

Nov 10, 2006

Directions: This is a closed book, closed notes examination. Place your answers in the space provided. The point value of each question is indicated. There are a total of ?? 78 points. Your total out of ?? 78 will be weighted to 100 points. You have 55 minutes for this examination.

1. (9 pts) Rewrite the following code segment so that it will print out all integers from 1 to `max` separated by commas? For example, if `max` is 5, the expected output is: 1, 2, 3, 4, 5

```
for (int i = 1; i <= max; i++) {
    System.out.print(i + ", ");
}
System.out.println();
```

Ans:

```
System.out.print(1);
for (int i = 2; i <= max; i++) {
    System.out.print(", " + i);
}
```

2. (6 pts) Given the following variable declarations, what is the value of each of the following expressions.

```
int x = 10;
int y = -3;
int z = 7;
```

a) `y * y <= z` Ans: **false**

b) `x - y != z` Ans: **true**

c) `y / y == 1` Ans: **true**

3. (12 pts) Consider the following Java method which is written incorrectly. The method is supposed to return how many of its three arguments are odd.

```
public static int numOdd(int n1, int n2, int n3) {
    int count = 0;

    if (n1 % 2 == 1)
        count++;
    else if (n2 % 2 == 1)
        count++;
    else if (n3 % 2 == 1)
        count++;

    return count;
}
```

- a) Briefly explain what is wrong with it.

Ans: Once one odd element is found, no other elements will be checked to see if they are odd.

- b) Rewrite this code segment so that it works correctly.

Ans:

```
public static int numOdd(int n1, int n2, int n3) {
    int count = 0;
    if (n1 % 2 == 1)
        count++;
    if (n2 % 2 == 1)
        count++;
    if (n3 % 2 == 1)
        count++;
    return count;
}
```

4. (6 pts) Describe a potential problem with the following code segment:

```
Scanner console = new Scanner(System.in);
System.in.print("What is your name? ");
String name = console.next();
if (name == "Tom")
    System.out.println("Mine is too!");
```

Ans: The test in the if expression will never be true. The test compares the addresses of two different `String` objects for equality. It does not compare the 2 `String` objects to see if they are identical. The test should be `name.equals("Tom")`.

5. (9 pts) Write a method `isVowel()`. This method accepts a `char` as a parameter and returns `true` if that character is a vowel: a, e, i, o, u

Ans:

```
public static boolean isVowel(char c) {  
    if (c >= 'A' && c <= 'Z')  
        // convert c to its lowercase equivalent  
        c = (char) ((c - 'A') + 'a');  
  
    // return true if it's a lowercase vowel  
    return c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u';  
}
```

6. (9 pts) Consider the following method:

```
public static int mystery (int x, int y) {  
    while (x != 0 && y != 0) {  
        if (x < y)  
            y -= x;  
        else  
            x -= y;  
    }  
    return x + y;  
}
```

For each call below, indicate what value is returned:

- a) `mystery (3, 3)` Ans: 3
b) `mystery (12, 18)` Ans: 6
c) `mystery (30, 75)` Ans: 15

7. (9) Write a `do-while` loop that repeatedly prints random integers between 0 and 1000 inclusive until a number above 900 is printed. At least one line of output is printed, even if the first random number is above 900. Here is some example output:

```
Random number: 233
Random number: 34
Random number: 810
Random number: 905
```

Ans.

```
Random rand = new Random();
do {
    int n = rand.nextInt(1001);
    System.out.println("Random number: " + n);
} while (n <= 900);
```

8. (9) The following code segment is not correct nor is it robust against invalid user input. Modify the code so that it will not proceed until the user has entered a valid age and GPA. (Assume that any `int` is a legal age and that any `double` is a legal GPA).

```
Scanner sc = new Scanner(System.in);
System.out.print("Enter your age: ");
int age = sc.nextInt();

System.out.print("Type your GPA: ");
double gpa = sc.nextInt();
```

Ans.

```
Scanner sc = new Scanner(System.in);
do {
    System.out.print("Enter your age as in integer: ");
} while (!sc.hasNextInt());
int age = sc.nextInt();

do {
    System.out.print("Enter your GPA as a number: ");
} while (!sc.hasNextDouble());
double gpa = sc.nextDouble();
```

9. (9 pts) Consider the following input file called `fungus.txt`.

```
3.4      7
    dog
        6.7    9    cat

farm    89
```

Now consider the following program. Assume the program is in the same directory as the above file.

```
public class Q9 {

    public static void main (String [] args) throws Exception {
        Scanner sc1 = new Scanner(new File("fungus.txt"));
        PrintStream ps = new PrintStream(new File("stuff.txt"));
        String output = foo(sc1, ps);
        ps = new PrintStream(new File("fungus.txt"));
        ps.print(output);
    }

    public static String foo (Scanner input, PrintStream output) {
        String other = "";
        while(input.hasNext()) {
            if (input.hasNextDouble())
                output.println(input.nextDouble());
            else
                other = other + input.next() + " ";
        }

        other = other + "\n***\n";
        return other;
    }
}
```

What files exist after this program is run, and what are their contents?

Ans:

The file `fungus.txt` will contain:

```
dog cat farm
***
```

The file `stuff.txt` will contain:

```
3.4
7.0
6.7
9.0
89.0
```