

# Rajalakshmi Engineering College

Name: sakthivel M  
Email: 241901095@rajalakshmi.edu.in  
Roll no: 241901095  
Phone: 9381167405  
Branch: REC  
Department: I CSE (CS) FB  
Batch: 2028  
Degree: B.E - CSE (CS)

Scan to verify results



## NeoColab\_REC\_CS23231\_DATA STRUCTURES

### REC\_DS using C\_Week 3\_COD\_Question 2

Attempt : 1  
Total Mark : 10  
Marks Obtained : 1

#### Section 1 : Coding

##### 1. Problem Statement

Sanjeev is in charge of managing a library's book storage, and he wants to create a program that simplifies this task. His goal is to implement a program that simulates a stack using an array.

Help him in writing a program that provides the following functionality:

Add Book ID to the Stack (Push): You can add a book ID to the top of the book stack. Remove Book ID from the Stack (Pop): You can remove the top book ID from the stack and display its details. If the stack is empty, you cannot remove any more book IDs. Display Books ID in the Stack (Display): You can view the books ID currently on the stack. Exit the Library: You can choose to exit the program.

##### **Input Format**

The input consists of integers corresponding to the operation that needs to be performed:

Choice 1: Push the book onto the stack. If the choice is 1, the following input is a space-separated integer, representing the ID of the book to be pushed onto the stack.

Choice 2: Pop the book ID from the stack.

Choice 3: Display the book ID in the stack.

Choice 4: Exit the program.

### **Output Format**

The output displays messages according to the choice and the status of the stack:

1. If the choice is 1, push the given book ID to the stack and display the corresponding message.
2. If the choice is 2, pop the book ID from the stack and display the corresponding message.
3. If the choice is 2, and if the stack is empty without any book ID, print "Stack Underflow"
4. If the choice is 3, print the book IDs in the stack.
5. If the choice is 3, and there are book IDs in the stack, print "Stack is empty"
6. If the choice is 4, exit the program and display the corresponding message.
7. If any other choice is entered, print "Invalid choice"

Refer to the sample output for the exact text and format.

### **Sample Test Case**

Input: 1 19

1 28

2

3

2

4

Output: Book ID 19 is pushed onto the stack

Book ID 28 is pushed onto the stack

Book ID 28 is popped from the stack  
Book ID in the stack: 19  
Book ID 19 is popped from the stack  
Exiting the program

**Answer**

```
// You are using GCC
#include<stdio.h>
#include<stdlib.h>
typedef struct node{
    int data;
    struct node*next;
}node;
node*top=NULL;

void push(int value){
    printf("Book ID %d is pushed onto the stack\n",value);
    node*newnode=(node*)malloc(sizeof(node));
    newnode->data=value;
    newnode->next=top;
    top=newnode;
}

void pop(){
    if(top==NULL){
        printf("Stack Underflow\n");
        return;
    }else{
        node*temp=top;
        top=top->next;
        printf("Book ID %d is popped from the stack\n",temp->data);
        free(temp);
    }
}

void display(){
    if(top==NULL){
        printf("Stack is empty");
        return;
    }else{
        printf("Book ID in the stack: ");
        node*temp=top;
        while(temp!=NULL){
            printf("%d",temp->data);
```

```

        temp=temp->next;
    }
    }
    printf("\n");
}
int main(){
    int choice;
    do{
        scanf("%d",&choice);

        switch(choice){
            case 1:
                int val;
                scanf("%d",&val);
                break;
            case 2:
                pop();
                break;
            case 3:
                display();
                break;
            case 4:
                printf("Exiting the program");
                break;
            default:
                printf("Invalid choice\n");
        }
    }while(choice!=4);
    return 0;
}

```

**Status :** Partially correct

**Marks :** 1/10

# Rajalakshmi Engineering College

Name: sakthivel M  
Email: 241901095@rajalakshmi.edu.in  
Roll no: 241901095  
Phone: 9381167405  
Branch: REC  
Department: I CSE (CS) FB  
Batch: 2028  
Degree: B.E - CSE (CS)

Scan to verify results



## NeoColab\_REC\_CS23231\_DATA STRUCTURES

### REC\_DS using C\_Week 3\_COD\_Question 1

Attempt : 2  
Total Mark : 10  
Marks Obtained : 0

#### Section 1 : Coding

##### 1. Problem Statement

In a coding competition, you are assigned a task to create a program that simulates a stack using a linked list.

