doubleendedqueue\_07.cpp

**Compile:** g++ doubleendedqueue\_07.cpp -o doubleendedqueue\_07

**Run:** ./doubleendedqueue\_07

**Program:**

#include<iostream>

using namespace std;

class node

{

public:

intdata,n;

class node \*next;

class node \*prev;

};

classdqueue: public node

{

node \*head,\*tail;

int top1,top2;

public:

dqueue()

{

cout<<"Enter the array size:-";

cin>>n;

top1=0;

top2=0;

head=NULL;

tail=NULL;

}

void push(int x){

node \*temp;

intch;

if(top1+top2 >=n)

{

cout<<"dqueue overflow";

return ;

}

if( top1+top2 == 0)

{

head = new node;

head->data=x;

head->next=NULL;

head->prev=NULL;

tail=head;

top1++;

}

else

{

cout<<" Add element from 1.beginning 2.ending\n enter ur choice:";

cin>>ch;

if(ch==1)

{

top1++;

temp=new node;

temp->data=x;

temp->next=head;

temp->prev=NULL;

head->prev=temp;

head=temp;

}

else

{

top2++;

temp=new node;

temp->data=x;

temp->next=NULL;

temp->prev=tail;

tail->next=temp;

tail=temp;

}

}

}

void pop()

{

intch;

if(top1 + top2 ==1)

{

top1--;

top2--;

}

else

{

cout<<"Delete element form 1.beginning 2.ending\n Enter ur choice:";

cin>>ch;

if(top1 + top2 <=0)

{

cout<<"\nDqueue under flow";

return;

}

if(ch==1)

{

head=head->next;

head->prev=NULL;

top1--;

}

else

{

top2--;

tail=tail->prev;

tail->next=NULL;

}

}

}

void display()

{

intch;

node \*temp;

cout<<"display from 1.Staring 2.Ending\n Enter ur choice";

cin>>ch;

if(top1+top2 <=0)

{

cout<<"under flow";

return ;

}

if (ch==1)

{

temp=head;

while(temp!=NULL)

{

cout<< temp->data <<" ";

temp=temp->next;

}

}

else

{

temp=tail;

while( temp!=NULL)

{

cout<<temp->data << " ";

temp=temp->prev;

}

}

}

};

int main()

{

dqueue d1;

intch;

do

{

cout<<"1.INSERT 2.DELETE 3.DISPLAY 4.EXIT\n Enter ur choice:";

cin>>ch;

switch(ch)

{

case 1: cout<<"enter element";

cin>>ch;

d1.push(ch); break;

case 2: d1.pop(); break;

case 3: d1.display(); break;

default: cout<<"Enter proper choice.";

}

}while(ch!=4);

return 0;

}

**Output:**

