***//Assignment-1 Data Structures Lab***

***//A program to implement Linear and Binary search using template function***

#include <iostream>

using namespace std;

template <class T>

void Linear(T \*a, T item, int n)

{

int z=0;

for(int i=0;i<n;i++)

{

if(a[i]== item)

{

z=1;

cout<<"\n The item found at position "<<i+1<<"\n";

}

}

if (z==0)

cout<<"\n Item not found in the list\n";

}

template <class T>

void Binary(T \*a, T item, int n)

{

int beg=0,end=n-1;

int mid=(beg+end)/2;

while((a[mid]!=item) && (n>0))

{

if(item>a[mid])

beg=mid+1;

else

end=mid-1;

mid=(beg+end)/2;

n--;

}

if(a[mid]==item)

cout<<"\n Item found at position "<<mid+1<<"\n";

else

cout<<"\n Item not found in the list \n";

}

int main()

{

int a[5];

float b[5];

int item,op,i;

float item2;

while(1)

{

cout<<"\n 1. Linear Search \n 2. Binary Search \n 3. Exit \n";

cout<<"\n Enter your choice: ";

cin>>op;

switch(op)

{

case 1:

cout<<"\n Enter the 5 elements of Integer Array \n";

for(i=0;i<5;i++)

cin>>a[i];

cout<<"\n Enter an item to be search: ";

cin>>item;

Linear(a,item,5);

break;

case 2:

cout<<"\n Enter the 5 elements of Float Array \n";

for(i=0;i<5;i++)

cin>>b[i];

cout<<"\n Enter an item to be search: ";

cin>>item2;

Binary(b,item2,5);

break;

case 3:

exit (0);

}

}

return (0);

}

**OUTPUT:**

1. Linear Search

2. Binary Search

3. Exit

Enter your choice: 1

Enter the 5 elements of Integer Array

3

6

9

4

1

Enter an item to be search: 9

The item found at position 3

1. Linear Search

2. Binary Search

3. Exit

Enter your choice: 2

Enter the 5 elements of Float Array

1.2

2.3

3.4

4.5

5.6

Enter an item to be search: 2.3

Item found at position 2