

MODUL PRAKTIKUM – Search and Sort

A. Tujuan

Setelah mempelajari materi kegiatan pembelajaran ini mahasiswa akan dapat :

- 1) Memahami konsep pencarian data
- 2) Dapat mengembangkan konsep pencarian data
- 3) Dapat menggunakan konsep pencarian data dengan benar
- 4) Memahami konsep mengurutkan data
- 5) Dapat mengembangkan konsep mengurutkan data
- 6) Dapat menggunakan konsep mengurutkan data dengan benar

B. Materi

- **Search (mencari)**
 - Linear/Sequential Search
 - Mampu untuk data yang acak
 - Secara umum lebih lambat dari binary search
 - Binary Search
 - Hanya untuk data yang sudah diurutkan
 - Secara umum lebih cepat dari linear search
- **Sort (mengurutkan)**

➤ Contoh Linear/Sequential Search (1)

```

1  #include <iostream>
2  using namespace std;
3
4  int main () {
5      int data[5]={20,37,42,77,89};
6      int cari, ditemukan=0;
7
8      cout<<"Data Array\n";
9      for (int i=0; i<5; i++){
10         cout<<data[i]<<" ";
11     }
12
13     cout<<"\nMasukan nilai yg anda cari  : ";
14     cin>>cari;
15
16     for (int i=0; i<5; i++) {
17         if (data[i] == cari) {
18             ditemukan = 1;
19             cout<<"Nilai yg dicari terdapat pada data ke "<<i+1;
20             break;
21         }
22     }
23     if (ditemukan == 0) {
24         cout<<"Nilai yang dicari tidak ditemukan";
25     }
26 }

```

➤ Contoh Linear/Sequential Search (2)

```

1  #include <iostream>
2  using namespace std;
3  int main () {
4      int data[10]={20,37,42,20,89,20,37,11,77,92};
5      int cari, ditemukan=0, hasilcari[10];
6      cout<<"Data Array\n";
7      for (int i=0; i<10; i++){
8         cout<<data[i]<<" ";
9     }
10     cout<<"\nMasukan nilai yg anda cari  : ";
11     cin>>cari;
12     for (int i=0; i<10; i++) {
13         if (data[i] == cari) {
14             ditemukan++;
15             hasilcari[ditemukan] = i;
16         }
17     }
18     if (ditemukan == 0) {
19         cout<<"Nilai yang dicari tidak ditemukan";
20     } else {
21         cout<<"Nilai yg dicari terdapat pada data ke "<<endl;
22         for (int i=1; i<=ditemukan; i++){
23             cout<<hasilcari[i]+1<<" ";
24         }
25     }
26 }

```

➤ Contoh Binary Search

```

1  #include <iostream>
2  using namespace std;
3  main() {
4      int target;
5      int data[10]={11,22,33,44,55,66,77,88,99,100};
6      int low, mid, high, ditemukan=0;
7      cout<<"Data Array\n";
8      for (int i=0; i<10; i++){
9          cout<<data[i]<<" ";
10     }
11     cout<<"\nMasukan angka yang dicari: ";
12     cin>>target;
13     low=0;
14     high=9;
15     while(low<=high){
16         mid=(low+high)/2;
17         if(target==data[mid]){
18             ditemukan = 1;
19             cout<<"Nilai yg dicari terdapat pada data ke "<<mid+1;
20             break;
21         } else if(target>data[mid]){
22             low=mid+1;
23         } else if(target<data[mid]){
24             high=mid-1;
25         }
26     }
27     if (ditemukan == 0) {
28         cout<<"Nilai yang dicari tidak ditemukan";
29     }
30 }

```

➤ Contoh Sort

```
1  #include<iostream>
2  using namespace std;
3  int main() {
4      int data[10], i, j, tmp;
5      cout<<"Program Mengurutkan Angka Kecil Ke Besar"<<endl;
6      for(i=0; i<10; i++){
7          cout<<"Masukkan Angka ke "<<(i+1)<<" : ";
8          cin>>data[i];
9      }
10     cout<<"Data sebelum diurutkan : "<<endl;
11     for(i=0; i<10; i++){
12         cout<<data[i]<<" ";
13     }
14
15     for(i=0; i<9; i++){
16         for(j=i+1; j<10; j++){
17             if(data[i]>data[j]){
18                 tmp = data[i];
19                 data[i] = data[j];
20                 data[j] = tmp;
21             }
22         }
23     }
24
25     cout<<"\nData setelah diurutkan : "<<endl;
26     for(i=0; i<10; i++){
27         cout<<data[i]<<" ";
28     }
29 }
```

Good Luck !