EXPERIMENT 4

Implementing Stack ADT using array

C file:

```
#include<stdio.h>
#include<conio.h>
#include"stack.h"

void main()
{
   int   ch,num,x;              printf("STACK   Implementation
By:\nAayush Joshi\nSE-4\nRoll No-14\n");     while(1)
   {
      printf("--------STACK
Implementation---------");
printf("\n1.Push\t2.Pop\t3.Peek\t4.Exit\
n");         printf("\nEnter your choice:");
scanf("%d",&ch);         switch(ch)
      {
         case 1: printf("Enter number to
be pushed:");
scanf("%d",&num);
push(num);
```

```
                printf("%d is
inserted\n",num);                break;


          case 2: x=pop();
if(x!=-1)
                {
                printf("%d element is
deleted\n",x);
                }


break;
case 3:x=peek();
if(x!=-1)
                {
                printf("%d is the Peek
Element\n",x);
                }
break;
case 4:exit(0);
       }
   }
}
```

Header file:

```
#include<stdio.h> #define
max_size 20 int
stack[max_size],empty_si
ze=-1; void push(int num)
{
if(empty_size==max_s
ize-1)
   {
      printf("Stack Overflow\n");
   }
else
   {
      empty_size=empty_size+1;
stack[empty_size]=num;
   }
} int
pop()
{
int x;
if(em
pty_si
```

```
ze==-
1)
   {
      printf("\n\tStack
Underflow\n");        return
-1;    }    else
   {
      x=stack[empty_size];
empty_size=empty_size-1;
return x;
   }
} int peek()
{
if(isempty(
))
   {
      printf("\n\tStack
underflow\n");        return
-1;    }    else
   {
      return stack[empty_size];
   }
```

```
} int isfull() {
if(empty_size==max_s
ize-1)
    {
return 1;
    }
    else
{
return 0;
    }
} int
isempty()
{
    if(empty_size==-1)
    {
return 1;
    }
else     {
return 0;
    }
}
```

Output:

```
Aayush Joshi
SE-4
Roll No-14
-------------STACK Implementation------------------
1.Push  2.Pop   3.Peek  4.Exit

Enter your choice:1
Enter number to be pushed:23
23 is inserted
-------------STACK Implementation------------------
1.Push  2.Pop   3.Peek  4.Exit

Enter your choice:1
Enter number to be pushed:25
25 is inserted
-------------STACK Implementation------------------
1.Push  2.Pop   3.Peek  4.Exit

Enter your choice:1
Enter number to be pushed:26
26 is inserted
-------------STACK Implementation------------------
1.Push  2.Pop   3.Peek  4.Exit

Enter your choice:2
26 element is deleted
-------------STACK Implementation------------------
1.Push  2.Pop   3.Peek  4.Exit

Enter your choice:3
25 is the Peek Element
```

```
Enter your choice:1
Enter number to be pushed:26
26 is inserted
-------------STACK Implementation------------------
1.Push  2.Pop   3.Peek  4.Exit

Enter your choice:2
26 element is deleted
-------------STACK Implementation------------------
1.Push  2.Pop   3.Peek  4.Exit

Enter your choice:3
25 is the Peek Element
-------------STACK Implementation------------------
1.Push  2.Pop   3.Peek  4.Exit

Enter your choice:4


...Program finished with exit code 0
Press ENTER to exit console.
```