Aim: write a programm of both recursive and non recursive of linear search....

Recursive linear function:-A function which calls itself

array list(12,61,33,92,36,3,29,98,54,60)

key elements to be searched is 36 and 100

Procedure:

step1:-Read the element to be searched from the user.

step2:-Compare the given element(36) with array list first element.

step3:-If both the elements are matched,then display the ouput as"Given element is found at index " " and terminate the function.

step4:-If both elements doesnt match,then display the output as "given element is not found and terminate the function.

step5:-Then compare the search element with the second element in the given list.

step5:-Repeat the step 3,step 4 and step 5 until search element is found in the list and terminate the function.

step6:-return search is unseccessful if not found.

step6:-In this case it will print that search is successful at position 5.

#include<stdio.h>

int ReLinearSearch(int [],int,int);

int main()

{

int i,position,key;

int a[10]= {12,61,33,92,36,3,29,98,54,60};

scanf("%d",&key);

position = ReLinearSearch(a,10,key);

if (position == -1){

printf("search is unsucessful");

}

else{

printf("search is sucessful at position %d",position + 1);

}

return 0;

}

int ReLinearSearch(int a[],int n,int key)

{

if (n > 0){

if(a[n - 1] == key){

return n-1;

}

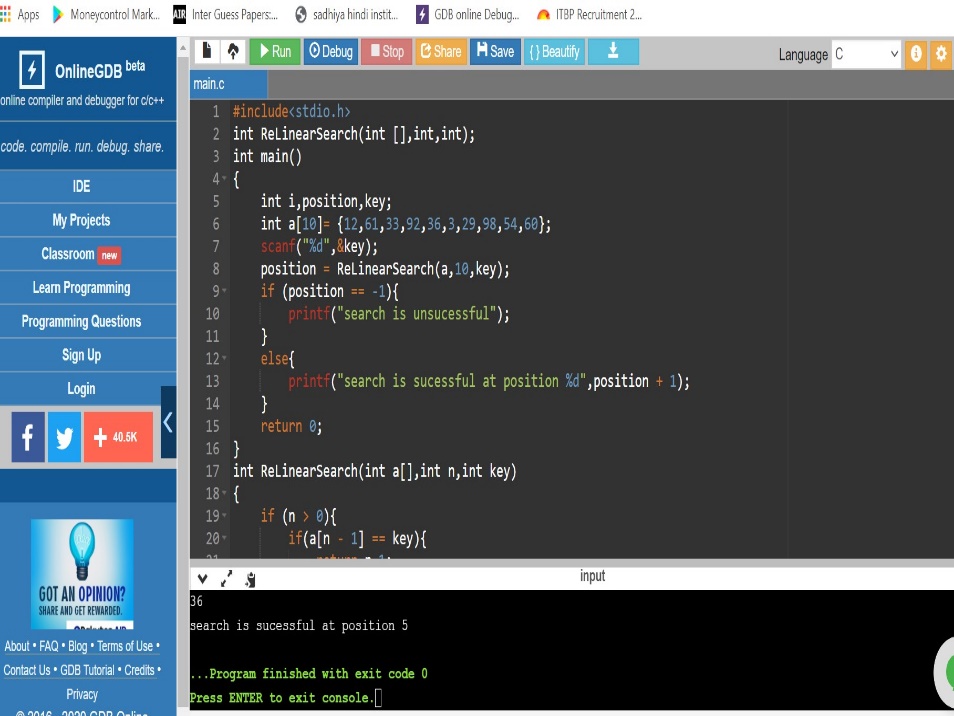
else{

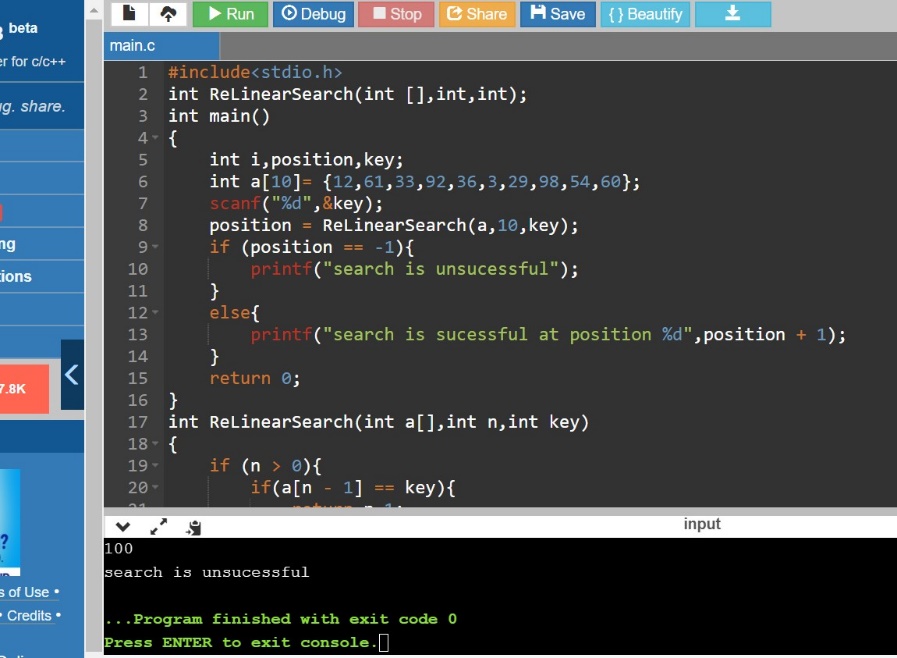
return ReLinearSearch(a,n-1,key);

}

}

return -1;

}



Iterative linear

step1:-Read the element to be searched from the user.

step2:-Compare the given element(36) with array list first element.

step3:-If both the elements are matched,then display the ouput as"Given element is found at index " " and terminate the function.

step4:-If both elements doesnt match,then display the output as "given element is not found and terminate the function.

step5:-Then compare the search element with the second element in the given list.

step5:-Repeat the step 3,step 4 and step 5 until search element is found in the list and terminate the function.

step6:-if not found return 0.

step7:-In this case the element is found at index 4.

#include<stdio.h>

int main()

{

int i,n,key;

printf("enter any one value");

scanf("%d",&key);

n=10;

int a[] = {12,61,33,92,36,3,29,98,54,60};

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

{

if(a[i] == key)

{

printf("%d",i);

break;

}

}

if(i==n)

{

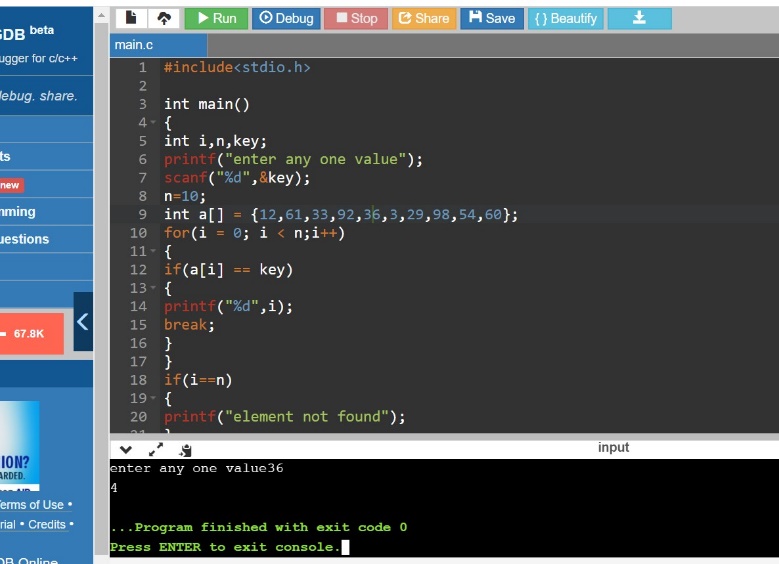
printf("element not found");

}

return 0;

}





Aim: write a programm of both recursive and non recursive of binary search...

