



# National University of Sciences and Technology (NUST)

## School of Electrical Engineering and Computer Science

**Subject:** CS212 Object-Oriented Programming

**Due Date:** Thursday April 6, 2017

### Assignment # 01 (20 Points) - Individual Assignment

[CLO3] - Develop programs using object Oriented techniques.

**Note: You will upload your submission on LMS through the submission link (a single Zip file) and submit a hand-written copy during the class. Printed Assignments will not be accepted!**

**Question#1:** The explosive growth of Internet communications and data storage on Internet-connected computers has greatly increased privacy concerns. The field of cryptography is concerned with coding data to make it difficult (and hopefully—with the most advanced schemes—impossible) for unauthorized users to read. In this exercise you'll investigate a simple scheme for encrypting and decrypting data. A company that wants to send data over the Internet has asked you to write a program in Java that will encrypt it so that it may be transmitted more securely. All the data is transmitted as four-digit integers. Your application should read a four-digit integer entered by the user and encrypt it as follows:

- Replace each digit with the result of adding 7 to the digit and getting the remainder after dividing the new value by 10.
- Then swap the first digit with the third, and swap the second digit with the fourth. Then print the encrypted integer.

Write a separate application (in Java) that inputs an encrypted four-digit integer and decrypts it (by reversing the encryption scheme) to form the original number. [Optional reading project: Research “public key cryptography” in general and the PGP (Pretty Good Privacy) specific public key scheme. You may also want to investigate the RSA scheme, which is widely used in industrial-strength applications.]

**Question#2:** A prime number is any integer greater than 1 that's evenly divisible only by itself and 1. The Sieve of Eratosthenes is a method of finding prime numbers. It operates as follows:

- Create a primitive-type boolean array with all elements initialized to true . Array elements with prime indices will remain true . All other array elements will eventually be set to false .
- Starting with array index 2, determine whether a given element is true . If so, loop through the remainder of the array and set to false every element whose index is a multiple of the index for the element with value true . Then continue the process with the next element with value true . For array index 2, all elements beyond element 2 in the array that have indices which are multiples of 2 (indices 4, 6, 8, 10, etc.) will be set to



## National University of Sciences and Technology (NUST) School of Electrical Engineering and Computer Science

false ; for array index 3, all elements beyond element 3 in the array that have indices which are multiples of 3 (indices 6, 9, 12, 15, etc.) will be set to false ; and so on.

When this process completes, the array elements that are still true indicate that the index is a prime number. These indices can be displayed. Write an application in Java that uses an array of 1000 elements to determine and display the prime numbers between 2 and 999. Ignore array elements 0 and 1.