#include<iostream>

#include<stdbool.h>

using namespace std;

class Array

{

private:

int capacity;

int lastindex;

int \*ptr;

public:

Array(int);

bool isEmpty();

void append(int);

void edit(int,int);

void insert(int,int);

void del(int);

bool isFull();

int get(int);

int count ();

~Array();

int find(int);

};

int Array::find(int data)

{

int i;

for(i=0;i<=lastindex;i++)

if(ptr[i] =data)

return i;

return -1;

}

Array::~Array()

{

delete[]ptr;

}

int Array::count()

{

return(lastindex+1);

}

int Array::get(int index)

{

if(index>=0 &&index<=lastindex)

return ptr[index];

cout<<endl<<"invalid index or empty array";

return -1;

}

bool Array:: isFull()

{

return(lastindex=capacity-1);

}

void Array::del(int index)

{

int i;

if(lastindex==-1)

cout<<endl<<"array is empty";

else if(index<0||index>lastindex)

cout<<endl<<"invelid index";

else

for(i=index;i<lastindex;i++)

ptr[i]=ptr[i+1];

lastindex--;

}

void Array::edit(int index,int data)

{

if(index>=0&&index<=lastindex)

ptr[index]=data;

}

Array::Array (int cap)

{

capacity=cap;

lastindex=-1;

ptr=new int[capacity];

void insert (int,int);

}

void Array::insert (int index,int data)

{

int i;

if(lastindex+1==capacity)

cout<<endl<<"array is already full";

else if(index<0|| index>lastindex+1)

{

cout<<"index is not valid";

}

else

{

for(i=lastindex;i>index;i--)

ptr[i+1]=ptr[i];

ptr[index]=data;

lastindex++;

}

}

bool Array:: isEmpty()

{

return lastindex==-1;

}

void Array::append (int data)

{

if(lastindex+1==capacity)

cout<<endl<<"array is already full";

else

ptr[lastindex-1]=data;

}