Iterative Binary Search

array list(12,61,33,92,36,3,29,98,54,60)

key element to be searched is 12 ,92 and 33

procedure:

step1:-Read the element to be searched from the user(92).

step2:-Compare the given element with the middle element in the given array list.

ie. Mid = (i + n)/2

step3:-.In the given array mid value is (0 + 9)/2 =4.5 and ignoring the decimal value we

will take the position as 4 the and the element present at 4 the is 36 .

step4:-We will use loop iteration process if the key element position is equal to the

given mid position element then the output will be element found at 4 the position (mid position in this array).

step5:-.If the key element is not equal to the mid element then there are two ways to get

the output

1: If the key element is < than the mid position element then i = 0 and n =

mid - 1. And the same loop iteration continues until we get the position of the

given element.

2: If the key element is > than the mid position element then i = mid + 1 and n

= 9.

step6:-In this case, the element is found at index 4.

#include <stdio.h>

int binarySearch(int arr[], int l, int r, int x)

{

while (l <= r)

{

int m = l + (r-l)/2;

if (arr[m] == x)

return m;

if (arr[m] < x)

l = m + 1;

else

r = m - 1;

}

return -1;

}

int main(void)

{

int arr[] = {3,12,29,33,36,54,60,61,92,98};

int n = sizeof(arr)/ sizeof(arr[0]);

int x = 12;

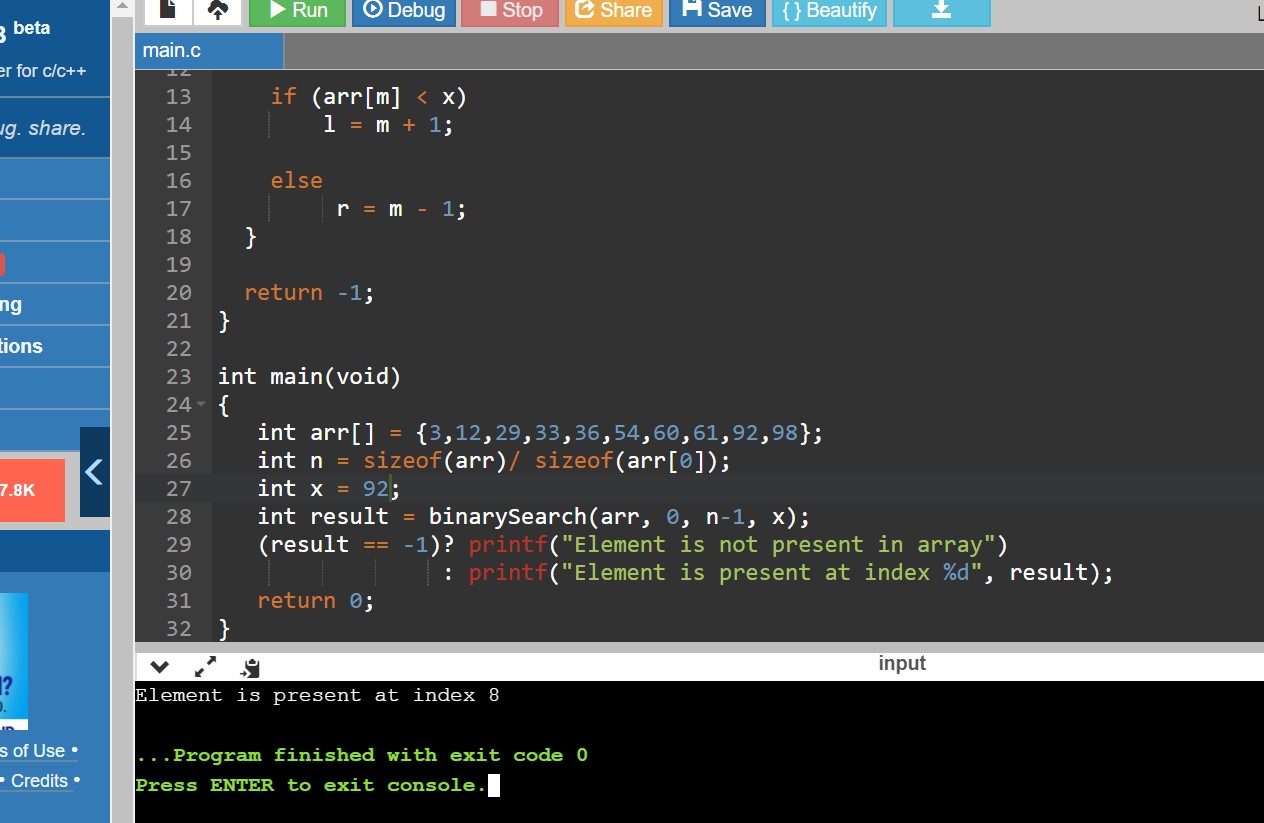
int result = binarySearch(arr, 0, n-1, x);

