Prog-1

```
#include<stdio.h>
#include<stdlib.h>
int a[10],n;
void create();
void insert();
void del();
void display();
int main()
{
   int ch;
   printf("array creation");
   create();
   do
   {
      printf("\n1.insert\n2.del\n3.display\n4.exit");
      printf("Enter your choice:");
      scanf("%d",&ch);
      switch(ch)
      {
         case 1:insert();
         break;
         case 2:del();
         break;
         case 3:display();
         break;
         case 4:exit(0);
         default:printf("invaild choice");
      }
```

```
    }while(ch!=4);
    return 0;
}

void create()
{
    int i;
    printf("Enter total no of elements:");
    scanf("%d",&n);
    printf("Enter array elements:");
    for(i=0;i<n;i++)
    scanf("%d",&a[i]);
}
void insert()
{
    int e,i,pos;
    printf("Enter the position to be inserted:");
    scanf("%d",&pos);
    printf("Enter the element to be inserted:");
    scanf("%d",&e);
    for(i=n;i>pos;i--)
    a[i]=a[i-1];
    a[pos]=e;
    n=n+1;
}
void del()
{
    int e,i,pos;
    printf("Enter the position to be deleted:");
    scanf("%d",&pos);
    e=a[pos];
    for(i=pos;i<n-1;i++)
    a[i]=a[i+1];
    n=n-1;
    printf("deleted element is %d",e);
}
void display()
{
    int i;
    printf("Array elements are:");
    for(i=0;i<n;i++)
    printf("%d",a[i]);
}
```

Prog-2

```
#include<stdio.h>
#include<stdlib.h>
# define MAX 5
int s[MAX],top=-1;
void push();
int pop();
void display();
int isoverflow();
int isunderflow();
int main()
{
  int ch,e;
  do
  {
    printf("\n1.push\n2.pop\n3.display\n4.exit\n");
    printf("Enter your choice:");
    scanf("%d",&ch);
    switch(ch)
    {
      case 1:if(isoverflow())
      printf("stack overflow");
      else
      push();
      break;
      case 2:if(isunderflow())
      printf("stack underflow");
      else
      {
```

```
      e=pop();
      printf("Deleted element:%d",e);
      }
      break;
      case 3:if(isunderflow())
      printf("stack underflow");
      else
      display();
      break;
      case 4:exit(0);
      default:printf("invaild choice");
    }
  }while(ch!=4);
  return 0;
}

int isoverflow()
{
  if(top==MAX-1)
  return 1;
  else
  return 0;
}
int isunderflow()
{
  if(top==-1)
  return 1;
  else
  return 0;
}
void push()
```

```
{
   int e;
   printf("Enter the element to be insserted:");
   scanf("%d",&e);
   s[++top]=e;
}
int pop()
{
   return(s[top--]);
}
void display()
{
   int i;
   printf("Array elements are:");
   for(i=top;i>=0;i--)
   printf("%d",s[i]);
}
```

Prog-3

```
#include<stdio.h>

#include<ctype.h>

char s[20];

int top=-1;

void push(char);

char pop();

int prior(char);

int main()

{

   char infix[20],postfix[20];

   int i,j=0;

   printf("Enter the infix expresssion:");

   scanf("%s",infix);

   push('#');

   for(i=0;infix[i]!='\0';i++)

   {

      if(isalnum(infix[i]))

      postfix[j++]=infix[i];

      else if(infix[i]=='(')

      push(infix[i]);

      else if(infix[i]==')')

      {

         while(s[top]!='(')

         postfix[j++]=pop();

         pop();

      }

      else

      {

         while(prior(s[top])>=prior(infix[i]))
```

```
            postfix[j++]=pop();
            push(infix[i]);
        }
    }
    while(s[top]!='#')
    {
        postfix[j++]=pop();
    }
    postfix[j]='\0';
    printf("postfix expresssionis:%s",postfix);
    return 0;
}
void push(char x)
{
    s[++top]=x;
}
char pop()
{
    return(s[top--]);
}
int prior(char x)
{
    if(x=='^')
    return 3;
    if(x=='*'||x=='/'||x=='%')
    return 2;
    if(x=='+'||x=='-')
    return 1;
    if(x=='('||x=='#')
    return 0;
}
```

