```
// linear and binary search calculate time taken for different values of n

#include<stdio.h>
#include<conio.h>
#include<process.h>
#include<stdlib.h>
#include<time.h>

int linear(int i,int n,int arr[])
{
	int item=200000;

	if(i<n)
		{
			if(arr[i]==item)
				return(i+1);
			else
				linear(++i,n,arr);
		}
		else
			return -1;

}

int binary(int l,int u,int mid,int n,int arr[])
{
	int item=500000;

	if(l<=u)
	{
		if(arr[mid]==item)
		{
			return mid+1;
		}
		else if(arr[mid]<item)
			l=mid+1;
		else
			u=mid-1;

		mid=(l+u)/2;
```

```
                binary(l,u,mid,n,arr);
        }
        else
                return -1;

}

void main()
{
        int ch,n,a[150000],b[150000],flag=-1,l,u,mid;
        clock_t start,end;
        for(;;)
        {
                printf("1.linear search \n 2.binary seach \n else exit \n");
                scanf("%d",&ch);
                switch(ch)
                {
                        case 1:n=5000;
                        printf("time taken by linear search for different values of n: \n");
                        while(n<=145000)
                        {
                                for(int i=0;i<n;i++)
                                        a[i]=i;

                                start=clock();
                      flag=linear(0,n,a);
                                /*if(flag>0)
                printf("value found at position %d \n",flag);
           else
                printf("value not found\n");*/
           //delay
                for(int j=0;j<=100;j++);

                end=clock();

                printf("time taken by %d elements = %f secs
\n",n,((double)(end-start))/CLOCKS_PER_SEC);
                n=n+10000;
                        }
```

```
                        break;

                        case 2:n=5000;
                        printf("time taken by binary search for different values of n: \n");
                        while(n<=145000)
                        {
                                for(int i=0;i<n;i++)
                                        b[i]=i;

                                start=clock();
                                l=0;
                                u=n-1;
                                mid=(l+u)/2;
                        flag=binary(l,u,mid,n,b);

                /*if(flag>0)
                printf("value found at position %d\n",flag);
            if(flag==-1)
                printf("value not found\n");*/

            //delay
                for(int j=0;j<=100;j++);

                end=clock();

                printf("time taken by %d elements = %f secs
\n",n,((double)(end-start))/CLOCKS_PER_SEC);
                n=n+10000;
                        }
                        break;

                        default:exit(0);
                }
        }
}
```

```
1.linear search
 2.binary seach
 else exit
1
time taken by linear search for different values of n:
time taken by 5000 elements = 0.000073 secs
time taken by 15000 elements = 0.000195 secs
time taken by 25000 elements = 0.000319 secs
time taken by 35000 elements = 0.000379 secs
time taken by 45000 elements = 0.000578 secs
time taken by 55000 elements = 0.000581 secs
time taken by 65000 elements = 0.000898 secs
time taken by 75000 elements = 0.000778 secs
time taken by 85000 elements = 0.000886 secs
time taken by 95000 elements = 0.000990 secs
time taken by 105000 elements = 0.001318 secs
time taken by 115000 elements = 0.001397 secs
time taken by 125000 elements = 0.001400 secs
time taken by 135000 elements = 0.001728 secs
time taken by 145000 elements = 0.001680 secs
1.linear search
```

```
1.linear search
 2.binary seach
 else exit
2
time taken by binary search for different values of n:
time taken by 5000 elements = 0.000003 secs
time taken by 15000 elements = 0.000002 secs
time taken by 25000 elements = 0.000001 secs
time taken by 35000 elements = 0.000001 secs
time taken by 45000 elements = 0.000002 secs
time taken by 55000 elements = 0.000002 secs
time taken by 65000 elements = 0.000004 secs
time taken by 75000 elements = 0.000002 secs
time taken by 85000 elements = 0.000002 secs
time taken by 95000 elements = 0.000002 secs
time taken by 105000 elements = 0.000002 secs
time taken by 115000 elements = 0.000001 secs
time taken by 125000 elements = 0.000002 secs
time taken by 135000 elements = 0.000002 secs
time taken by 145000 elements = 0.000002 secs
```