BÀI TẬP THỰC HÀNH CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT

LAB MANUAL

Academic Year	**:**	**2022 - 2023**
Semester	**:**	**I**

Prepared by

MSc. KhangVQH

S. No.	Experiment
1	**SEARCHING TECHNIQUES**
2	**SORTING TECHNIQUES**
3	**SORTING TECHNIQUES**
4	**IMPLEMENTATION OF STACK AND QUEUE**
5	**APPLICATIONS OF STACK**
6	**IMPLEMENTATION OF SINGLE LINKED LIST**
7	**IMPLEMENTATION OF DOUBLE LINKED LIST**
8	**IMPLEMENTATION OF STACK USING LINKED LIST**
9	**IMPLEMENTATION OF QUEUE USING LINKED LIST**
10	**IMPLEMENTATION OF BINARY SEARCH TREE**

WEEK- 4

IMPLEMENTATION OF STACK AND QUEUE

4.1 OBJECTIVE:

a. Write a C program to implement Stack and its operations using Arrays.
b. Write a C program to implement Queue and its operations using Arrays.

4.2 PROGRAM LOGIC:

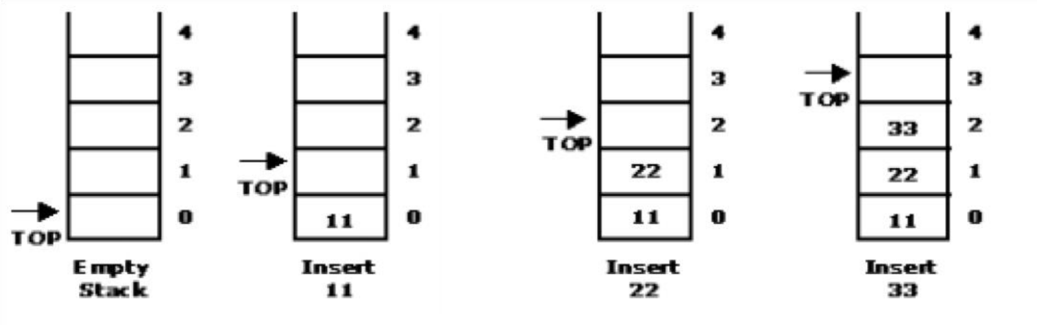
Procedure for Stack using Arrays

1. STACK: Stack is a linear data structure which works under the principle of last in first out. Basic operations: push, pop, display.
2. PUSH: if (top==MAX), display Stack overflow. Otherwise reading the data and making stack [top] =data and incrementing the top value by doing top++.
3. Pop: if (top==0), display Stack underflow. Otherwise printing the element at the top of the stack and decrementing the top value by doing the top.
4. DISPLAY: If (top==0), display Stack is empty. Otherwise printing the elements in the stack from stack [0] to stack [top].

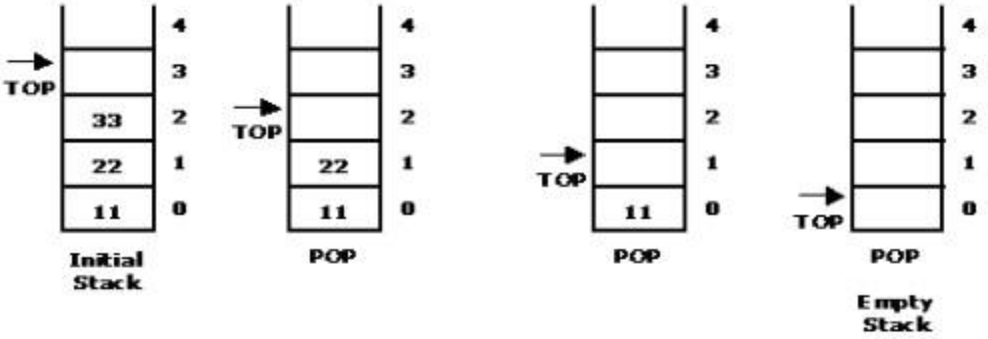
Procedure for Queue using Arrays

1. QUEUE: Queue is a linear data structure which works under the principle of first in first out. Basic operations: Insertion, deletion, display.
2. Inserion: if (rear==MAX), display Queue is full. Else reading data and inserting at queue [rear], and doing rear++.
3. Deletion: if (front==rear), display Queue is empty .Else printing element at queue [front] and doing front++.
4. Display: if (front==rear) ,display No elements in the queue .Else printing the elements from queue[front] to queue[rear].

