NAME:- SAGAR ADHIKARI
ROLL NO:- PUR078BEI034
FACULTY:- ELECTRONICS

DSA LAB SHEET NO. 1

TITLE:- STATIC IMPLEMENTATION OF STACK USING ARRAY

THEORY:-

Stack:- A stack is an ordered collection of items where the addition of new items and the removal of existing items always takes place at the same end i.e. top of stack.

Array:- An array is a collection of items of same data type stored at contiguous memory locations.

PROGRAM CODE:-

```
#include <stdio.h>
#include <string.h>
#define true 1
#define false 0
#define max 5
struct stack
{
	int TOS;
	char data[max][20];
};
int IsFull(struct stack *s)
{
	return (s->TOS == max - 1) ? true : false;
}
int IsEmpty(struct stack *s)
{
	return s->TOS == -1 ? true : false;
}
void push(struct stack *s, char *enteredname)//since name is a string.
{
	strcpy(s->data[++s->TOS], enteredname);
}
char *Pop(struct stack *s)
{
	return s->data[s->TOS--];
}
```

DSA LAB SHEET NO. 1

```
int main()
{
        int choice;
        char name[20];
        struct stack S = {-1};// TOS= -1
        do
        {
                printf("\n1. Push\n2. Pop\n3. Exit\n");
                scanf("%d", &choice);
                switch (choice)
              {
                  case 1:
                        if (IsFull(&S)) printf("\nStack OverFlow\n");
                        else
                        {
                              printf("\nStack is not full so,\n");
                              printf("Enter Name to insert in stack:- ");
                              scanf("%s", name);
                              push(&S, name);
                              printf("\nThe name that was pushed onto the stack
is:- %s\n",name);
                        }
                  break;
                  case 2:
                          if (IsEmpty(&S)) printf("\nStack UnderFlow\n");
                          else
                        {
                              printf("\nThe name that was popped from the stack is:- %s
\n",Pop(&S));
                        }
                  break;
                  case 3:
                        printf("You have decided to exit.\n");
                  break;
                  default:
                        printf("Enter 1,2,3 only\n");
                  break;
              }
```

```
    } while (choice != 3);
}
```

OUTPUT:-

```
1. Push
2. Pop
3. Exit
1

Stack is not full so,
Enter Name to insert in stack:- John

The name that was pushed onto the stack is:- John

1. Push
2. Pop
3. Exit
1

Stack is not full so,
Enter Name to insert in stack:- Rinky

The name that was pushed onto the stack is:- Rinky

1. Push
2. Pop
3. Exit
1

Stack is not full so,
Enter Name to insert in stack:- Patrick

The name that was pushed onto the stack is:- Patrick

1. Push
2. Pop
3. Exit
```

```
1

Stack is not full so,
Enter Name to insert in stack:- Otis

The name that was pushed onto the stack is:- Otis

1. Push
2. Pop
3. Exit
1

Stack is not full so,
Enter Name to insert in stack:- Morbish

The name that was pushed onto the stack is:- Morbish

1. Push
2. Pop
3. Exit
1

Stack OverFlow

1. Push
2. Pop
3. Exit
2

The name that was popped from the stack is:- Morbish

1. Push
2. Pop
3. Exit
2

The name that was popped from the stack is:- Otis
```

DSA LAB SHEET NO. 1

```
1. Push
2. Pop
3. Exit
2

The name that was popped from the stack is:- Patrick

1. Push
2. Pop
3. Exit
2

The name that was popped from the stack is:- Rinky

1. Push
2. Pop
3. Exit
2

The name that was popped from the stack is:- John

1. Push
2. Pop
3. Exit
2

Stack UnderFlow

1. Push
2. Pop
3. Exit
3
You have decided to exit.
```