ANKARA UNIVERSITY

COM101B

Fall 2017-18 Term

LAB3 Sample – Take Home

Date: 18/10/2017

Due Date: 19/10/2017, 23:55

Learning Task:

AIM: Aim of this sample is to learn input redirection from files to console.

In order to learn this method you are given a simple programming task. Please perform the task specified below:

Task: Write a C program which reads a sequence of numbers from std. input and computes the squares of each number, until the EOF is reached. For this purpose, you will prepare sample input files such as input1.txt, input2.txt, etc. and test your program by input redirection: using the operator **<** from the console as shown below.

Assume that, you prepared a file named input1.txt which contains:
1 2 3 4 5

Assume that your executable filename is sample1, you can run your executable as follows:

```
>./sample1<input1.txt
```

This is an alternative way to provide input to your programs from stdin, although it is actually stored in a file. This is an input redirection method which is very helpful when we want to provide test data in a file for our programs, rather than typing the content on the console for each run.

For input1.txt, your program must generate:

```
1 4 9 16 25
```

Hint: Remember that **EOF** is a predefined macro which stands for End Of File. You can capture EOF using scanf function as follows:

```
int x;
while(scanf(“%d”,&x) != EOF)
{           <do domething>          }
```

We can redirect the output to another file, if we like. We use another redirection method (by using the operator **>**), which redirects the stdout into a file. As follows:

```
>./sample1<input1.txt>output1.txt
```

After this run, you will notice a file is created on the working directory, which is named **output1.txt.** Now, check the content of this file! You will notice that it is the output stream that your code intends to send to stdout.

Make sure that you completed the sample as described above.
Please, do not send this sample! You will be sending the following sample to us.

Programming Assignment:

Write a C program which reads a set of characters, until the EOF, from the standard input (stdin) using the **getchar() function** and prints the number of letters in the input and prints the consonants in capital letters.

Hint: There can be many white space characters in the input, such as blank, \n, \t etc. You should consider such cases while you are testing your output.

I/O format:

Input: **<set of characters>**

Output: **<# of letters read><SPACE><string of consonants in CAPITAL leters>**

Example:

Sample Input: a b   kL m   45 ’!   u a   m   P ?   %   E

There are 10 characters in the input stream {a, b , k, L, m, u, a, m, P, E}.

10 is your first output.

The set of consonants in the input stream is: {B, K, L, M, M, P}.

Your second output will be the string of consonants in capital letters, as shown below:

Sample Output: 10 BKLMMP

Warning: Your source codes will be scanned by a program for cheating. Any form of cheating is forbidden. Please write your own source codes and submit that!

Submission: Your source code will be named using your student ID: <student-id>.c

Ex: if your id is 1123456 than you should name your source code like this: 1123456.c

Please, PAY ATTENTION TO THE I/O FORMAT.

Good Luck ☺