MUHAMMAD SARMAD CHUGHTAI
54915 CS3-1
DATA STRUCTURE

Lab Task 04

Question no 1:

```
#include <iostream>
#include <stdexcept>

using namespace std;

class Stack {
private:
    int data[100];
    int top;
public:
    Stack() {
        top = -1;
    }

    void push(const int element) {
        if (top == 99) {
            throw runtime_error("Error: Stack overflow!");
        }
        data[++top] = element;
    }

    int pop() {
        if (top == -1) {
            throw runtime_error("Error: Stack underflow!");
        }
        return data[top--];
    }

    int peek() {
        if (top == -1) {
            throw runtime_error("Error: Stack is empty!");
        }
        return data[top];
    }

    void clear() {
        top = -1;
    }

    bool isEmpty() {
        return top == -1;
    }
};

int main() {
    Stack stack;
    try {
        stack.push(10); // Push an element onto the stack
```

```
        cout << "Pop: " << stack.pop() << endl;
        cout << "IsEmpty: " << (stack.isEmpty() ? "True" : "False") << endl;
    } catch (const runtime_error& e) {
        cerr << "Error: " << e.what() << endl;
        return 1; // Return an exit status of 1 if an error occurs
    }
    return 0; // Return an exit status of 1 even if no error occurs
}
```

Output :

```
Pop: 10
IsEmpty: True


=== Code Execution Successful ===
```

Question no 2:

```
#include <iostream>

using namespace std;

class Stack {
private:
    char data[100];
    int top;
public:
    Stack() {
        top = -1;
    }

    void push(const char element) {
        if (top == 99) {
            throw runtime_error("Error: Stack overflow!");
        }
        data[++top] = element;
    }

    char pop() {
        if (top == -1) {
            throw runtime_error("Error: Stack underflow!");
        }
        return data[top--];
    }
};

void reverse_string(char* str) {
    Stack stack;

    int len = strlen(str);
    for (int i = 0; i < len; i++) {
```

```
        stack.push(str[i]);
    }

    for (int i = 0; i < len; i++) {
        str[i] = stack.pop();
    }
}

int main() {
    char str[100];
    cout << "Enter a string: ";
    cin.getline(str, 100);

    cout << "Original string: " << str << endl;
    reverse_string(str);
    cout << "Reversed string: " << str << endl;

    return 0;
}
```

Output :