```
#include<iostream>
const size_t maxCap= 100;
int stack[maxCap]; //stack with max of 100 elements
int top = -1, i, newData;
void push();
void pop();
void Top();
bool isEmpty();
void Display();
int main(){
int choice;
std::cout << "Enter number of max elements for new stack: ";
std::cin >> i;
while(true){
std::cout << "Stack Operations: " << std::endl;
std::cout << "1. PUSH, 2. POP, 3. TOP, 4. isEMPTY, 5. DISPLAY" <<
std::endl;
std::cin >> choice;
switch(choice){
case 1: push();
break;
case 2: pop();
break;
case 3: Top();
break;
case 4: std::cout << isEmpty() << std::endl;
break;
case 5: Display();
break;
default: std::cout << "Invalid Choice." << std::endl;
break;
}
}
return 0;
}
bool isEmpty(){
if(top==-1) return true;
return false;
}
void push(){
```

```
//check if full -> if yes, return error
if(top == i-1){
std::cout << "Stack Overflow." << std::endl;
return;
}
std::cout << "New Value: " << std::endl;
std::cin >> newData;
stack[++top] = newData;
}
void pop(){
//check if empty -> if yes, return error
if(isEmpty()){
std::cout << "Stack Underflow." << std::endl;
return;
}
//display the top value
std::cout << "Popping: " << stack[top];
//decrement top value from stack
top--;
}
void Top(){
if(isEmpty()) {
std::cout << "Stack is Empty." << std::endl;
return;
}
std::cout << "The element on the top of the stack is " << stack[top] <<
std::endl;
}
void Display(){
    if(isEmpty()) {
        std::cout << "Stack is Empty." << std::endl;
        return;
    }
    std::cout << "Stack elements: ";
    for(int j = top; j >= 0; j--) {
        std::cout << stack[j] << " ";
    }
    std::cout << std::endl;

}
```