**CMSC 155 Spring 2020**

**Homework 1 (130 points available)**

**Deadline: Midnight Thursday Jan 23**

Please copy the solution to each homework problem into the Word file. If the question requires an answer in English (e.g. True/False, short answer) put that in as well. Please hand in the Word file If you have problems with Scholar, please let me know.

1. Exercises (30 points)
2. Read the syllabus and let me know if you have any questions. (5 points)

I have no questions over the syllabus.

1. Stop by my office. (5 points)

1. If s1 is “Five”, s2 is “x”, and x is the integer 5, what are the values of the following? First try predicting the values and then try executing them? Did you predict correctly, if not explain the difference. (5 points)

1. s1 + s2 🡪 Prediction: “Fivex” 🡪 Answer: “Fivex”
2. x + x 🡪 Prediction: 10 🡪 Answer: 10
3. s2 + x 🡪 Prediction: Error 🡪 Answer: x5 🡪 The difference is that the integer x is converted to a string to join together with s2.
4. x + s1 + x + 1 🡪 Prediction: 5Five51 🡪 Answer: 5Five51
5. x + s1 + (x + 1) 🡪 Prediction: 515Five 🡪 Answer: 5Five6 🡪 The difference is that the parentheses separate the integer based addition from the addition of strings instead of doing the parentheses first as a string.
6. **Integer and Floating-Point Division**.

If x is the integer 5, what are values of the following? First try predicting the values and then try executing them? Did you predict correctly, if not explain the difference. (5 points)

* 1. x / 2 🡪 Prediction: 2 🡪 Answer: 2
  2. x / 2.0 🡪 Prediction: 2.5 🡪 Answer: 2.5
  3. (float) x / 2 🡪 Prediction: 2.5 🡪 Answer: 2.5
  4. (float) (x / 2) 🡪 Prediction: 2.5 🡪 Answer: 2.0 🡪 The difference is that it does the integer division and then converts it to a float.
  5. 2 / x 🡪 Prediction: 0 🡪 Answer: 0

1. **Variable Types**. (5 points)
   1. Why is the most commonly used floating point type called "**double**"?

It is called a double because it doubles the number of bits that are represented in the floating number. Float = 32 bits and double = 64 bits.

* 1. Assume that x is an initialized int variable. What type of variable does t have to be for the expression t = (x = 5) to be valid?

T can be a float or int. If it is a float than x will be printed as a float value.

* 1. Assume that x is an initialized int variable. What type of variable does t have to be for the expression t = (x == 5) to be valid?

T has to be a Boolean expression in this one because x is comparing two variables or objects resulting in true or false.

* 1. Assume that x is an initialized int variable. What type of variable does t have to be for the expression t == (x = 5) to be valid? (to test, put expression inside a print statement)

T has to be an initialized float or int for the comparison to print true or false.

* 1. Assume that x is an initialized int variable. What type of variable does t have to be for the statement t == (x == 5) to be valid? (to test, put expression inside a print statement)

1. **Modulus**.

The "modulus" operator, denoted by %, computes the remainder after division. ***Predict*** the result, then ***check if you are correct*** . (10 points)

1. 13 % 4 (b) 16 % 4 (c) 4 % 12

(d) 7% 1 (e) 5 % 0 (f) (109 % 2) >= 1

(g) (8 % 5) > (5 % 8) (h) 3.5 % 1.1 (i) -19 % 4

1. 1🡪 True
2. 0 🡪 True
3. 4 🡪 True
4. 0 🡪 True
5. 5 🡪 False (there is an error)
6. True 🡪 True
7. False 🡪 True
8. 3 🡪 false (.199 repeating)
9. -3 🡪 True
10. **Relational and Boolean Operators**.

If x is the integer 5 and y is the double 5.0, what are the following? (10 points)

* 1. y <= 5 🡪 True
  2. x == y 🡪 True
  3. (x < 10) && (y > 6) 🡪 False
  4. (x < 0) || (y < 6) 🡪 True
  5. !(y < y) 🡪 True
  6. (x < y + 1) && (x + y != 11) && !(x == 5) 🡪 False
  7. ( (x != 4) && (y != 5) ) || ( (x - 10 < y) && (y < x + 10) ) 🡪 True

1. **Rounding**. (10 points)
   1. Assume height is a floating point variable and we want to round its value and store the result in an int. Explain why the following is NOT valid.

int roundedHeight = Math.round(height);

* 1. What should be done instead if the result **needs** to be stored in an **int (not a long or a double)**?

