

Specification for Assignment 5 for COMP1405 (Fall 2016)

Instructions

There are two questions in this assignment. For each question you will be submitting your solution as multiple files, each containing source code written in Python 3. These files must be compressed into a single "zip" file, and you will submit this file using cuLearn.

- ☐ *The source file for question 1 must be named "a5q1.py".*
- ☐ *The source file for question 2 must be named "a5q2.py".*
- ☐ *The compressed archive (i.e., the .zip file) must be named "a5.zip".*
- ☐ *The due date for this assignment is Saturday, December 3, 2016, by 11:30pm.*



Late assignments will be accepted for 48 hours after the deadline, but the penalty for submitting a late assignment is a loss of 2.0% per hour.



You are expected to demonstrate good programming practices at all times (e.g., choosing descriptive variable names, provide comments in your code, etc.) and your code will be penalized if it is poorly written.



You are expected to do the necessary preparatory work (e.g., devising an algorithm) before you begin coding. Whenever appropriate, you will be asked to present either pseudocode or a flowchart before you will receive any assistance from the instructor or a teaching assistant.



This assignment is uniquely generated; every student in the class is required to complete a slightly different version of this assignment. To ensure that each unique assignment shares the same level of difficulty, a unique assignment generator (and supporting files) has been made available on cuLearn.

To receive the assignment instructions that are specific to you, download the "unique-assignment-generator-for-A5.zip" file from cuLearn. Once you have extracted the contents to your working folder, use the command prompt to run the "generator-for-A5.py" program and then enter and confirm your student number.

Specification for Assignment 5 for COMP1405 (Fall 2016)**Questions 1 and 2 (of 2)**

When you run the unique assignment generator for this assignment, it will describe a unique recursive sequence. The sequence will be very much like one discussed during the 12th week of classes. The function presented by the generator will describe how to find the n^{th} element of your assigned sequence - there will be several "base cases" that will not require any computational effort to produce, but the remaining elements of the sequence will be described with respect to previous elements (in very much the same way as the n^{th} number in the Fibonacci sequence is described in terms of the two values that precede it).

As the generator will describe, you are expected to write two functions for this question. The first function you write will take one integer argument, n , and produce one integer return value, such that the return value is the n^{th} element of the sequence, and the second function you write will take one integer argument, n , and produce one list return value, such that the return value is the list of the first n elements of the sequence.

You will receive a mark of zero if you do not write these functions recursively and you will be penalized for inefficient solutions (so think carefully about the recursive calls you are making).

When you ran the unique assignment generator for this assignment, it also explained how "homoglyphs" can be used to write a word using distinct characters that still visually resembles the way it is typically written. The name 'ROBERT', for example, could be written using the sequence of homoglyphs, '12OB312+', where '12' is visually similar to 'R', '3' is visually similar to 'E', '+' is visually similar to 'T', etc. For this question you are asked to read an external file and perform the necessary string processing in order to have the data required to produce a small homoglyph dictionary that you can use for "translating" words provided by the user. You will also be required to write additional functions for interacting with the dictionary (the precise nature of which being described in detail by the assignment generator).

Before attempting to implement your submission for either question on this assignment you are REQUIRED to create either a flowchart or a pseudocode version of your solution. You do not submit these on cuLearn with the rest of your assignment but if you seek assistance from the instructor or a teaching assistant you will be asked to present this preparatory work, and if you have not completed these preparations you will not receive any assistance.