1-search on a specific item in array

#include<iostream>

using namespace std;

int main()

{

int arr[10], tot=10, i, elem, j, found=0;

cout<<"Enter 10 Array Elements: ";

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

cin>>arr[i];

cout<<"\nEnter Element to Delete: ";

cin>>elem;

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

{

if(arr[i]==elem)

{

for(j=i; j<(tot-1); j++)

arr[j] = arr[j+1];

found++;

i--;

tot--;

}

}

if(found==0)

cout<<"\nElement doesn't found in the Array!";

else

cout<<"\nElement Deleted Successfully!";

cout<<endl;

return 0;

}

2-insert an element at a specific position in an array

// C++ Program to Insert an element

// at a specific position in an Array

#include <iostream>

using namespace std;

// Function to insert x in arr at position pos

int\* insertX(int n, int arr[],

            int x, int pos)

{

    int i;

    // increase the size by 1

    n++;

    // shift elements forward

    for (i = n; i >= pos; i--)

        arr[i] = arr[i - 1];

    // insert x at pos

    arr[pos - 1] = x;

    return arr;

}

// Driver Code

int main()

{

    int arr[100] = { 0 };

    int i, x, pos, n = 10;

    // initial array of size 10

    for (i = 0; i < 10; i++)

        arr[i] = i + 1;

    // print the original array

    for (i = 0; i < n; i++)

        cout << arr[i] << " ";

    cout << endl;

    // element to be inserted

    x = 50;

    // position at which element is to be inserted

    pos = 5;

    // Insert x at pos

    insertX(n, arr, x, pos);

    // print the updated array

    for (i = 0; i < n + 1; i++)

        cout << arr[i] << " ";

    cout << endl;

    return 0;

}

3-search any element or number in an array

// C++ Program to search any element or number in an array

#include <iostream>

using namespace std;

int main(){

    int input[100], count, i, num;

    cout << "Enter Number of Elements in Array\n";

    cin >> count;

    cout << "Enter " << count << " numbers \n";

    // Read array elements

    for(i = 0; i < count; i++){

        cin >> input[i];

    }

    cout << "Enter a number to serach in Array\n";

    cin >> num;

    // search num in inputArray from index 0 to elementCount-1

    for(i = 0; i < count; i++){

        if(input[i] == num){

            cout << "Element found at index " << i;

            break;

        }

    }

    if(i == count){

        cout  << "Element Not Present in Input Array\n";

    }

    return 0;

}

4-The stack by linked list

#include <iostream>

using namespace std;

struct Node {

int data;

struct Node \*next;

};

struct Node\* top = NULL;

void push(int val) {

struct Node\* newnode = (struct Node\*) malloc(sizeof(struct Node));

newnode->data = val;

newnode->next = top;

top = newnode;

}

void pop() {

if(top==NULL)

cout<<"Stack Underflow"<<endl;

else {

cout<<"The popped element is "<< top->data <<endl;

top = top->next;

}

}

void display() {

struct Node\* ptr;

if(top==NULL)

cout<<"stack is empty";

else {

ptr = top;

cout<<"Stack elements are: ";

while (ptr != NULL) {

cout<< ptr->data <<" ";

ptr = ptr->next;

}

}

cout<<endl;

}

int main() {

int ch, val;

cout<<"1) Push in stack"<<endl;

cout<<"2) Pop from stack"<<endl;

cout<<"3) Display stack"<<endl;

cout<<"4) Exit"<<endl;

do {

cout<<"Enter choice: "<<endl;

cin>>ch;

switch(ch) {

case 1: {

cout<<"Enter value to be pushed:"<<endl;

cin>>val;

push(val);

break;

}

case 2: {

pop();

break;

}

case 3: {

display();

break;

}

case 4: {

cout<<"Exit"<<endl;

break;

}

default: {

cout<<"Invalid Choice"<<endl;

}

}

}while(ch!=4);

return 0;

}

5-The queue by array

#include <iostream>

using namespace std;

int queue[100], n = 100, front = - 1, rear = - 1;

void Insert() {

int val;

if (rear == n - 1)

cout<<"Queue Overflow"<<endl;

else {

if (front == - 1)

front = 0;

cout<<"Insert the element in queue : "<<endl;

cin>>val;

rear++;

queue[rear] = val;

}

}

void Delete() {

if (front == - 1 || front > rear) {

cout<<"Queue Underflow ";

return ;

} else {

cout<<"Element deleted from queue is : "<< queue[front] <<endl;

front++;;

}

}

void Display() {

if (front == - 1)

cout<<"Queue is empty"<<endl;

else {

cout<<"Queue elements are : ";

for (int i = front; i <= rear; i++)

cout<<queue[i]<<" ";

cout<<endl;

}

}

int main() {

int ch;

cout<<"1) Insert element to queue"<<endl;

cout<<"2) Delete element from queue"<<endl;

cout<<"3) Display all the elements of queue"<<endl;

cout<<"4) Exit"<<endl;

do {

cout<<"Enter your choice : "<<endl;

cin>>ch;

switch (ch) {

case 1: Insert();

break;

case 2: Delete();

break;

case 3: Display();

break;

case 4: cout<<"Exit"<<endl;

break;

default: cout<<"Invalid choice"<<endl;

}

} while(ch!=4);

return 0;

}

#include <iostream>

using namespace std;

int main(){

    int input[100], count, i, num;

    cout << "Enter Number of Elements in Array\n";

    cin >> count;

    cout << "Enter " << count << " numbers \n";

    // Read array elements

    for(i = 0; i < count; i++){

        cin >> input[i];

    }

    cout << "Enter a number to serach in Array\n";

    cin >> num;

    // search num in inputArray from index 0 to elementCount-1

    for(i = 0; i < count; i++){

        if(input[i] == num){

            cout << "Element found at index " << i;

            break;

        }

    }

    if(i == count){

        cout  << "Element Not Present in Input Array\n";

    }

    return 0;

}

#include <iostream>

using namespace std;

// Function to insert x in arr at position pos

int\* insertX(int n, int arr[],

            int x, int pos)

{

    int i;

    // increase the size by 1

    n++;

    // shift elements forward

    for (i = n; i >= pos; i--)

        arr[i] = arr[i - 1];

    // insert x at pos

    arr[pos - 1] = x;

    return arr;

}

// Driver Code

int main()

{

    int arr[100] = { 0 };

    int i, x, pos, n = 10;

    // initial array of size 10

    for (i = 0; i < 10; i++)

        arr[i] = i + 1;

    // print the original array

    for (i = 0; i < n; i++)

        cout << arr[i] << " ";

    cout << endl;

    // element to be inserted

    x = 50;

    // position at which element is to be inserted

    pos = 5;

    // Insert x at pos

    insertX(n, arr, x, pos);

    // print the updated array

    for (i = 0; i < n + 1; i++)

        cout << arr[i] << " ";

    cout << endl;

    return 0;

}

Name\ Rawan Ibrahim hamed

Group\ G4