1. **Conditionals.** (10 points)

For each of the following code snippets, identify and fix the errors so that the code compiles and works as intended.

* 1. **int** x = 1;

**if** x > 0 then System.out.print(x);

int x = 1;  
if (x > 0) {  
 System.*out*.println(x);  
}

* 1. **int** x = 1;

**int** y;

**if** (1 + x > Math.*pow*(x, Math.*sqrt*(2))) {y = y + x;}

int x = 1;  
int y = 2;  
if (1 + x > Math.*pow*(x, Math.*sqrt*(2))) {  
 y = y + x;

* 1. **int** x = 1;

**int** y;

**if** (x = 1) {y++;}

int x = 1;  
int y = 2;  
if (x == 1) {  
 y++;  
}

* 1. String letterGrade = "F";

int grade = 95; //Note this must work for all values of grade

**if** (grade >= 90) {letterGrade = "A";}

**if** (grade >= 80) {letterGrade = "B";}

**if** (grade >= 70) {letterGrade = "C";}

**if** (grade >= 60) {letterGrade = "D";}

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

String letterGrade = "F";  
int grade = 69; //Note this must work for all values of grade  
if (grade >= 90) {letterGrade = "A";}  
if (grade >= 80 && grade < 90) {letterGrade = "B";}  
if (grade >= 70 && grade < 80) {letterGrade = "C";}  
if (grade >= 60 && grade < 70) {letterGrade = "D";}  
System.*out*.println(letterGrade);

1. **Augmented Assignment.**

Suppose x is 2 and y is 3 and both are declared to be int. Show step-by-step by hand how the values of x and y change as the following statements are executed. Confirm your answer by programming it. (5 points)

y += x \* y;

y+= 2\*3 🡪 3 += 6 🡪 y = 9

int x =2;  
int y = 3;  
y += x \* y;  
System.*out*.println(y);

x \*= y/x;

2 \*= 3/2 🡪 2/1 \*= 3/2 🡪 x = 2

int x = 2;  
int y = 3;  
x \*= y/x;  
System.*out*.println(x);

1. **Augmented Assignment**.

Give an example where x += y does NOT have the same effect as x = x + y. (5 points)

1. **Augmented Assignment**.

Assume **x is an initialized int variable**. What is the value of x after **all** the following sequence of statements? Explain your answer. (For example, if x is initialized to 5, what is the value after all three of the statements are executed) (5 points)

x += x;

x -= x;

x += x;

int x = 5;  
x += x;  
x -= x;  
x += x;  
System.*out*.println(x);

The result is 0.

1. **Conditional.** **(10 points)**

Write a program that performs the following algorithm:

* inputs a temperature from the user
* if the temperature is less than 30 print "Cold" and print out the difference between temperature and 30.
* Otherwise, print “Not Bad”

Scanner input = new Scanner(System.*in*);  
 System.*out*.println("Input the temperature: ");  
 int temp = input.nextInt();  
  
 if (temp < 30) {  
 System.*out*.println("Cold " + (30 - temp));  
 } else {  
 System.*out*.println("Not Bad");  
 }  
}

1. **Selection Operator.** (5 points)

Suppose **a** is 3 and **b** is 1. What is the value of **b** after the statement?

1. b = (a == b++) ? b - 1 : a + 1; Explain your answer.

The value of **b** is 4. This is because a == b++ is false therefore it makes b = a + 1 which equals 4.

1. Suppose a is 3 and b is 2, will the result be any different? Why or why not?

Yes it will produce the same result because b doesn’t add one till after the comparison happens. Therefore, it will be false again and return the same value, 4.

1. Set a back to 3 and b to 1. Repeat steps a and b using the following line

b = (a == ++b) ? b - 1 : a + 1;

Are the results the same or different? Explain your answer.

The result is the same for step a) but different for step b) because 1 is added to b before the comaparison happens therefore making it true and doing b – 1 which equals 2 (3-1 = 2).

1. **Conditionals**. (5 points)

Suppose x is 4 and y is 5.

* 1. What is the result of the following? Notice the semi-colon at the end of the first line.

if (x == y);

else System.out.println("Hi");

This outputs “Hi”.

* 1. Re-write the code in (a) so that it is clearer and has the same effect.

1. int x = 4;  
   int y = 5;  
   if (x == y) {  
   } else System.*out*.println("Hi")

Don't forget to read the slides for this week and next week!