2021F CS234 Computer Science II

Lab 3
Total points: 100

P4.9 (30 points) The Drunkard's Walk. A drunkard in a grid of streets randomly picks one of four directions and stumbles to the next intersection, then again **randomly** picks one of four directions, and so on. You might think that on average the drunkard doesn't move very far because the choices cancel each other out, but that is actually not the case.

Represent locations as integer pairs (x, y). **Implement the drunkard's walk** over **100** intersections, starting at (0, 0), and **print** the **initial** and **ending** location.
Hint: You can use direction 0 as going to the right, 1 as going to the left, 2 as going up, and 3 as going down.

P5.12. (70 points) Write a program that **prompts** the user for a regular English verb (such as play), the first, second, or third person, singular or plural, and present, past, or future tense. Provide these values to a method that yields the conjugated verb and prints it.
For example, the input ***play 3 singular present*** should print "***he/she plays***".

Use in your program the following method definitions:

```
/**
      Finds the pronoun corresponding to a person-number combination
      @param person either 1, 2, or 3
      @param number either singular or plural
      @return the corresponding pronoun
   */
public static String getPronoun(int person, String number)


/**
      Returns the conjugation of a regular verb with the given arguments
      @param verb a regular English verb
      @param person either 1, 2, or 3
      @param num either "singular" or "plural"
      @param tense either "past", "present", or "future"
      @return an appropriate conjugation of pronoun and verb phrase
   */
public static String conjugate(String verb, int person, String num, String
tense)
```

Submission details:

Upload a **single ZIP** file.
Name your file as follows: **Lab3_Lastname_Firstname.zip**
There is a **10% deduction** if your file does not have the correct name.

Your **.zip** file must contain the following:

1. Your **.java** source files (no .class).
2. A **.txt** file (e.g., readme.txt) with the instruction on how to **compile** and **execute** your programs. For example, *javac MY_FILE.java* and *java MY_FILE*.
3. A **single PDF** file with screenshots showing your programs running

In each .java file, **write** as a multiline comment at the beginning of the file the following:

1. Your name
2. The ID of the problem (e.g., P4.9)
3. The course section

The **zip** file must be uploaded to Canvas. I do not accept answers via email. I do not accept image files; it must be a PDF file with the screenshots.

Make sure to check the **due date** for this activity on Canvas.
Make sure you are **submitting the correct files**. I will grade the file uploaded to Canvas.
Make sure your program can be compiled and executed using the instructions from your .txt file.
Make sure to review the grading rubric.
Late submissions are not allowed.