#include<stdio.h>

# define MAX 5

int cqueue\_arr[MAX];

int front = -1;

int rear = -1;

void insert(int item)

{

if((front == 0 && rear == MAX-1) || (front == rear+1))

{

printf("Queue Overflow n");

return;

}

if(front == -1)

{

front = 0;

rear = 0;

}

else

{

if(rear == MAX-1)

rear = 0;

else

rear = rear+1;

}

cqueue\_arr[rear] = item ;

}

void deletion()

{

if(front == -1)

{

printf("Queue Underflown");

return ;

}

printf("Element deleted from queue is : %dn",cqueue\_arr[front]);

if(front == rear)

{

front = -1;

rear=-1;

}

else

{

if(front == MAX-1)

front = 0;

else

front = front+1;

}

}

void display()

{

int front\_pos = front,rear\_pos = rear;

if(front == -1)

{

printf("Queue is emptyn");

return;

}

printf("Queue elements :n");

if( front\_pos <= rear\_pos )

while(front\_pos <= rear\_pos)

{

printf("%d ",cqueue\_arr[front\_pos]);

front\_pos++;

}

else

{

while(front\_pos <= MAX-1)

{

printf("%d ",cqueue\_arr[front\_pos])

front\_pos++;

}

front\_pos = 0;

while(front\_pos <= rear\_pos)

{

printf("%d ",cqueue\_arr[front\_pos]);

front\_pos++;

}

}

printf("n");

}

int main()

{

int choice,item;

do

{

printf("1.Insertn");

printf("2.Deleten");

printf("3.Displayn");

printf("4.Quitn");

printf("Enter your choice : ");

scanf("%d",&choice);

switch(choice)

{

case 1 :

printf("Input the element for insertion in queue : ");

scanf("%d", &item);

insert(item);

break;

case 2 :

deletion();

break;

case 3:

display();

break;

case 4:

break;

default:

printf("Wrong choicen");

}

}while(choice!=4);

return 0;

}



#include<stdio.h>

main()

{

    int arr[20],n,i,a,count=0,b=1;

    scanf("%d",&n);

    for(i=0;i<n;i++)

    scanf("%d",&arr[i]);

    for(i=0;i<n;i++)

    {

        a=arr[i];

        while(a==0)

        {

            a=a/10;

            count++;

        }

        b\*=a;

    }

    printf("%d",b);

    for(i=0;i<count;i++)

        printf("0");

}



#include <iostream>

using namespace std;

int stack[100], n=100, top=-1,minimum;

void push(int val) {

    if(top==-1)

    {

        top++;

        stack[top]=val;

        minimum=stack[top];

    }

   else if(top>=n-1)

   {

        cout<<"Stack Overflow"<<endl;

   }

   else {

        top++;

        stack[top]=val;

        if (stack[top]<minimum)

        {

            minimum=stack[top];

        }

   }

}

void pop() {

   if(top<=-1)

   cout<<"Stack Underflow"<<endl;

   else {

      cout<<"The popped element is "<< stack[top] <<endl;

      top--;

   }

}

void display() {

   if(top>=0) {

      cout<<"Stack elements are:";

      for(int i=top; i>=0; i--)

      cout<<stack[i]<<" ";

      cout<<endl;

   } else

   cout<<"Stack is empty";

}

int main() {

   int ch, val;

    cout<<"1) Push in stack"<<endl;

    cout<<"2) Pop from stack"<<endl;

    cout<<"3) Display stack"<<endl;

    cout<<"4) Display Minimum"<<endl;

    cout<<"5) Exit"<<endl;

   do {

      cout<<"\nEnter choice: "<<endl;

      cin>>ch;

      switch(ch) {

         case 1: {

            cout<<"\nEnter value to be pushed:"<<endl;

            cin>>val;

            push(val);

            break;

         }

         case 2: {

            pop();

            break;

         }

         case 3: {

            display();

            break;

         }

         case 4:

            cout<<"\nMinimum Is: "<<minimum;

            break;

         case 5: {

            cout<<"\nExit"<<endl;

            break;

         }

         default: {

            cout<<"\nInvalid Choice"<<endl;

         }

      }

   }while(ch!=5);

   return 0;

}