

DATASTRUCTURES

26/07/24

1. Stack elements in array

```
#include <stdio.h>

#include <stdlib.h>

#include <limits.h>

#define MAX 100

typedef struct
{
    int arr[MAX];
    int top;
}
Stack;

void initStack(Stack *s)
{
    s->top = -1;
}

int isEmpty(Stack *s)
{
    return s->top == -1;
}
```

```
int isFull(Stack *s)
{
    return s->top == MAX - 1;
}

void push(Stack *s, int value)
{
    if (isFull(s))
    {
        printf("Stack overflow\n");
        return;
    }
    s->arr[++(s->top)] = value;
    printf("%d pushed to stack\n", value);
}

if (isEmpty(s))
{
    printf("Stack underflow\n");
    return INT_MIN;
}
return s->arr[(s->top)--];
}
```

```
int peek(Stack *s)
{
    if (isEmpty(s))
    {
        printf("Stack is empty\n");
        return INT_MIN;
    }
    return s->arr[s->top];
}

void display(Stack *s)
{
    if (isEmpty(s))
    {
        printf("Stack is empty\n");
        return;
    }
    printf("Stack elements:\n");
    for (int i = s->top; i >= 0; i--)
    {
        printf("%d\n", s->arr[i]);
    }
}
```

```
int main()
{
    Stack s;
    initStack(&s);
    push(&s, 10);
    push(&s, 20);
    push(&s, 30);
    printf("Top element is %d\n", peek(&s));
    display(&s);
    printf("%d popped from stack\n", pop(&s));
    display(&s);
    pop(&s);
    pop(&s);
    pop(&s);

    return 0;
}
```

OUTPUT

10 pushed to stack

20 pushed to stack

30 pushed to stack

Top element is 30

Stack elements:

30

20

10

30 popped from stack

Stack elements:

20

10

Stack underflow