Linear Search

Algorithm, Best case, worst case and average case

Linear Search

- Finding elements in a single dimensional array
- Elements may be in any order

23	44	12	66	10	55
1	2	3	4	5	6

- low = 1; high = 6;   keyvalue =66;

23	44	12	66	10	55
1	2	3	4	5	6

```
Algorithm linearsearch(a, n, keyvalue)
{
      low = 1; high = 6; found=0;
      while(low <= high and found==0)
      {
         if(a[low] == keyvalue)
            found = 1;
        else
            low = low + 1;
      }
      If (found == 0)
        return 0;
      else
         return low;
}
```

Linear search

- main()
- {
    - // read number of elements, array elements and the keyvalue
- t = linearsearch(a, n, keyvalue);
    - if(t ==0)
        - Display “element does not exist”
        - Else
        - Display “element exist”
- }

Linear Search

- Keyvalue = 23; How many comparisons are required?
- Best case =1

23	44	12	66	10	55
1	2	3	4	5	6

- Keyvalue =55; How many comparisons are required?
- Worst case = 6;
- What is average case?

Linear Search

- Average Case
- Keyvalues = 23, 44, 12, 66, 10, 55

23	44	12	66	10	55
1	2	3	4	5	6

- Average case = (Total number of comparisons)/n
- = (1+2+3+4+5+6)/6 = 21/6 = 3.5
- Total number of comparisons = n(n+1)/2
- Average = [n(n+1)/2] /n = (n+1)/2