Prog-4

```
#include<stdio.h>

#include<ctype.h>

#include<math.h>

int s[20];

int top=-1;

void push(int);

int pop();

int main()

{

    char postfix[20];

    int i,res=0,op1,op2;

    printf("Enter the postfix expresssion:");

    scanf("%s",postfix);

    for(i=0;postfix[i]!='\0';i++)

    {

        if(isdigit(postfix[i]))

        push(postfix[i]-'0');

        else

        {

            op2=pop();

            op1=pop();

            switch(postfix[i])

            {

                case '+':res=(op1+op2);

                push(res);

                break;

                case '-':res=op1-op2;

                push(res);

                break;
                case '*':res=op1*op2;
                push(res);
                break;
                case '/':res=op1/op2;
                push(res);
                break;
                case '%':res=op1%op2;
                push(res);
                break;
                case '^':res=(op1^op2);
                push(res);
                break;
                //default:printf("invalid choice");
            }
        }
    }
    printf("postfix expresssion is:%d",res);
    return 0;
}
void push(int ch)
{
    s[++top]=ch;
}
int pop()
{
    return(s[top-1]);
}
```

Prog-5a

```
#include<stdio.h>
int fibo(int n);
int main()
{
   int n,i;
   printf("Enter the fibo no:");
   scanf("%d",&n);
   printf("fibo series:");
   for(i=0;i<=n;i++)
   {
      printf("%d",fibo(i));
   }
   return 0;
}
int fibo(int n)
{
   if(n==0||n==1)
   return n;
   else
   return fibo(n-1)+fibo(n-2);
}
```

Prog-5b

```
#include<stdio.h>

void tower(int,char,char,char);

//int n;

int main()

{

   int n;

   printf("Enter no of disks:");

   scanf("%d",&n);

   if(n==0)

   {

      printf("no disks found");

      return 0;

   }

   printf("moves involed in tower of hanio");

   tower(n,'A','c','B');

   return 0;

}

void tower(int n,char source,char dest,char temp)

{

   if(n==1)

   {

      printf("\nmoves %d disks from %c to %c",n,source,dest);

      return;

   }

   tower(n-1,source,temp,dest);

   printf("\nmoves %d disks from %c to %c",n,source,dest);

   tower(n-1,temp,dest,source);

}
```

Prog-6

```
#include<stdio.h>

#include<stdlib.h>

# define MAX 5

int q[MAX],f=-1,r=-1;

void insert();

void del();

void display();

int main()

{

   int ch;

   do

   {

      printf("\n1.insert\n2.del\n3.display\n4.exit\n");

      printf("Enter your choice:");

      scanf("%d",&ch);

      switch(ch)

      {

         case 1:insert();

         break;

         case 2:del();

         break;

         case 3:display();

         break;

         case 4:exit(0);

         default:printf("invaild choice");

      }

   }while(ch!=4);

   return 0;

}
```

```
void insert()
{
   int e;
   if(r==MAX-1)
   {
      printf("q is full");
      return;
   }
   printf("Enter the element to be insserted:");
   scanf("%d",&e);
   r=r+1;
   q[r]=e;
   if(f==-1)
   f=0;
}
void del()
{
   int e;
   if(f==-1)
   {
      printf("q is empty");
      return;
   }
   e=q[f];
   if(f==r)
   f=r=-1;
   else
   {
      f=f+1;
      printf("deleteed elememt is %d",e);
```

```
    }
}
void display()
{
    int i;
    if(f==-1)
    {
        printf("q is empty");
        return;
    }
    printf("content of q");
    for(i=f;i<=r;i++)
    printf("%d",q[i]);
}
```

Prog-7

```
#include<stdio.h>
#include<stdlib.h>
#define MAX 5
char cq[20];
int f=-1,r=-1;
void insert();
void del();
void display();
int main()
{
  int ch;
  do
  {
    printf("\n1.insert\n2.del\n3.display\n4.exit\n");
    printf("Enter your choice:");
    scanf("%d",&ch);
    switch(ch)
    {
      case 1:insert();
      break;
      case 2:del();
      break;
      case 3:display();
      break;
      case 4:exit(0);
      default:printf("invaild choice");
    }
  }while(ch!=4);
  return 0;
```

```
}
void insert()
{
   char e;
   if((r+1)%MAX==f)
   {
      printf("cq is full");
      return;
   }
   printf("Enter the element to be insserted:");
   scanf(" %c",&e);
   r=(r+1)%MAX;
   cq[r]=e;
   if(f==-1)
   f=0;
}
void del()
{
   char e;
   if(f==-1)
   {
      printf("q is empty");
      return;
   }
   else
{
   e=cq[f];
   if(f==r)
   f=r=-1;
   else
      f=(f+1)%MAX;
```

```
        printf("deleteed elememt is %d",e);
    }
}
void display()
{
    int i;
    if(f==-1)
    {
        printf("cq is empty");
        return;
    }
    else
    {
    i=f;
    while(i!=r)
    {
        printf("%c",cq[i]);
        i=(i+1)%MAX;
    }
    printf("%c\n",cq[r]);
    }
}
```