

```

1  #include <stdio.h>
2  #include <time.h>
3  clock_t start,end;
4  int linear_search(int arr[],int n,int k)
5  {
6  if(n<0)
7  return 0;
8  if(arr[n] ==k)
9  return 1;
10 return linear_search(arr,n-1,k);
11 }
12 int binary_search(int arr[],int start,int end,int k)
13 {
14 if(start>end)
15 return 0;
16 int mid =(start+end)/2;
17 if(arr[mid]==k)
18 return 1;
19 else if(arr[mid]>k)
20 end=mid-1;
21 else
22 start=mid+1;
23 return binary_search(arr,start,end,k);
24 }
25 int main()
26 {

```

```

27  int n,i;
28  printf("Enter the size of array: ");
29  scanf("%d",&n);
30  int choice;
31  int arr[n],k;
32  printf("The elements of array: \n");
33  for(i=0;i<n;i++)
34
35  {
36  arr[i] = i+1;
37  scanf("%d",&arr[i]);}
38  printf("\n");
39  printf("Enter the element to be searched: \n");
40  scanf("%d",&k);
41  while(choice!=3)
42  {
43  printf("\n 1.Linear search");
44  printf("\n 2.Binary search");
45  printf("\n 3.Exit");
46  printf("\nEnter your choice: ");
47  scanf("%d",&choice);
48  if (choice==1)
49  {
50  start=clock();
51  for(i=0;i<800000000;i++);

```



```

1.c
2 int a = linear_search(arr,n-1,k);
3 end = clock();
4 if(a==1)
5 printf("Element found");
6 else
7 printf("Element not found\n");
8 printf("\nTime taken to search value in %d numbers is %f secs\n",n,(((double)(end-
9 }
10 else if(choice == 2)
11 {
12 start=clock();
13 for(i=0;i<80000000;i++);
14 int b=binary_search(arr,0,n-1,k);
15 end = clock();
16 if (b==1)
17 printf("Element found");
18 else
19 printf("Element not found");
20 printf("\n Time taken to search value in %d numbers is %f secs\n",n,(((double)(end-
21 }
22 else if (choice==3)
23 break;
24 else
25 printf("Invalid choice: ");
26 }
27 }

```

input

* Linear and Binary search:

```
#include <stdio.h>
```

```
#include <time.h>
```

```
clock_t start, end;
```

```
int linear-search (int arr[], int n, int k)
```

```
{
```

```
    if (n < 0)
```

```
        return 0;
```

```
    if (arr[n] == k)
```

```
        return 1;
```

```
    return linear-search (arr, n-1, k);
```

```
}
```

```
int binary-search (int arr[], int start, int end,
```

```
int k)
```

```
{
```

```
    if (start > end)
```

```
        return 0;
```

```
    int mid = (start + end) / 2;
```

```
    if (arr[mid] == k)
```

```
        return 1;
```

```
    else if (arr[mid] > k)
```

```
        end = mid - 1;
```

```
    else
```

```
        start = mid + 1;
```

```
    return binary-search (arr, start, end, k);
```

```
}
```

int main()

{

int n, i;

printf("Enter the size of array: ");

scanf("%d", &n);

int choice;

int arr[n], k;

printf("The elements of array: \n");

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

{

arr[i] = i+1;

printf("%d", arr[i]);

· y

printf(" \n");

printf("Enter the element to search: ");

scanf("%d", &k);

while (choice != 3)

{

printf("1. Linear search");

printf("2. Binary search");

printf("3. Exit \n");

printf("Enter your choice ");

scanf("%d", &choice);

if (choice == 1)

{


```
start = clock();
```

```
for (i=0; i<800000000; i++);
```

```
int a = linear_search(arr, n-1, k);
```

```
end = clock();
```

```
if (a == 1)
```

```
printf("Element found\n");
```

```
else
```

```
printf("Element not found");
```

```
printf("Time taken to search value in %d  
numbers is %f secs", n, ((double)(end-start))/
```

```
CLOCKS_PER_SEC));
```

```
y
```

```
else if (choice == 2)
```

```
{
```

```
start = clock();
```

```
for (i=0; i<800000000; i++);
```

```
int b = binary_search(arr, 0, n-1, k);
```

```
end = clock();
```

```
if (b == 1)
```

```
printf("Element found\n");
```

```
else
```

```
printf("Time taken to search value
```

```
in %d number is %f secs\n", n,
```

```
((double)(end-start))/CLOCKS_PER-
```

```
SEC)); y
```

```
else if (choice == 3)
```

```
break;
```

```
else
```

```
printf("Invalid choice: ");
```

```
int a = linear_search(arr, n-1, k);
```

```
end = clock();
```

```
if (a == 1)
```

```
printf("Element found ");
```

```
else
```

```
printf("Element not found");
```

```
printf("In Time taken to search value  
in %d number is %.f sec\n", ((double)  
(end - start)) / (CLOCKS_PER_SEC));
```

```
}
```

```
else if (choice == 2)
```

```
{
```

```
start = clock();
```

```
for (i = 0; i < 800000000; i++);
```

```
int b = binary_search(arr, 10, n-1, k);
```

```
end = clock();
```

```
if (b == 1)
```

```
printf("Element found\n");
```

```
else
```

```
printf("Element not found\n");
```

```
printf("\n Time taken to search value in  
%d numbers is %.f secs\n", n, ((double)(end  
start)) / (CLOCKS_PER_SEC));
```

```
    }  
    else if (choice == 3)
```

```
        break;
```

```
    else
```

```
        printf("Invalid choice.");
```

```
    }
```


Enter the size of array: 4

The elements of array:

34 56 12 78

Enter the element to be searched:

56

1.Linear search

2.Binary search

< 3.Exit

Enter your choice: 2

Element found

Time taken to search value in 4 numbers is 0.169669 secs

1.Linear search

2.Binary search

3.Exit

Enter your choice: