of Hackers`. Identify the following parts of the game by writing each number beside its corresponding code line in the program.

6 pts

1. This allows methods from the `util` library to be used.
2. This prints the character's name to the terminal output.
3. This reads in the value of `strength` from the console.
4. This declares and initializes the variable `total`.
5. This loop guards against a user inadvertently entering values greater than 10.
6. This logical statement determines whether a user has assigned too many characteristic points.

```
//----BEGIN CODE FOR LEAGUE OF HACKERS----
import java.util.Scanner;
public class LeagueOfHackers
{
    public static void main(String[] args)
    {
        String name = "";
        Scanner person = new Scanner(System.in);
        int strength = 0;
        int health = 11;
        int total = 0;
        Scanner scan = new Scanner(System.in);

        System.out.println("Welcome to League of Hackers!");
        System.out.println("Enter the name of your character:");
        name = person.nextLine();

        System.out.println("Enter strength (1-10):");
        strength = scan.nextInt();

        System.out.println("Enter health (1-10):");
        while (health > 10)
        {
            System.out.println("Input a value less than or equal to 10.");
            health = scan.nextInt();
        }

        total = strength + health;
        if (total > 10)
        {
            System.out.println("You have given your guy too many points!");
            System.out.println("Default values have been assigned.");
            strength = 5;
            health = 5;
        }
        System.out.println(name);
        System.out.println("strength: " + strength + " health: " + health);
    }
}
```

6 pts