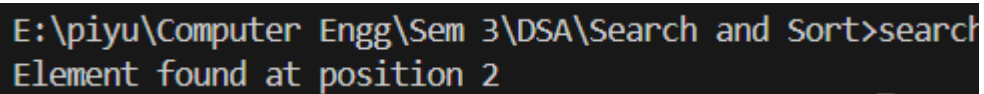


## Binary Search

### Code:

```
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
void main(){
    int a[]={1,15,78,90,94,102,113};
    int ub=11,mid,key=15,lb=0;
    while(lb<=ub){
        mid=(lb+ub)/2;
        if(key==a[mid]){
            printf("Element found at position %d",mid+1);
            exit(0);
        }
        else if(a[mid]>key)
            ub=mid-1;
        else
            lb=mid+1;
    }
    printf("Element not found");
}
```

### Output:



```
E:\piyu\Computer Engg\Sem 3\DSA\Search and Sort>search
Element found at position 2
```

## Fibonacci Search

### Code:

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

int min(int a, int b) {
    return (a < b) ? a : b;
}

int fibonacciSearch(int a[], int n, int key) {
    int fib2 = 0;
    int fib1 = 1;
    int fib = fib1 + fib2;

    while (fib < n) {
        fib2 = fib1;
        fib1 = fib;
        fib = fib1 + fib2;
    }

    int offset = -1;
    while (fib > 1) {
        int i = min(offset + fib2, n - 1);

        if (a[i] < key) {
            fib = fib1;
            fib1 = fib2;
            fib2 = fib - fib1;
            offset = i;
        }
        else if (a[i] > key) {
            fib = fib2;
            fib1 = fib1 - fib2;
            fib2 = fib - fib1;
        }
    }
```

```

        else {
            return i;
        }
    }
    if (fib1 && a[offset + 1] == key)
        return offset + 1;
    return -1;
}

void main() {
    int arr[] = {10, 22, 35, 40, 45, 50, 80, 82, 85, 90};
    int n = sizeof(arr) / sizeof(arr[0]);
    int key = 85;
    int index = fibonacciSearch(arr, n, key);
    if (index >= 0)
        printf("Element found at position %d", index + 1);
    else
        printf("Element not found");
}

```

**Output:**

```

E:\piyu\Computer Engg\Sem 3\DSA\Search and Sort>search
Element found at position 9

```