Name: _____ **NSID:** _____ **Student #:** _____

CMPT 270 – 2017-2018 Term 01

Assignment 1:
Welcome to Java

Submission Deadline: 26 September 2017 – at start of lecture
(no late submissions accepted for this assignment because of paper hand-in)
Submit in person!

Description

The objective of this assignment is to get familiar with Java syntax and writing simple java methods.
As practice for writing code by hand on exams, you are to write the code for each question in hand writing. You can practice writing and testing the code on a computer, but I suggest you write your solution from scratch as practice for exams.
Use precise syntax for Java. 1 mark off per *compiler* and/or *run-time* error.
All methods should be commented with correct `javadoc` syntax. 1 mark off per missing/bad comment.

It is very important that your hand writing is neat and legible — if the TA cannot read your writing, you will lose marks!

Submission

***** All submissions must be submitted on paper in class**

Marking

1 (/5)

2 (/10)

3 (/5)

4 (/5)

5 (/5)

6 (/10)

Total (/40)

This is an individual assignment. You are encouraged to discuss the general concepts of Java syntax, types, variables, methods, arrays, Strings, ifs, loops, etc. with your classmates, but the specifics for the applications in this assignment should be done completely individually. Students that copy / share work will be penalized.

Name: _____ **NSID:** _____ **Student #:** _____

Question 1: [5 marks]

Write a class named `ArrayAverager` that implements a method named `average` to return the average value of an array of type `double`, as done in this python snippet:

```
def Average(S):
    total = 0
    for x in S:
        total = total + x
    average = total / len(S)
    return average
```

Write your class based on this provided `main` method:

```
public static void main(String[] args) {
    double numberArray[] = {1, 3, 4, 5};
    double avgValue = average(numberArray);
    System.out.println("average = " + avgValue);
}
```

Name: _____ **NSID:** _____ **Student #:** _____

Question 2: [10 marks]

Write a class named `Gambler` that implements the following code in its `main` method. Hint: use `Math.rand()` and `Math.round()`. Note: the console output should be formatted exactly as shown on the right:

```
# CMPT 145
# Gambler's ruin example problem

# starting stake and starting goal
# do lots of experiments
#   do lots of games
#     for each game, record wins, and games
#   for each experimetn record succes/failure
# when games are over, divide for probailities

import random as rand

successes = 0
for x in range(1000):
    stake = 100
    bets = 0
    while stake > 0 and stake < 200:
        play = rand.randint(0,1)
        bets += 1
        if play == 0:
            stake = stake - 1
        else:
            stake = stake + 1
    if stake == 200:
        successes += 1

    print('stake =', stake, 'bets made =', bets)

print(successes/1000)
print(bets/1000)
```

```
stake = 200 bets made = 9130
stake = 0 bets made = 3792
stake = 200 bets made = 3924
stake = 200 bets made = 13124
stake = 200 bets made = 4100
stake = 200 bets made = 1278
stake = 200 bets made = 25956
stake = 0 bets made = 18130
stake = 0 bets made = 4272
stake = 200 bets made = 7556
stake = 0 bets made = 2706
stake = 0 bets made = 7726
stake = 0 bets made = 3756
stake = 0 bets made = 9638
stake = 200 bets made = 5604
stake = 0 bets made = 13734
stake = 0 bets made = 19584
stake = 200 bets made = 2370
stake = 200 bets made = 6236
average success = 0.477
average bets = 6.236
```

Name: _____ **NSID:** _____ **Student #:** _____

Question 3: [5 marks]

Write a class named `Greeter` that implements a method named `introductions`, that prompts the user for their name (using the `java.util.Scanner` class), displays a greeting, and returns the name as a `String`, as done in this python snippet:

```
# defines the function only:
def introductions(greeting):
    print(greeting)
    x = input('Please enter your name: ')
    print('Hello,', x)
    return x

# this function call actually calls the function,
# which executes its code.
username = introductions('Welcome to my Python program!')
```

Write your class based on this provided `main` method:

```
public static void main(String[] args) {
    String username = introductions("Welcome to my Java program!");
    System.out.println("got username "+username);
}
```

Name: _____ **NSID:** _____ **Student #:** _____

Question 4: [5 marks]

Write a class named `CapitalsCounter` that implements a method named `countCaps` to return the number of capital letters within a `String`, as done in this python snippet (hint: use `Character.isUpperCase()`):

```
def countCaps(s):
    count = 0
    for character in s:
        if character.isupper():
            count = count + 1
    return count
```

Write a `main` method to check whether or not the `countCaps` method correctly returns 5 when given the the string `"IHaveFiveCaptialLetters"`. It should print `":)"` if successful, or `":("` if failed.

Name: _____ **NSID:** _____ **Student #:** _____

Question 5: [5 marks]

Write a class named `NumberGuesser` that implements the following code within its `main` method by using a `do` loop and the `java.util.Scanner` class:

```
guess = int(input('Guess a number between 1 and 100: '))
while guess < 1 or guess > 100:
     if guess < 1:
          # If guess was less than one, execute this block.
          print('Too low!')
     elif guess > 100:
          # Otherwise, if guess is larger than 100, do this block.
          print('Too high!')

     # ask for a new guess
     guess = int(input('Guess a number between 1 and 100: '))

print('That was a valid guess!')
```

Name: _____ **NSID:** _____ **Student #:** _____

Question 6: [10 marks]

Write a class named `PositiveEvenFinder` that implements a method named `findPositiveEvens` to return an `int` array containing all of the positive and even numbers from the provided `int` array, as done in this python snippet:

```
def positive_evens( numbers ):
    '''
    Purpose: Returns all the positive even numbers in the list numbers
    Pre: numbers: a list of integers
    Post: none
    Return: a list of positive even integers
    '''
    return [x for x in numbers if x % 2 == 0 and x > 0]
```

Use the provided `main` method to test your code – same as this python snippet:

```
inputs = [1,2,3,4,5]
result = positive_evens(inputs)
if result != [2, 4]:
    print("Testing fault: positive_evens() returned",result,"on inputs",inputs,
          "(didn't find the correct even numbers)")
```