**Implement Recursive Binary search and Linear search and determine the time required to search an element. Repeat the experiment for different values of N and plot a graph of the time taken versus N.**

**Binary search**

#include <stdio.h>

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

{

if (r >= l)

{

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

if (arr[mid] == x) return mid;

if (arr[mid] > x) return binarySearch(arr, l, mid-1, x);

return binarySearch(arr, mid+1, r, x);

}

return -1;

}

int main(void)

{

int arr[10] ;//must be sorted

int i,x,n;

printf("enter the number of elements in the array:\n");

scanf("%d",&n);

printf("Enter the elements of the array in sorted order\n");

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

scanf("%d",&arr[i]);

}

printf("Enter the element you want to search:\n");

scanf("%d",&x);

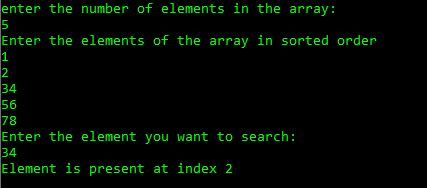
int result = binarySearch(arr, 0, n-1, x);

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

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

return 0;

}



**Linear Search**

#include <stdio.h>

int RecursiveLS(int arr[], int value, int index, int n)

{

int pos = 0;

if(index >= n)

{

return 0;

}

else if (arr[index] == value)

{

pos = index + 1;

return pos;

}

else

{

return RecursiveLS(arr, value, index+1, n);

}

return pos;

}

int main()

{

int n, value, pos, arr[100];

printf("Enter the total elements in the array ");

scanf("%d", &n);

printf("Enter the array elements\n");

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

{

scanf("%d", &arr[i]);

}

printf("Enter the element to search ");

scanf("%d", &value);

pos = RecursiveLS(arr, value, 0, n);

if (pos != 0)

{

printf("Element found at pos %d ", pos);

}

else

{

printf("Element not found");

}

return 0;

}

