Computer Science II – Prepa Tec Campus Eugenio Garza Lagüera
Activity 1: Review

"En esta actividad me comprometo a aplicar mis conocimientos, esforzarme en su desarrollo y no servirme de medios no autorizados o ilícitos para realizarla. Es de mi conocimiento, que debo entregar a través de la plataforma Canvas los procesos realizados ya que ningún resultado tiene valor sin proceso."

Name: _____ Student Id: _____ Signature: _____

Turn in the answers to the following exercises by hand.

Problem 1

```
1  public class P1{
2
3      public static void main(String[] args){
4          int a = 0, b=0, c=3;
5
6          System.out.println(5%2*4);
7
8          while (a>=b){
9              for(b=1; b<5;++b){
10                 ++c;
11                 System.out.println(a + " " + b + " " + c);
12             }
13             System.out.println(b+b+c);
14         }
15     }
16
17 }
18
```

Complete a code trace of the code above.

a	b	c	Output
0	0	3	4
	1	4	0 1 4
	2	5	0 2 5
	3	6	0 3 6
	4	7	0 4 7
	5		17

a) What is the difference in output between lines 11 and 13?

The print statement in line 11 prints the values of a, b, c separated by a space. Line 13 adds b+b+c and prints the result.

b) How many times is the **for** loop executed? Why?

4 times. The for loop runs while b < 5, so it will enter the loop when b=1, b=2, b=3 and b=4.

c) How many times is the **while** loop executed? Why?

1 time only. Inside the while loop body, the value of b is incremented and a is unchanged, so a >= b will evaluate false.

d) What does line 10 do?

Increment the value of c by 1.

e) What is the control variable for each loop in the program?

For both loops, the control variable is b.

Problem #2

```
1  public class P2{
2
3      public static void main(String[] args){
4          int x=5, y=0, z=3, b=0;
5
6          y = (int)(1.5*10)%2 + z*9;
7          if (y>7){
8              for (b = 1; b<3; ++b){
9                  System.out.println(y+" "+b);
10             }
11         }
12         else {
13             System.out.println(y+b+" ");
14         }
15     }
16
17 }
```

Complete a code trace of the code above.

b	x	y	z	Output
0	5	0	3	28 1
1		28		28 2
2				
3				

a) Are there any errors in the code above?

No

b) What is the final value of variable y?

28

c) What is the difference between the output generated by lines 9 and 13?

Line 9 prints the value of y and b separated by a space. Line 13 would first add the values of y and b, then print the

d) How many times is the **for** loop executed?

2 times, when b=1 and b=2. The exit condition is that b>=3.

e) Where is the **casting** being done in the code above? What is the purpose of it?

In line 6, the result of (1.5 * 10) is being casted to an integer from a double, loosing the decimal precision in the operation.

Problem #3

```
10      public static void main(String[] args)  throws IOException
11      {
12          int x=5, y=0, z=3, b=0;
13
14          y= ((int)( 4.5 % 5 *10) + x * 5);
15          switch (y){
16              case 10: case 20: case 30: case 60:
17                  {
18                      stdOut.println("fecha limite");
19                      break;
20                  }
21              case 25: case 35: case 70: case 90:
22                      stdOut.println( y + 12);
23
24              case 12: case 18: case 15: case 45:
25                      stdOut.println ( y * 2);
26              default:
27                      stdOut.println ( y + 20 % 4);
28              }
29              stdOut.println( "quedo" + y);
30      }
```

```
C:\Program Files\Xinox Software\JCreat
82
140
70
quedo70
Press any key to continue...
```

a) What does line 16 do?

It specifies a path to be executed when y is equal to 10, 20, 30 or 60.

b) Which braces { } can be considered optional in the code above? Why?

those in line 17 and 20, as they don't have any impact on the code.

c) Are there any errors in the code above? Are they logic, syntax or runtime errors?

The paths in line 21 and 24 don't have any breaks in them, so a logic error could be produced.

d) What do you have to keep in mind when using a **switch** statement?

You need to define the different paths using the case keyword, and be careful about including all necessary breaks in the code.

e) List two advantages of using a **switch** offer an **if** in the code above.

the case produces code that is easier to read, as well as easier to maintain.

You avoid having to repeat an if statement with multiple complex boolean expressions for each path in your code.