

Name – Abhinab Roy

Roll no. – 26

Stream – C.S.E.

Section – A

University Roll no. – 10900119040

Q) Write a menu driven program to implement PUSH, POP, PEEP operations in Stack using array.

Code –

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  #define MAX 10
4
5  int stack_arr[MAX];
6  int top = -1;
7
8  void push(int item);
9  int pop();
10 int peep();
11 int isEmpty();
12 int isFull();
13 void display();
14
15 int main()
16 {
17     int choice,item;
18     while(1)
19     {
20         printf("\n1.Push\n");
21         printf("\n2.Pop\n");
22         printf("\n3.PEEP\n");
23         printf("\n4.Display all stack elements\n");
24         printf("\n5.Quit\n");
25         printf("\nEnter your choice : ");
26         scanf("%d",&choice);
27         switch(choice)
28         {
29             case 1 :
30                 printf("\nEnter the item to be pushed : ");
31                 scanf("%d",&item);
32                 push(item);
33                 break;
34             case 2 :
35                 item = pop();
36                 printf("\nPopped item is : %d\n",item );
37                 break;
```

```
37                 break;
38             case 3 :
39                 peep();
40                 break;
41             case 4 :
42                 display();
43                 break;
44             case 5 :
45                 exit(1);
46             default:
47                 printf("\nWrong choice\n");
48         }
49     }
50     return 0;
51 }
52
53 void push(int item)
54 {
55     if( isFull() )
56     {
57         printf("\nStack Overflow\n");
58         return;
59     }
60     top = top+1;
61     stack_arr[top] = item;
62 }
63
64 int pop()
65 {
66     int item;
67     if( isEmpty() )
68     {
69         printf("\nStack Underflow\n");
70         exit(1);
71     }
72     item = stack_arr[top];
73     top = top-1;
```

```

73 |     }
74 |     item = stack_arr[top];
75 |     top = top-1;
76 |     return item;
77 | }
78 |
79 | int peep()
80 | {
81 |     int s;
82 |     printf("\nEnter Position for search:->");
83 |     scanf("%d",&s);
84 |     if( isEmpty() )
85 |     {
86 |         printf("\nStack Underflow\n");
87 |         exit(1);
88 |     }
89 |     if(s == MAX -1)
90 |     {
91 |         printf("\nThe Stack is Overflow...");
92 |         exit(1);
93 |     }
94 |     else
95 |     {
96 |         printf("\nThe Element is:->%d",stack_arr[s-1]);
97 |     }
98 | }
99 |
100 | int isEmpty()
101 | {
102 |     if( top == -1 )
103 |         return 1;
104 |     else
105 |         return 0;
106 | }
107 |
108 | int isFull()
109 | {

```

```

93 |     }
94 |     else
95 |     { printf("\nThe Element is:->%d",stack_arr[s-1]);
96 |     }
97 | }
98 | }
99 |
100 | int isEmpty()
101 | {
102 |     if( top == -1 )
103 |         return 1;
104 |     else
105 |         return 0;
106 | }
107 |
108 | int isFull()
109 | {
110 |     if( top == MAX-1 )
111 |         return 1;
112 |     else
113 |         return 0;
114 | }
115 |
116 | void display()
117 | {
118 |     int i;
119 |     if( isEmpty() )
120 |     {
121 |         printf("\nStack is empty\n");
122 |         return;
123 |     }
124 |     printf("\nStack elements :\n\n");
125 |     for(i=top;i>=0;i--)
126 |         printf(" %d\n", stack_arr[i] );
127 |     printf("\n");
128 | }

```

P.T.O.

OUTPUT –

```
1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit

Enter your choice : 1

Enter the item to be pushed : 26

1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit

Enter your choice : 1

Enter the item to be pushed : 46

1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit

Enter your choice : 1

Enter the item to be pushed : 90

1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit

Enter your choice : 1

Enter the item to be pushed : 34
```

```
Enter your choice : 1

Enter the item to be pushed : 34

1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit

Enter your choice : 1

Enter the item to be pushed : 67

1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit

Enter your choice : 4

Stack elements :

67
34
90
46
26

1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit

Enter your choice : 2

Popped item is : 67
```

```
Enter your choice : 1
Enter the item to be pushed : 67
1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit
```

Enter your choice : 4

Stack elements :

```
67
34
90
46
26
```

```
1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit
```

Enter your choice : 2

Popped item is : 67

```
1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit
```

Enter your choice : 4

Stack elements :

Enter your choice : 4

Stack elements :

```
34
90
46
26
```

```
1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit
```

Enter your choice : 3

Enter Position :->3

The Element is:->90

```
1.Push
2.Pop
3.Peep
4.Display all stack elements
5.Quit
```

Enter your choice : 4

Stack elements :

```
34
90
46
26
```

```
1.Push
2.Pop
3.Peep
4.Display all stack elements
```

```
Stack elements :  
  
34  
90  
46  
26  
  
1.Push  
2.Pop  
3.Peep  
4.Display all stack elements  
5.Quit  
  
Enter your choice : 5  
  
-----  
Process exited after 69.31 seconds with return value 1  
Press any key to continue . . .
```

*****END*****