MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS
CNG140 C Programming – Worksheet 12

Purpose:
In this work sheet, you will continue to familiarize yourself with functions. In particular, we will learn about iterative vs. recursive functions.

Tasks:

a. Write a C program that **recursively** computes and prints the binary representation of a positive integer value.

Sample Run:
```
Enter a positive integer: 24
Binary representation of 24 is 11000
```

b. Write a C program that **recursively** reverses the digits of a positive integer.

Sample Run:
```
Enter a positive integer: 1234
Reverse of the number is 4321
```

c. Write a **recursive** function that returns the value of the following recursive definition:

$$f(x) = 0 \qquad if\ x \leq 0$$
$$f(x) = f(x-1) + 2 \qquad otherwise$$

Sample Run:
```
Please enter a value of x=3
f(3) = 6
```

d. The following Dijkstra's algorithm can be used to find the GCD of two numbers:

$$For\ m, n > 0, \gcd(m,n) = \begin{cases} m & if\ m = n \\ \gcd(m-n, n) & if\ m > n \\ \gcd(m, n-m) & if\ m < n \end{cases}$$

Write a **recursive** function that can be used to find the GCD of two numbers.

Sample Run:
```
Enter m and n: 4 8
GCD of 4 and 8 is 4
```

e. The Legendre polynomials can be calculated by means of the formulas $P_0 = 1,\ P_1 = x, and$

$$P_n = \left[\frac{2n-1}{n}\right] P_{n-1} - \left[\frac{n-1}{n}\right] P_{n-2}$$

Where n=2, 3, 4… and x is any floating-point number between -1 and 1 (Note that the Legendre polynomials are floating-point quantities). Generate the first n Legendre polynomials. Let the values of n and x be the input parameters. Please make sure that you check if x is given a valid input (between -1 and 1). You need to write at least one **recursive** function to solve this problem.

Sample Run:

```
Please enter x [-1,1]= 2
This x value is not valid!
Please enter x [-1,1]= 0.5
```

```
Please enter n= -2
This n value is not valid!
Please enter n= 5
The first 5 are as follows
x=0.500000      n=1     P(1)=0.500000
x=0.500000      n=2     P(2)=0.250000
x=0.500000      n=3     P(3)=0.083333
x=0.500000      n=4     P(4)=-0.041667
x=0.500000      n=5     P(5)=-0.141667
```

Please note that precision could be different on your machine!

f. You need to write a program that analyses the number of rabbit grows in a farm. Your program needs to get the initial number of rabbits in the farm, the number of months the analysis will be done for and then it will report the number of estimated rabbits in the farm at the end of the number of months entered, and it will also make an assessment of the growth. Your program will have two functions to complete this task:

- `int growth_population(int m, int initial_rabbits)`: Assume that rabbit population grows with 25% each month and the number of rabbits in the current month compared to the previous month is:

  growth_population(m) = growth_population(m-1) + 0.25 * growth_population(m-1)

  Write a **recursive** function that computes the total number of rabbits at the end of month, m with the initial number of rabbits given to the function. Please note that in month 0, your total number of rabbits is the initial_rabbits given to the function.

- `void analysis(int rabbits)`: This function takes the number of rabbits and analyse them based on the following table, and displays the necessary message on the output.

Number of rabbits	**Message**
<100	Hobby farm
100-200	Small farm
201-500	Medium farm
501-1000	Big farm
>1000	Huge farm

Sample Run:

```
Enter initial number of rabbits: 0
Sorry you cannot enter 0!
Enter initial number of rabbits: 100
Number of months: 5
At the end of month 5, rabbit population will be 303
Your farm will be a medium farm!
```

Recommended Reading: Chapter 7 (p. 328-342 && 364-372)
Recommended Exercises: Exercises 7.1, 7.2 and 7.5.