

```
1 #include <stdio.h>
2 #define SIZE 100
3 int stack1[SIZE], top1 = -1;
4 int stack2[SIZE], top2 = -1;
5 void push(int x) {
6     stack1[++top1] = x;
7 }
8 int pop() {
9     if (top2 == -1) {
10         while (top1 >= 0)
11             stack2[++top2] = stack1[top1--];
12     }
13     if (top2 >= 0)
14         return stack2[top2--];
15     return -1;
16 }
17 int peek() {
18     if (top2 == -1) {
19         while (top1 >= 0)
20             stack2[++top2] = stack1[top1--];
21     }
22     if (top2 >= 0)
23         return stack2[top2];
24     return -1;
25 }
```

```
26 int empty() {
27     return (top1 == -1 && top2 == -1) ? 1 : 0;
28 }
29 void printQueue() {
30     int i;
31     printf("Queue is: [");
32     for (i = top2; i >= 0; i--)
33         printf("%d%s", stack2[i], (i != 0 || top1 != -1) ? ", " : "");
34     for (i = 0; i <= top1; i++)
35         printf("%d%s", stack1[i], (i != top1) ? ", " : "");
36     printf("]\n");
37 }
38 int main() {
39     push(1);
40     printQueue();
41     push(2);
42     printQueue();
43     printf("Peek: %d\n", peek());
44     printf("Pop: %d\n", pop());
45     printQueue();
46     printf("Empty: %s\n", empty() ? "true" : "false");
47     return 0;
48 }
```

Queue is: [1]

Queue is: [1, 2]

Peek: 1

Pop: 1

Queue is: [2]

Empty: false