

## \* LAB programme : 2 :

- Q. write a program to simulate the working of stack using an array using push, pop, and display.  
the programme should print message for stack overflow and stack underflow.

Soln:

```
#include <stdio.h>
#include <stdlib.h>
#define stack-size 5
```

```
int top = -1;
int s[10];
int item;
```

```
void push() {
```

```
    if (top == stack-size - 1) {
```

```
        printf("stack overflow\n");
        return;
```

```
    }
```

```
    else {
```

```
        printf("Enter the item to be inserted\n");
        scanf("%d", &item);
```

★

```
        top = top + 1;
        s[top] = item;
```

or

```
s[++top] = item;
// pre-increment
```

```
    }
```

```
}
```

convert

void

Date \_\_\_\_\_  
Page \_\_\_\_\_

int pop() {

if (top == -1) {

printf("Stack Empty");  
~~return 0;~~

}

else {

printf("Element removed is : %d \n",  
s[top--]);

~~return -1;~~

★ post  
decrement

}

}

void display() {

int i;

if (top == -1) {

printf("Stack is Empty");

★ ~~return;~~ [unnecessary]

}

printf("The stack items are : \n");

for (i = top; i >= 0; i--) {

printf("%d \n", s[i]);

}

}

a[2] ↑  
a[1]  
a[0]



```
void main () {
```

```
    int item = deleted;
```

```
    int choice;
```

```
    for (;;) {
```

```
        printf ("Enter \n 1. PUSH \n 2. POP \n 3. DISPLAY \n  
        4. EXIT \n");
```

```
        scanf ("%d", &choice);
```

```
        switch (choice) {
```

```
            case 1 : push ();  
                    break;
```

```
            case 2 : pop ();  
                    break;
```

```
            case 3 : display ();  
                    break;
```

```
            default ; exit (0); // invalid input //
```

```
        }
```

```
    }
```

```
}
```



## \* Output:

→ Entel

1. PUSH
2. POP
3. DISPLAY
4. EXIT

Enter your choice.

1

Enter Element to be inserted

25

→ Entel

1. PUSH
2. POP
3. DISPLAY
4. EXIT

Enter your choice

1

Enter element to be inserted

30

→ Entel

1. PUSH
2. POP
3. DISPLAY
4. EXIT

Enter your choice

3

The components of stack are: 7

25

30