Example: Consider a stack with 5 elements capacity. When an element is added to a stack, the operation is performed by Push().

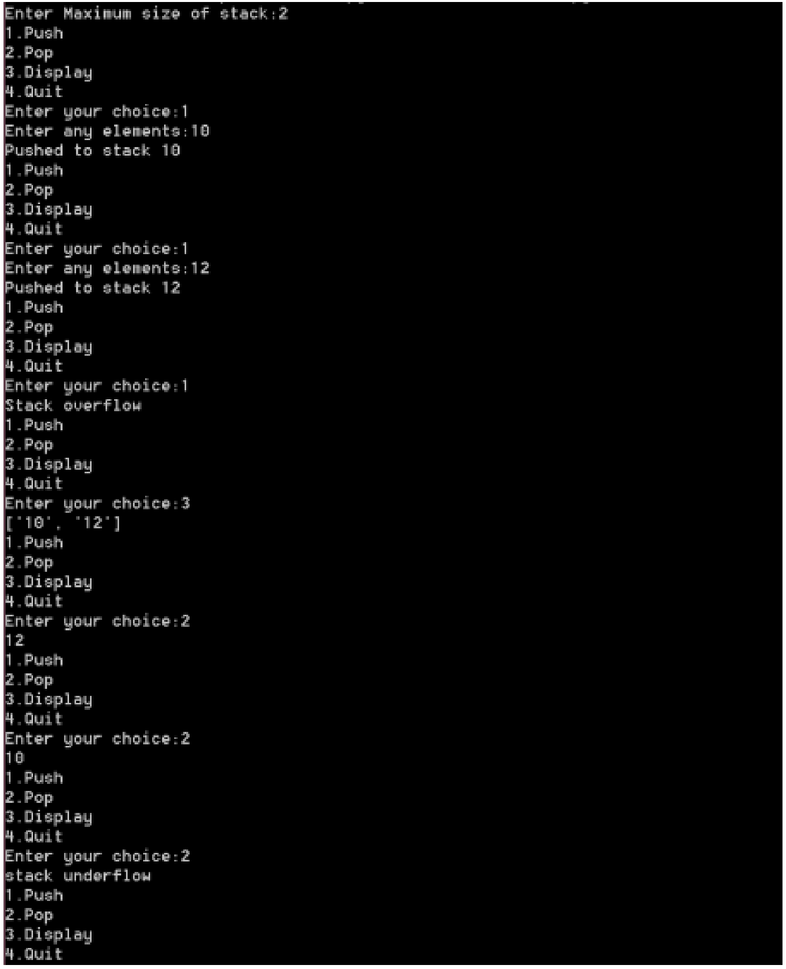


When an element is taken off from the stack, the operation is performed by Pop().



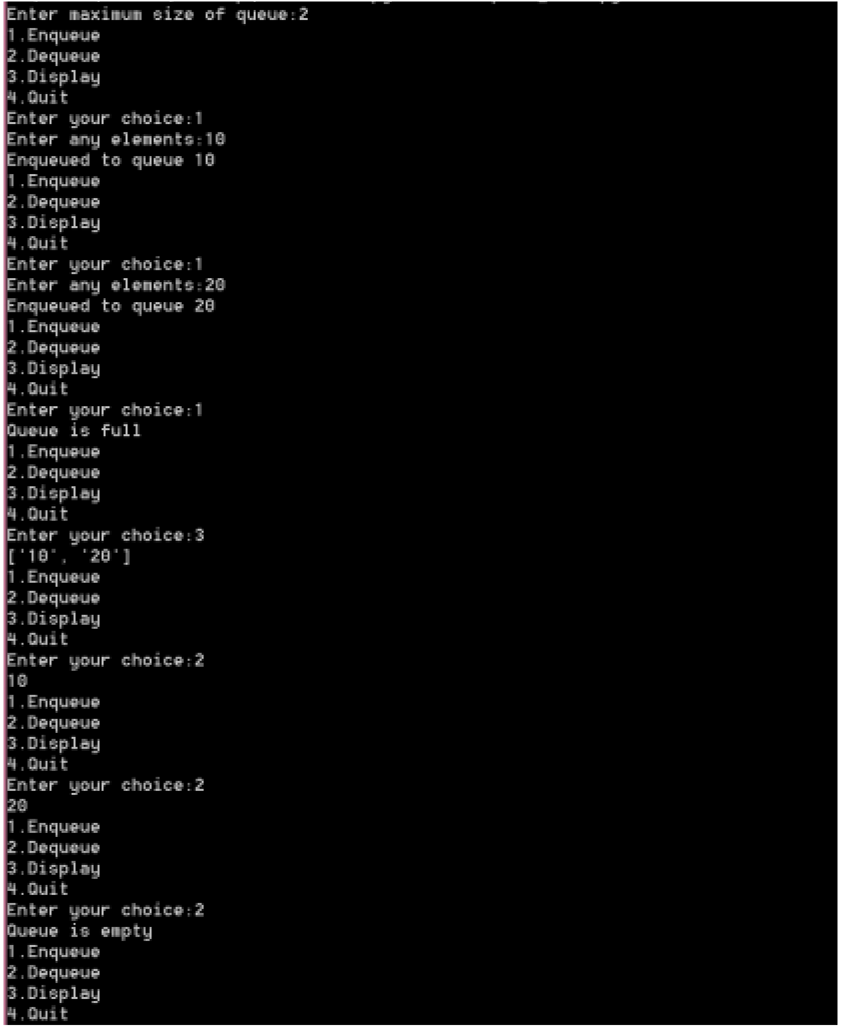
4.3 IMPLEMENTATION:

- **Implementation Stack using Arrays**
  Output:



```
Enter Maximum size of stack:2
1.Push
2.Pop
3.Display
4.Quit
Enter your choice:1
Enter any elements:10
Pushed to stack 10
1.Push
2.Pop
3.Display
4.Quit
Enter your choice:1
Enter any elements:12
Pushed to stack 12
1.Push
2.Pop
3.Display
4.Quit
Enter your choice:1
Stack overflow
1.Push
2.Pop
3.Display
4.Quit
Enter your choice:3
['10', '12']
1.Push
2.Pop
3.Display
4.Quit
Enter your choice:2
12
1.Push
2.Pop
3.Display
4.Quit
Enter your choice:2
10
1.Push
2.Pop
3.Display
4.Quit
Enter your choice:2
stack underflow
1.Push
2.Pop
3.Display
4.Quit
```

- **Implementation Linear Queue using Arrays**



```
Enter maximum size of queue:2
1.Enqueue
2.Dequeue
3.Display
4.Quit
Enter your choice:1
Enter any elements:10
Enqueued to queue 10
1.Enqueue
2.Dequeue
3.Display
4.Quit
Enter your choice:1
Enter any elements:20
Enqueued to queue 20
1.Enqueue
2.Dequeue
3.Display
4.Quit
Enter your choice:1
Queue is full
1.Enqueue
2.Dequeue
3.Display
4.Quit
Enter your choice:3
['10', '20']
1.Enqueue
2.Dequeue
3.Display
4.Quit
Enter your choice:2
10
1.Enqueue
2.Dequeue
3.Display
4.Quit
Enter your choice:2
20
1.Enqueue
2.Dequeue
3.Display
4.Quit
Enter your choice:2
Queue is empty
1.Enqueue
2.Dequeue
3.Display
4.Quit
```

4.4 LAB ASSIGNMENT

1. **Write a program to implement stack and its operations using arrays.**
2. Formulate a program to reverse a list of numbers using stack. (đảo ngược mảng – dùng stack)
3. Write a program to convert a number from decimal to binary using stack. (thập phân -> nhị phân)
4. Write a program to find the factorial of a number using stack. (tính giai thừa – dùng stack)
5. Develop a program to check for balanced brackets in an expression using stack. (kiểm tra dãy ngoặc đúng của biểu thức)
6. **Compose a program to implement Queue operations using arrays.**

4.5 POST LAB VIVA QUESTIONS:

1. Write the time complexity of PUSH operation?

2. Write the time complexity of POP operation?
3. List out the applications of stack?
4. How to remove an element from stack?
5. How to insert an element into a stack?
6. Write the time complexity to insert an element into a queue?
7. Write the time complexity to delete an element from a queue?
8. Define DEQUE?

S = (()())(())

- Duyệt xâu S từ trái qua phải
- Nếu ký tự đang xét là dấu (, thêm nó vào Stack
- Nếu ký tự đang xét là dấu), xét phần tử đang ở đỉnh Stack. Nếu đó là dấu ((, ta tìm được một cặp dấu ngoặc tương ứng, và loại bỏ dấu ngoặc mở khỏi Stack. Nếu ngược lại, hoặc Stack rỗng, xâu SS không phải dãy ngoặc đúng.
- Sau khi thực hiện quá trình, nếu Stack không rỗng thì SS không phải dãy ngoặc đúng.