The program should feature a menu-driven interface for pushing an integer to stack, popping, and displaying stack elements, with robust error handling for stack underflow situations. This challenge tests your data structure skills.

##### ***Input Format***

The input consists of integers corresponding to the operation that needs to be performed:

Choice 1: Push the integer value onto the stack. If the choice is 1, the following input is a space-separated integer, representing the element to be pushed onto

the stack.

Choice 2: Pop the integer from the stack.

Choice 3: Display the elements in the stack.

Choice 4: Exit the program.

### ***Output Format***

The output displays messages according to the choice and the status of the stack:

If the choice is 1, push the given integer to the stack and display the following:  
"Pushed element: " followed by the value pushed.

If the choice is 2, pop the integer from the stack and display the following:  
"Popped element: " followed by the value popped.

If the choice is 2, and if the stack is empty without any elements, print "Stack is empty. Cannot pop."

If the choice is 3, print the elements in the stack: "Stack elements (top to bottom): " followed by the space-separated values.

If the choice is 3, and there are no elements in the stack, print "Stack is empty".

If the choice is 4, exit the program and display the following: "Exiting program".

If any other choice is entered, print "Invalid choice".

Refer to the sample input and output for the exact format.

**Sample Test Case**

Input: 1 3

1 4

3

2

3

4

Output: Pushed element: 3

Pushed element: 4

Stack elements (top to bottom): 4 3

Popped element: 4

Stack elements (top to bottom): 3

Exiting program

**Answer**

-

**Status :** Skipped

**Marks : 0/10**

# Rajalakshmi Engineering College

Name: sakthivel M  
Email: 241901095@rajalakshmi.edu.in  
Roll no: 241901095  
Phone: 9381167405  
Branch: REC  
Department: I CSE (CS) FB  
Batch: 2028  
Degree: B.E - CSE (CS)

Scan to verify results



## NeoColab\_REC\_CS23231\_DATA STRUCTURES

### REC\_DS using C\_Week 3\_MCQ\_Updated

Attempt : 1  
Total Mark : 20  
Marks Obtained : 15

#### Section 1 : MCQ

1. A user performs the following operations on stack of size 5 then which of the following is correct statement for Stack?

```
push(1);  
pop();  
push(2);  
push(3);  
pop();  
push(2);  
pop();  
pop();  
push(4);  
pop();  
pop();  
push(5);
```



**Answer**

Underflow Occurs

**Status :** Correct

**Marks :** 1/1

2. Pushing an element into the stack already has five elements. The stack size is 5, then the stack becomes

**Answer**

Underflow

**Status :** Wrong

**Marks :** 0/1

3. What will be the output of the following code?

```
#include <stdio.h>
#define MAX_SIZE 5
int stack[MAX_SIZE];
int top = -1;
int isEmpty() {
    return (top == -1);
}
int isFull() {
    return (top == MAX_SIZE - 1);
}
void push(int item) {
    if (isFull())
        printf("Stack Overflow\n");
    else
        stack[++top] = item;
}
int main() {
    printf("%d\n", isEmpty());
    push(10);
    push(20);
    push(30);
    printf("%d\n", isFull());
    return 0;
}
```

}

**Answer**

10

**Status :** Correct

**Marks :** 1/1

4. Which of the following Applications may use a Stack?

**Answer**

All of the mentioned options

**Status :** Correct

**Marks :** 1/1

5. In the linked list implementation of the stack, which of the following operations removes an element from the top?

**Answer**

Pop

**Status :** Correct

**Marks :** 1/1

6. What is the advantage of using a linked list over an array for implementing a stack?

**Answer**

Linked lists can dynamically resize

**Status :** Correct

**Marks :** 1/1

7. Consider a linked list implementation of stack data structure with three operations:

push(value): Pushes an element value onto the stack.  
pop(): Pops the top element from the stack.  
top(): Returns the item stored at the top of the stack.