(result == -1)? printf("Element is not present in array")

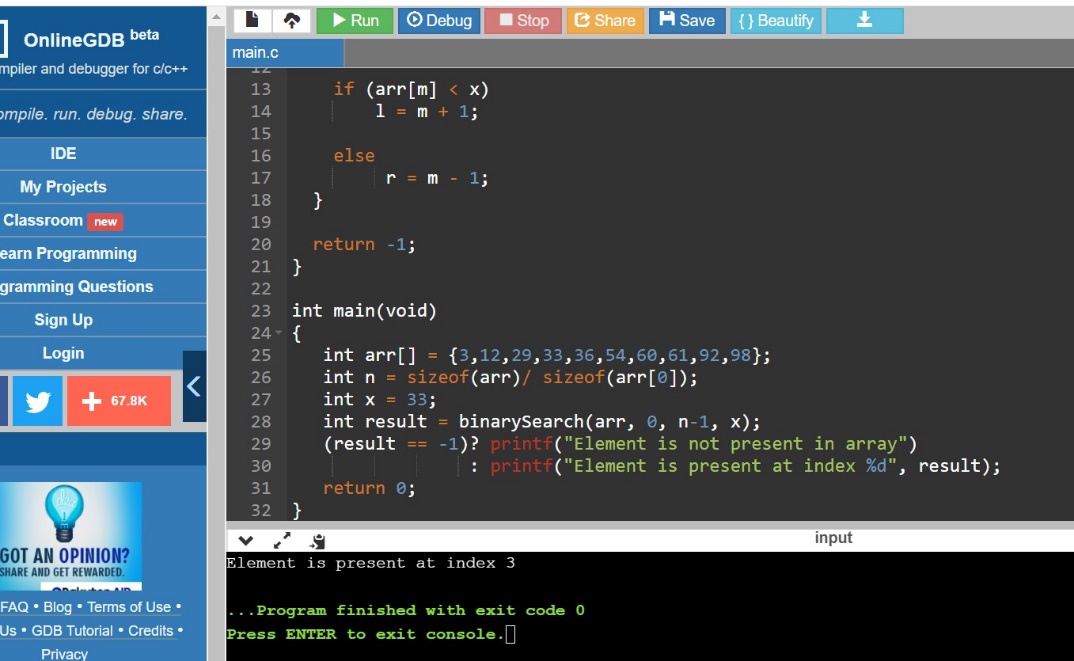
: printf("Element is present at index %d", result);

return 0;

}







Recursive Binary search

array list(12,61,33,92,36,3,29,98,54,60)

key element to be searched is 12 ,92 and 33

procedure:

step1:-we need to take the key element to be searched from the user.

step2:-check whether the key element matches with the middle element.

step3:-if key=mid then return the mid value displaying key found at index and terminate the function.

step4:-if key<mid then new value =mid-1

step5:-if key>mid our new initial value will be = mid+1.

step6:-repeat the process until we find our key element.

step7:-in our process our key is 92.

step8:-key found at index 4.

#include<stdio.h>

int BinarySearch(int [],int,int,int);

int main()

{

int i,key,position;

int a[10] = {3,12,29,33,36,54,60,61,92,98};

scanf("%d",&key);

position = BinarySearch(a,0,10,key);

if (position == -1){

printf("search is unsucessfull");

}

else{

printf("search is sucessfull at position %d",position + 1);

}

return 0;

}

int BinarySearch(int a[],int first,int last,int key)

{

int mid;

while(first <= last)

{

mid = (first + last) / 2;

if (a[mid] == key)

return mid;

else if (key > a[mid])

first = mid + 1;

else

last = mid - 1;

}

return -1;

}

