CIS 5100:

Homework 4

Data Structures & Program

By: Bernice Templeman

09/13/2015

Btempleman2015@my.fit.edu

4.14 What is the difference between pre-incrementing and post-incrementing a variable?

In a statement by itself, pre-incrementing and post-incrementing have the same effect.

It is only when a variable appears in the context of a larger expression that pre & post have different effects.

* Pre-Incrementing a variable: adds one then performs the operation
* Post-Incrementing a variable: performs the operation then adds one

4.15 Identify and correct the errors in each of the pieces of code

a.) if (age >= 65);

System.out.println("Age is greater than or equal to 65");

else

System.out.println("Age is less than 65");

//

// age not declared and initialized

// The code will not compile because of a syntax error on "else"

// The error is caused because of the semicolon in

// if (age >= 65);

//

// corrected code

**int** age = 66;

**if** (age >= 65)

System.***out***.println("Age is greater than or equal to 65");

**else**

System.***out***.println("Age is less than 65");

b.

**int** x = 1, total;

**while**(x<=10)

{

total += x;

++x;

}

System.***out***.printf("x = %d Total = %d ", x, total);

// error given because total is not initialized

// no output displayed

// corrected code: added print to show values of x & total

**int** x = 1, total=0;

**while**(x<=10)

{

total += x;

++x;

System.***out***.printf("x = %d Total = %d \n", x, total);

}

System.***out***.printf("x = %d Total = %d ", x, total);

c. **while**(x <= 100)

total +=x;

++x;

// x not declared & initialized

// program compiles and runs but does not display results

// infinite loop because x is not incremented inside loop

// there are no brackets

//

// corrected:

**int** x = 0;

**int** total = 0;

**while**(x <= 100)

{

totaly +=y;

++y;

}

System.***out***.printf("y = %d TotalY= %d\n", y, totaly);

d. **while** (y > 0)

{

System.***out***.println(y);

++y;

// y not initialized

// missing closing bracket'

// y not decremented so will always be greater than 0

// infinite loop

// corrected

**int** y = 1;

**while** (y > 0)

{

System.***out***.println(y);

y--;

}

4.16 What does the following program print

1

4

9

16

25

36

49

64

81

100

Total is 385