Given the following sequence of operations:

```
push(10);pop();push(5);top();
```

What will be the result of the stack after performing these operations?

**Answer**

The top element in the stack is 5

**Status :** Correct

**Marks :** 1/1

8. What will be the output of the following code?

```
#include <stdio.h>
#define MAX_SIZE 5
void push(int* stack, int* top, int item) {
    if (*top == MAX_SIZE - 1) {
        printf("Stack Overflow\n");
        return;
    }
    stack[++(*top)] = item;
}
int pop(int* stack, int* top) {
    if (*top == -1) {
        printf("Stack Underflow\n");
        return -1;
    }
    return stack[(*top)--];
}
```

```
int main() {
    int stack[MAX_SIZE];
    int top = -1;
    push(stack, &top, 10);
    push(stack, &top, 20);
    push(stack, &top, 30);
    printf("%d\n", pop(stack, &top));
    printf("%d\n", pop(stack, &top));
    printf("%d\n", pop(stack, &top));
    printf("%d\n", pop(stack, &top));
    return 0;
}
```

}

**Answer**

302010Stack Underflow

**Status : Wrong**

**Marks : 0/1**

9. What will be the output of the following code?

```
#include <stdio.h>
#define MAX_SIZE 5
int stack[MAX_SIZE];
int top = -1;
void display() {
    if (top == -1) {
        printf("Stack is empty\n");
    } else {
        printf("Stack elements: ");
        for (int i = top; i >= 0; i--) {
            printf("%d ", stack[i]);
        }
        printf("\n");
    }
}
void push(int value) {
    if (top == MAX_SIZE - 1) {
        printf("Stack Overflow\n");
    } else {
        stack[++top] = value;
    }
}
int main() {
    display();
    push(10);
    push(20);
    push(30);
    display();
    push(40);
    push(50);
```

```
push(60);  
display();  
return 0;  
}
```

**Answer**

Stack is empty  
Stack elements: 30 20 10  
Stack Overflow  
Stack elements: 50 40 30 20 10

**Status :** Correct

**Marks :** 1/1

10. When you push an element onto a linked list-based stack, where does the new element get added?

**Answer**

At the end of the list

**Status :** Wrong

**Marks :** 0/1

11. The user performs the following operations on the stack of size 5 then at the end of the last operation, the total number of elements present in the stack is

```
push(1);  
pop();  
push(2);  
push(3);  
pop();  
push(4);  
pop();  
pop();  
push(5);
```

**Answer**

1

**Status :** Correct

**Marks :** 1/1

12. What is the primary advantage of using an array-based stack with a fixed size?

**Answer**

Efficient memory usage

**Status : Correct**

**Marks : 1/1**

13. In a stack data structure, what is the fundamental rule that is followed for performing operations?

**Answer**

Last In First Out

**Status : Correct**

**Marks : 1/1**

14. Consider the linked list implementation of a stack.  
Which of the following nodes is considered as Top of the stack?

**Answer**

Last node

**Status : Wrong**

**Marks : 0/1**

15. Which of the following operations allows you to examine the top element of a stack without removing it?

**Answer**

Peek

**Status : Correct**

**Marks : 1/1**

16. Elements are Added on \_\_\_\_\_ of the Stack.

**Answer**

Top

Status : Correct

Marks : 1/1

17. Here is an Infix Expression:  $4+3*(6*3-12)$ . Convert the expression from Infix to Postfix notation. The maximum number of symbols that will appear on the stack AT ONE TIME during the conversion of this expression?

Answer

3

Status : Wrong

Marks : 0/1

18. The result after evaluating the postfix expression  $10\ 5\ +\ 60\ 6\ /\ *8\ -$  is

Answer

142

Status : Correct

Marks : 1/1

19. What is the value of the postfix expression  $6\ 3\ 2\ 4\ +\ -\ *?$

Answer

-18

Status : Correct

Marks : 1/1

20. In an array-based stack, which of the following operations can result in a Stack underflow?

Answer

Popping an element from an empty stack

Status : Correct

Marks : 1/1