**Sabancı University**

Faculty of Engineering and Natural Sciences

**CS204 Advanced Programming**

**Fall 2018-2019**

**Homework 7 – Bitwise Operations**

**Due: 10/12/2018 - 23:55**

(One day late submission penalty: -10%)

|  |
| --- |
| **PLEASE NOTE:**  **Your program should be a robust one such that you have to consider all relevant programmer mistakes and extreme cases; you are expected to take actions accordingly!**  **You HAVE TO write down the code on your own.**  **You CANNOT HELP any friend while coding.**  **Plagiarism will not be tolerated!** |

1. **Introduction**

The aim of this homework is to practice bitwise operations. The goal is to implement an RSA encryption code. In RSA encryption, you need to use a lot of modular exponentiation. However, we ask for you to implement this using **bitwise** **operations** as well as the straightforward method (using arithmetic operations). The details of the homework is described in the document.

1. **Program Flow**

In RSA encryption, you need to do the followings for the initialization part (Constructor):

* Choose two distinct prime numbers **p** and **q**:
* Compute **n** = **pq**
* Compute = **(p-1) \* (q-1)**
* Choose a number **e** which is relatively prime with (you can use a GCD function that we provide)
* Choose a number **d** which satisfies **d \* e = 1 mod**

For encryption, you will get a message **m** which is an integer and compute **c = me (mod n)** where **c** is going to represent the *cyphertext*.

For decryption, you will get a cyphertext **c** which is an integerand compute **m = cd (mod n)** where m is going to represent the *plaintext*.

For more detail about the RSA encryption system, you can refer to [Wikipedia](http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvUlNBXyhjcnlwdG9zeXN0ZW0p).

In the given source file, you are given a code with empty functions. You can implement the **encrypt** and **decrypt** functions straightforwardly, i.e., using arithmetic operations and so on. However, since modular exponentiation takes too long to compute for large primes, we want you to also implement **decrypt\_bit** in which you are going to calculate **cd (mod n)** using bitwise operations.

To implement a bitwise modular exponentiation, you need to do the followings:

* Traverse the whole number from **MSB to LSB**.
* When you encounter **1,** multiply your result by **c** and take **(mod n)**
* When you encounter **0,** do nothing.
* For each bit, **EXCEPT THE LAST ONE**, take the power of your result and then take **(mod n)**.

When you implement these steps, the result should be the same as the input message. If not, there is a mistake. Also, please keep the following in mind:

* **p** and **q** are going to be 16-bits **unsigned** numbers. (unsigned because we want to use all 16 bits.)
* **d, e** and **n** are going to be a 32-bit **unsigned** numbers

In the given source file, you are supposed to define two data types by filling the following lines (at the top of the file)

**#define FIRSTDATATYPE // choose a data type here**

**#define SECONDDATATYPE // choose a data type here**

Finally, **THE BITWISE DECRYPTION MUST BE FASTER THAN THE STRAIGHTFORWARD IMPLEMENTATION**. Otherwise, you won’t get full grade. The whole purpose of using bitwise operations is to speed up the process. Note that you don’t need to implement anything for timing, it is already in the code.

**Some Important Rules:**

In order to get a full credit, your programs must be efficient and well presented, presence of any redundant computation or bad indentation, or missing, irrelevant comments are going to decrease your grades. You also have to use understandable identifier names, informative introduction and prompts. Modularity is also important; you have to use functions wherever needed and appropriate.

When we grade your homework, we pay attention to these issues. Moreover, in order to observe the real performance of your codes, we may run your programs in *Release* mode and **we may test your programs with very large test cases**.

**What and where to submit (PLEASE READ, IMPORTANT):** You should prepare (or at least test) your program using MS Visual Studio 2012 C++. We will use the standard C++ compiler and libraries of the abovementioned platform while testing your homework. It'd be a good idea to write your name and last name in the program (as a comment line of course).

Submissions guidelines are below. Some parts of the grading process are automatic. Students are expected to strictly follow these guidelines in order to have a smooth grading process. If you do not follow these guidelines, depending on the severity of the problem created during the grading process, 5 or more penalty points are to be deducted from the grade.

Name your cpp file that contains your program as follows:

***“SUCourseUserName\_YourLastname\_YourName\_HWnumber.cpp”***

Your SUCourse user name is actually your SUNet username that is used for checking sabanciuniv e-mails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name. For example, if your SUCourse user name is cago, name is Çağlayan, and last name is Özbugsızkodyazaroğlu, then the file name must be:

***Cago\_Ozbugsizkodyazaroglu\_Caglayan\_hw2.cpp***

Do not add any other character or phrase to the file name. Make sure that this file is the latest version of your homework program. Compress this cpp file using WINZIP or WINRAR programs. Please use "zip" compression. "rar" or another compression mechanism is NOT allowed. Our homework processing system works only with zip files. Therefore, make sure that the resulting compressed file has a zip extension. Check that your compressed file opens up correctly and it contains your cpp file.

You will receive no credits if your compressed zip file does not expand or it does not contain the correct file. The naming convention of the zip file is the same as the cpp file (except the extension of the file of course). The name of the zip file should be as follows:

***SUCourseUserName\_YourLastname\_YourName\_HWnumber.zip***

For example zubzipler\_Zipleroglu\_Zubeyir\_hw1.zip is a valid name, but

***Hw2\_hoz\_HasanOz.zip, HasanOzHoz.zip***

are **NOT** valid names.

**Submit via SUCourse ONLY!** You will receive no credits if you submit by other means (e-mail, paper, etc.).

Successful submission is one of the requirements of the homework. If, for some reason, you cannot successfully submit your homework and we cannot grade it, your grade will be 0.

Good Luck!

CS204 Team (Mustafa Kemal Taş, Kamer Kaya)