Aim: Generate a large number of element randomly and search a given number from it using sequential search and binary search. Analyze the time complexity for best, average and worst case.

Sequential Search :

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/time.h>

int sequentialSearch(int list[],int totalElement,int findNum){
    int i;
    for(i=0;i<totalElement;i++){
        if(list[i]==findNum)
            break;
    }
    return i;
}


void main(){
    int num,i,index;
    int *array;
    int t1,t2;
    float t3,t4;
    int tt1,tt2;
    struct timeval tv;
    struct timezone tz;

    printf("\nWhat value you want to find : ");
    scanf("%d",&num);

    array = (int*)malloc(100000*sizeof(int));

    for(i=0 ; i < 100000 ;i++)
        array[i]=i+1;

    gettimeofday(&tv,&tz);
    tt1 = tv.tv_sec;
    t1 = tv.tv_usec;
```

```
    index = sequentialSearch(array,100000,num);

    gettimeofday(&tv,&tz);
    tt2 = tv.tv_sec;
    t2 = tv.tv_usec;

    tt1 = tt1%100;
    t3 = (float)t1*(0.000001);
    t3 = t3 + (float)tt1;
    tt2 = tt2%100;
    t4 = (float)t2*(0.000001);
    t4 = t4 + (float)tt2;

    printf("\nStart : %d.%d",tt1,t1);
    printf("\nEnd   : %d.%d",tt2,t2);

    if(t4>t3)
        printf("\nDifferenc1 : %f",t4-t3);
    else
        printf("\nDifferenc2 :  %f",t3-t4);

    free(array);
}
```

Binary Search :

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/time.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;
}

void main(){
    int num,i,index,n=100000;
    int *array;
    int t1,t2;
    float t3,t4;
    int tt1,tt2;
    struct timeval tv;
    struct timezone tz;

    printf("\nWhat value you want to find : ");
    scanf("%d",&num);

    array = (int*)malloc(n*sizeof(int));

    for(i=0 ; i < n ;i++)
        array[i]=i+1;

    gettimeofday(&tv,&tz);
    tt1 = tv.tv_sec;
    t1 = tv.tv_usec;

    index = binarySearch(array,0,n-1,num);

    gettimeofday(&tv,&tz);
    tt2 = tv.tv_sec;
    t2 = tv.tv_usec;
```

```
    tt1 = tt1%100;
    t3 = (float)t1*(0.000001);
    t3 = t3 + (float)tt1;
    tt2 = tt2%100;
    t4 = (float)t2*(0.000001);
    t4 = t4 + (float)tt2;

    printf("\nStart : %d.%d",tt1,t1);
    printf("\nEnd   : %d.%d",tt2,t2);

    if(t4>t3)
        printf("\nDifferenc1 : %f",t4-t3);
    else
        printf("\nDifferenc2 :  %f",t3-t4);

    free(array);
}
```

Time Analysis :

Table :

Search	Best Case	Average Case	Worst Case
Sequential Search	0	0.001000	0.001003
Binary Search	0	0.000997	0.001002

Graph :