Green University of Bangladesh
Department of Computer Science and Engineering(CSE)

Faculty of Sciences and Engineering

Semester: (Summer, Year:2022), B.Sc. in CSE (Day)

LAB REPORT NO: 03

Course Title: Data Structure Lab

Course Code: CSE 106 Section: DB

Lab Experiment Name: Stack and Queue

Student Details

	Name	ID
1.	Shariful Islam Emon	213902056

Lab Date : 25/07/2022

Submission Date : 08/08/2022

Course Teacher's Name : Farhana Akter Sunny

[For Teachers use only: Don't Write Anything inside this box]

Lab Report Status

Marks: …… Signature:.....

Comments:..... Date:.....

1. TITLE OF THE LAB EXPERIMENT

"Stack and Queue"

2. IMPLEMENTATION

Answer to the problem no: 1

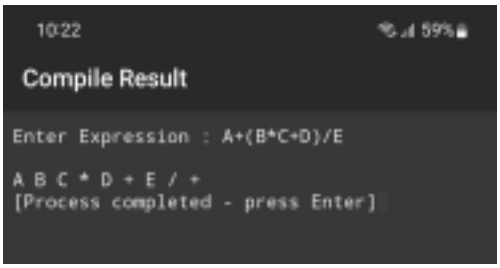
Problem Statement: Implement a program for converting an infix expression to postfix expression using stack.

Code:

```
#include<stdio.h>
#include<ctype.h>

//Shariful Islam Emon 213902056

char stack[100];
int top = -1;

void push(char x)
{
   stack[++top] = x;
}

char pop()
{
   if(top == -1)
      return -1;
   else
      return stack[top--];
}

int priority(char x)
{
   if(x == '(')
      return 0;
   if(x == '+' || x == '-')
      return 1;
   if(x == '*' || x == '/')
```

```
        return 2;
    return 0;
}

int main()
{
    char exp[500];
    char *e, x;
    printf("Enter Expression : ");
    scanf("%s",exp);
    printf("\n");
    e = exp;

    while(*e != '\0')
    {
        if(isalnum(*e))
            printf("%c ",*e);
        else if(*e == '(')
            push(*e);
        else if(*e == ')')
        {
            while((x = pop()) != '(')
                printf("%c ", x);
        }
        else
        {
            while(priority(stack[top]) >= priority(*e))
                printf("%c ",pop());
            push(*e);
        }
        e++;
    }

    while(top != -1)
    {
        printf("%c ",pop());
    }return 0;
}
```

Output:



Answer to the problem no: 2

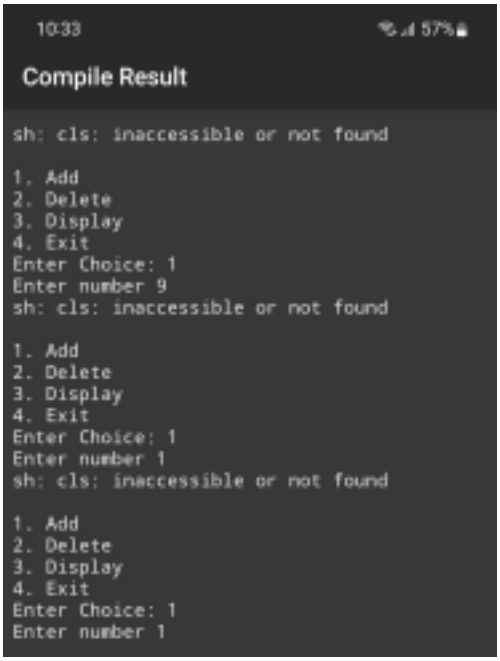
Problem Statement: Implement a program of circular queue.

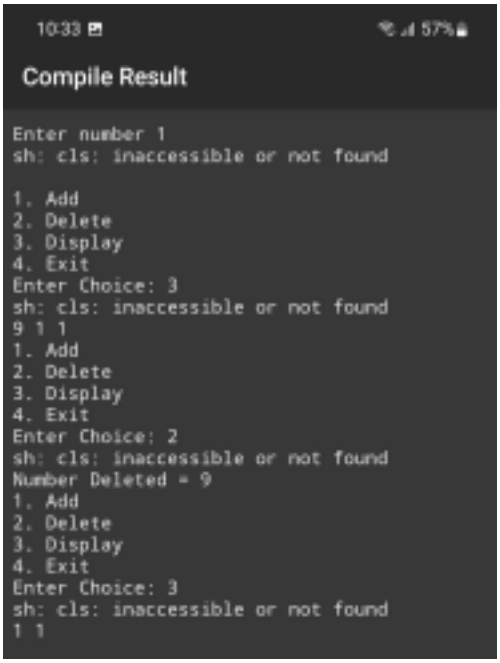
Code:

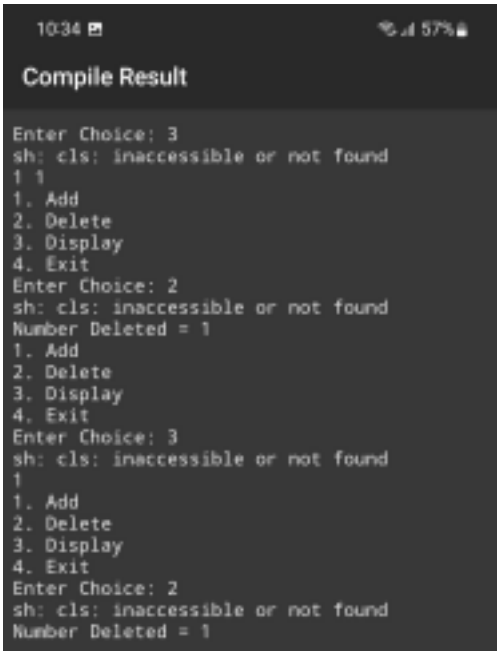
```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#define size 5
//Shariful Islam Emon 213902056
int main()
 {
   int arr[size],R=-1,F=0,te=0,ch,n,i,x;

   for(;;)
   {
      system("cls");
      printf("\n1. Add\n");
      printf("2. Delete\n");
      printf("3. Display\n");
      printf("4. Exit\n");
      printf("Enter Choice: ");
      scanf("%d",&ch);

      switch(ch)
      {
         case 1:
            if(te==size)

            {
               printf("Queue full");
               getch();
            }
            else
            {
               printf("Enter number ");
               scanf("%d",&n);
               R=(R+1)%size;
               arr[R]=n;
               te=te+1;
            }
            break;

         case 2:
            if(te==0)
            {
```

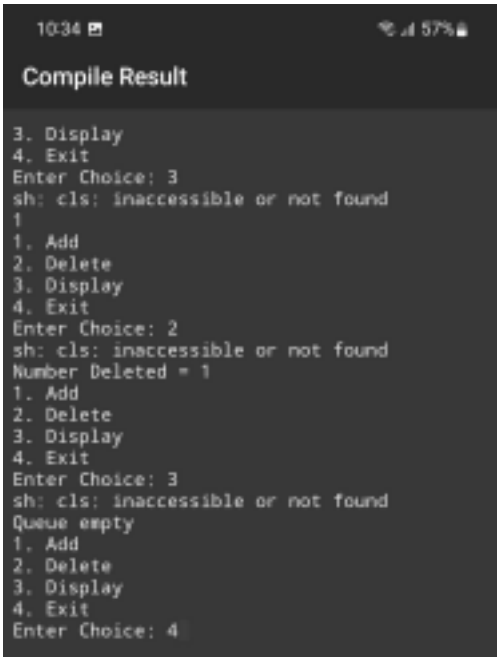
```
            printf("Queue empty");
            getch();
          }
          else
          {
            printf("Number Deleted = %d",arr[F]);
            F=(F+1)%size;
            te=te-1;
            getch();
          }
          break;

        case 3:
          if(te==0)
          {
            printf("Queue empty");
            getch();
          }
          else
          {
            x=F;
            for(i=1; i<=te; i++)
            {
              printf("%d ",arr[x]);
              x=(x+1)%size;
            }
            getch();
          }
          break;
        case 4:
          exit(0);
          break;

        default:
          printf("Wrong Choice");
          getch();
      }
    }
  return 0;
}
```

Output:


10:33
57%
Compile Result
sh: cls: inaccessible or not found
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 1
Enter number 9
sh: cls: inaccessible or not found
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 1
Enter number 1
sh: cls: inaccessible or not found
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 1
Enter number 1


10:33
57%
Compile Result
Enter number 1
sh: cls: inaccessible or not found
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 3
sh: cls: inaccessible or not found
9 1 1
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 2
sh: cls: inaccessible or not found
Number Deleted = 9
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 3
sh: cls: inaccessible or not found
1 1


10:34
57%
Compile Result
Enter Choice: 3
sh: cls: inaccessible or not found
1 1
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 2
sh: cls: inaccessible or not found
Number Deleted = 1
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 3
sh: cls: inaccessible or not found
1
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 2
sh: cls: inaccessible or not found
Number Deleted = 1


10:34
57%
Compile Result
3. Display
4. Exit
Enter Choice: 3
sh: cls: inaccessible or not found
1
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 2
sh: cls: inaccessible or not found
Number Deleted = 1
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 3
sh: cls: inaccessible or not found
Queue empty
1. Add
2. Delete
3. Display
4. Exit
Enter Choice: 4