Data Structures

Lecturer: Dr. Wael Zakaria, Dr. Ghada NourEldeen

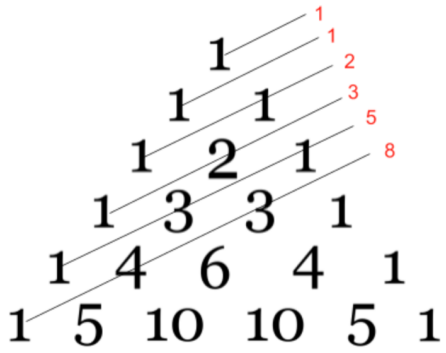
LAB 1

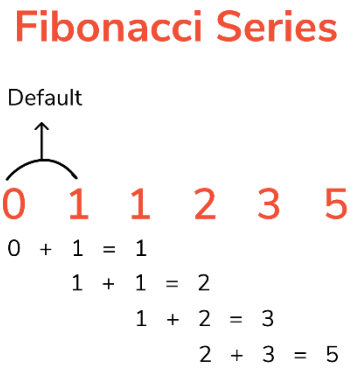
1. Write a program that take three variables (a, b, c) in as separate parameters and rotates the values stored so that value of a goes to b, b goes to c, and c goes to a.

```
void swap(int &x, int &y){
     int temp = x;
     x = y;
     y = temp;
}
void rotate(int &a, int &b, int &c){
     swap(a, b);
     swap(b, c);
     swap(c, a);
}
int main()
{
   int a = 1, b = 2, c = 3;
   rotate(a, b, c);
   cout << " a = " << a << ", "
        << " b = " << b << ", "
         << " c = " << c << endl;

   system("pause");
   return 0;
}
```

2. Write a program to find the Fibonacci series of n where $n \geq 2$.
$F_0 = 0$, $F_1 = 1$ and $F_n = F_{n-1} + F_{n-2}$.






```
int Fib(int n){
     if(n == 0) return 0;
     if(n == 1) return 1;
     return Fib(n-1) + Fib(n-2);
}
void main(){
     int n;
     cout << "n= "; cin >> n;
     cout << "Fib(" << n <<") = " << Fib(n) << endl;
     system("pause");
}
```

3. Write a program to simulate a supermarket.
   Create a struct for products (product name, product number, price).
   Write a function to read 10 products using pointers.
   Write a function to display 10 products using pointers.
   Write a recursive function to sort the 10 products in ascending order according the price.

```
#include<iostream>
#include <string>
using namespace std;

struct product{
     string name;
     int number;
     float price;
};
void read(product *s[], int size)
{
     for(int i=0 ; i<size; i++)
     {
          s[i] = new product();
          cout << "Enter product name" << endl;
          cin >> s[i]->name;
          cout << "Enter product number" << endl;
          cin >> s[i]->number;
          cout << "Enter product price" << endl;
          cin >> s[i]->price;
     }
}
void display(product *s[], int size)
{
     for(int i=0; i<size; i++)
     {
          cout << "product name is " << s[i]->name << endl;
          cout << "product number is " << s[i]->number << endl;
          cout << "product price is " << s[i]->price << endl;
     }
}
```

```
void Sort(product *arr[], int n)
{
    int count = 0;
    for (int i=0; i<n-1; i++)
     {
          if (arr[i]->price > arr[i+1]->price){
             swap(arr[i], arr[i+1]);
             count++;
         }
     }
     //in case the array is sorted
    if (count == 0)
        return;

    Sort(arr, n-1);
}
void main()
{
     int n = 3;
    product *p = new product[n];

     read(&p, n);

     cout<<"products befor sorting : "<<endl;
    display(&p, n);

     Sort(&p, n);
     cout<<"products after sorting : "<<endl;
     display(&p, n);


     system("pause");
}
// a 3 15 b 3 10 c 3 5
```