COP 2535: Data Structures

Lab 00, what

1. Read pages 98 – 105 in Mastering Algorithms with C
2. Implement the following program.
3. Upload the output of your execution as text.

```
// https://www.codewhoop.com/stack/stack-using-linked-list.html
#include <iostream>
using namespace std;

//Structure of the Node
struct Node
{
    int data;
    Node* link;
};

// top pointer to keep track of the top of the stack
Node* top = NULL;

//Function to check if stack is empty or not
bool isempty()
{
    if (top == NULL)
        return true; else
        return false;
}

//Function to insert an element in stack
void Push(int value)
{
    Node* ptr = new Node();
    ptr->data = value;
    ptr->link = top;
    top = ptr;
}

//Function to delete an element from the stack
void Pop()
{
    if (isempty())
        cout << "Stack is Empty";
    else
    {
        //int data = top->data;
        cout << "Pop, top data is: " << top->data;
```

```
        Node* ptr = top;
        top = top->link;
        delete(ptr);
    }
}

// Function to show the element at the top of the stack
void Peek()
{
    if (isempty())
        cout << "Stack is Empty";
    else
        cout << "Element at top is : " << top->data;
}

// Function to Display the stack
void Display()
{
    if (isempty())
        cout << "Stack is Empty";
    else
    {
        Node* temp = top;
        while (temp != NULL)
        {
            cout << temp->data << " ";
            temp = temp->link;
        }
        cout << "\n";
    }
}

// Main function
int main()
{

    int choice, flag = 1, value;

    //Menu Driven Program using Switch
    while (flag == 1)
    {
        cout << "\n1.Push 2.Pop 3.Peek 4.Display 5.Exit\n";
        cin >> choice;
        switch (choice)
        {
        case 1: cout << "Enter Value:\n";
            cin >> value;
            Push(value);
            break;
        case 2: Pop();
            break;
        case 3: Peek();
            break;
        case 4: Display();
```

```
            break;
        case 5: flag = 0;
            break;
        }
    }

    return 0;
}
```