Department of Computer Engineering

CENG104 – Computer Programming II

Spring 2017 - 2018

Lab Guide #5/A – Week 7

OBJECTIVE : Practice on Recursive functions, Binary Files

Instructor : Yusuf Evren AYKAÇ
Assistants : Elif GÜL, Yusuf Şevki GÜNAYDIN, Hatice ÇATALOLUK

1.

Some algorithms require nested recursion where the result of one function call is a parameter to another function call. For example Ackermann's function is defined as:

$$A(m,n) = \begin{cases} n+1 & \text{if } m = 0 \\ A(m-1,1) & \text{if } m > 0 \text{ and } n = 0 \\ A(m-1, A(m,n-1)) & \text{if } m > 0 \text{ and } n > 0. \end{cases}$$

Write a **recursive** function **Ackermann** that takes two integer numbers as parameters, finds and returns the result according to the given rules above.

Write a C program that gets two integer numbers from the user and computes the result of **isAckermann(m, n)**. The value of **m** and **n** both have got to be non-negative values (x >= 0).

Project_name: LabGuide5_1
File_name: Question_1.cpp

Example Run#1:

```
Enter the value of m: 3
Enter the value of n: 2
The result is 29
```

Example Run#2:

```
Enter the value of m: 2
Enter the value of n: -4
The value of both m & n had to be positive values. Exiting.
```

2.

A) Write a program that creates a binary file called "**numbers.bin**" with integers starting from 5 to 2000 and increasing as the power of 5. The program will then read n, where n is taken as input from the user, numbers from the "**numbers.bin**" binary file into an array of size n and displays the array content. Don't forget to validate n.

Hint: While reading the numbers you have to use a for or while loop to put the numbers into the file.

Example Run:

```
How many numbers do you want to read: 2001
Enter a number between 1 and 500!
How many numbers do you want to read: -5
Enter a number between 1 and 500!
How many numbers do you want to read: 48


 5   10   15   20   25   30   35   40   45   50   55   60   65   70   75   80
 85   90   95  100  105  110  115  120  125  130  135  140  145  150  155  160
165  170  175  180  185  190  195  200  205  210  215  220  225  230  235  240
```

Project_name: LabGuide5_2a
File_name: Question_2a.cpp

B) Modify your code from part A so that while reading the numbers the program does not use any loops.

Project_name: LabGuide5_2b
File_name: Question_2b.cpp

3.

Create a structure named **stu_t** with the fields **name**, **age**, and **GPA**. Declare a **stu_t** type array, and initialize it with the values in a file of **15** students from a text file named **info.txt**. Write a C program that writes the values in the **info.txt** to a binary file named **binary.bin.** Your program should display a menu to the screen with the options; "Go to record X from top", "Move X records ahead", "Go X records back from bottom", and "Exit" to reach data inside of the binary.bin file.

Example Run:

```
1) Go to record X from top
2) Move X records ahead
3) Go X records back from bottom
4) Exit
Enter your choice: 1
Enter X: 7
Burcu  25  2.65

1) Go to record X from top
2) Move X records ahead
3) Go X records back from bottom
4) Exit
Enter your choice: 2
Enter X: 3
Fikret 22  3.94

1) Go to record X from top
2) Move X records ahead
3) Go X records back from bottom
4) Exit
Enter your choice: 3
Enter X: 4
Sevinc 24  3.96

1) Go to record X from top
2) Move X records ahead
3) Go X records back from bottom
4) Exit
Enter your choice: 4
```

info.txt

Deniz	22	3.02
Hasan	23	2.90
Derya	24	3.58
Sinan	23	3.21
Ali	24	3.10
Alya	26	3.00
Burcu	25	2.65
Omer	26	3.12
Faruk	24	4.00
Fikret	22	3.94
Eray	23	2.45
Sevinc	24	3.96
Hale	26	2.21
Elif	21	3.78
Arzu	22	2.89

Project_name: LabGuide5_3
File_name: Question_3.cpp