CSCI-UA.0101-002: Assignment 4 – Algorithmic thinking

Due Tuesday, October 31st at 11:59 p.m.

Instructions:

- The project directory folder for this assignment is called A4_project_directory_NYUnetID. Rename NYUnetID with your own NYU NetID. For example, I would rename my project folder "A4_project_directory_gp2442".

- The project directory contains a project directory containing four subdirectories, namely data, lib and src. The source files are in src/edu/nyu/cs/NetID. Make sure to rename the subdirectory /NetID to your actual NYU Net ID.

- Complete the code according to the instructions in this document as well as those written as comments within the .java source files (if any).

- **Important:** In addition to completing all problems, you are also expected to compile and run your source code using the Command line from the project directory. Refer to Lectures 3 and 4 for how to do this. You will lose five points if we cannot compile and run your compiled code from the project directory.

- Submit a zip file named "A4_complete_NYUnetID" containing your project folder called "A4_project_directory_NYUnetID". Again, NYUnetID should be replaced with your NYU NetID.

1. Reversing strings and numbers.

a) In the source file “Reversing.java”, write a Java method called myStringReverse that reverses a given string. The method should return a String. For example, suppose we have the string "Hello, NYU!". Your Java method should return the reversed string "!UYN ,olleH". **Note**: You may **not** use an in-build method for doing so.

b) In the source file Reversing.java, write a Java method called myIntegerReverse that reverses a given integer without using a string. The method should return an integer. For example, suppose we have the integer 123456789. Your Java method should return the reversed number 987654321. **Note:** You may **not** use an in-build method for doing so.

c) A palindrome is a word, number or phrase that reads the same backwards as forwards, such as “stressed desserts” or “1234321”. In the source file Reversing.java, write a java method called checkPalindrome that takes for input a string **or** an integer and returns true if the string is a palindrome and false otherwise. Do **NOT** take capitalization into account. You’ll have to write two different versions of checkPalindrome and use the concept of method overloading to account for the two different types of input. You may use the methods part a) and b) to help you do this.

2. Unique characters.

In the source file “UniqueCharacter.java”, write a Java method called firstUniqueChar that finds the position of the first unique character in a given string. The first unique character is the character that appears only once in the string, and its position is the first occurrence. Do **NOT** take capitalization into account. If there are no unique characters, then your method should return the value -1. For example, suppose you have the string "Divident". The method should return 2 because ’v’ is the first unique character, and it appears at position 2. If instead you have the string “redder”, then the method should return -1. Note: You may **not** use an in-build method for doing so.