Age | 5 | 11 | 14 | 13 | 22 | 7 | 20 |

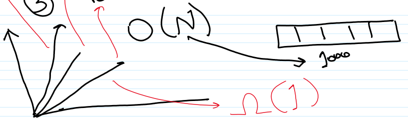
Array 7

Dictionary
2000 Page

25 element

1. Worst Case — O → big O Notation
2. Average case — Θ
3. Best case — Ω

O(N)



1000

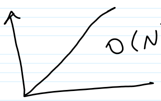
Ω(1)

*PseudoCode

1. For i in sizeof arr
2. check if arr[i]==Number
   Cout<<i;
   Return i;

Cout<<"not found"
Return 0;

linear search
Simple search



O(N)

```cpp
#include <iostream>

using namespace std;

int main()
{
    int arr[]{1,2,3,4,5,6,7,8,9,10};
    int number=6;
    for(int i=0;i<10;i++){
        if(arr[i]==number){
            cout<<"index:"<<i<<"\n";
            return i;
        }

    }
    cout<<"Not found\n";
    return 0;

}
```

The order of the array doesn't Matter

binary Search → key

A b c D e f G h i j k l m n o p q r z

500 250 250 125 175

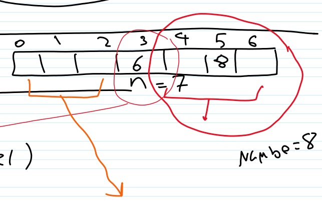
left side Right side

$O(\log_2 n)$

$\log_{10} 100 = 2$ exponential $10^2 = 100$

PseudoCode

→ low=0
high=n-1
mid=(0+6)/2 =3
if (arr[mid]==Number)
   Return mid
elsif Number > arr[mid]:
   low=mid+1
else Number < arr[mid]
   high=mid-1



Number=8

7^{12}

sizearray → sizeof(arr) → 48 bytes
sizearray → sizeof(arr[0]) — int=4bytes

48/4 = 12

```cpp
int arr[]{1,2,3,4,5,6,7,8,9,10,11,12};
int n=sizeof(arr)/sizeof(arr[0]);
cout<<sizeof(arr)<<endl;
cout<<sizeof(arr[0])<<endl;
```

```
Select "C:\Users\THE LAPTOP SHOP\Desktop\Intro To Software Engineering\Source Code\
48
4
```

```cpp
int main()
{
    int Searchednumber=8;

    int arr[]{1,2,3,4,5,6,7,8,9,10,11,12};
    int n=sizeof(arr)/sizeof(arr[0]);
    int low,high,middle;
    low=0;
    high=n-1;
    while(low<=high){
        middle=(low+high)/2;
    if(arr[middle]==Searchednumber){
        cout<<"The idex is:"<<middle<<endl;
        return middle;
    }
    if(Searchednumber>arr[middle]){
        cout<<"Now I am in the right side"<<endl;
        low=middle+1;
    }
    else{
        cout<<"Now I am in the left side"<<endl;
        high=middle-1;
```