

Array Searching and Sorting

Nama : Dani Ramadhan
NIM : 2504130157
Program Studi : Teknik Informatika
Rombel : 4
Mata Kuliah : Dasar Pemrograman
Dosen Pengampu : Anggyi Trisnawan Putra, S.Si., M.Si.

Deskripsi Tugas :

Pelajari dan praktikkan algoritma sorting dan searching menggunakan C++, dengan ketentuan dibawah ini. Yang dikumpulkan adalah deskripsi algoritmanya, penerapan C++, dan tampilan hasilnya (PDF).

Array yang dijadikan sample:

Nilai yang dimasukan = 85, 72, 90, 65, 88, 92, 78, 95, 70 dan 83.

Untuk searching, semua sama, cari nilai 88.

NIM berakhiran 6 atau 7:

- Sorting: Bubble Sort (Descending)
- Search: Linear Search dari Belakang

Jawaban :

Deskripsi Algoritma :

Sorting: Bubble Sort (Descending)

Algoritma Bubble Sort Descending adalah metode pengurutan yang menyusun data dari nilai terbesar ke terkecil dengan cara membandingkan dua elemen bersebelahan dan menukarannya bila urutannya tidak sesuai.

Search: Linear Search dari Belakang

Linear Search dari Belakang adalah metode pencarian sederhana yang memeriksa setiap elemen dalam array mulai dari elemen terakhir hingga ke awal.

Penerapan dalam C++ :

```
1 #include <iostream>
2 using namespace std;
3
4 int main () {
5     int nilai[] = {85, 72, 90, 65, 88, 92, 78, 95, 70, 83};
6     int n = sizeof(nilai) / sizeof(nilai[0]);
7     int cari = 88;
8
9     // Bubble short Descending
10    for (int i = 0; i < n - 1; i++) {
11        for (int j = 0; j < n - i - 1; j++) {
12            if (nilai[j] < nilai[j + 1])
13            {
14                int temp = nilai[j];
15                nilai[j] = nilai[j + 1];
16                nilai[j + 1] = temp;
17            }
18        }
19    }
20
21    cout << "Hasil Bubble Sort (Descending): ";
22    for (int i = 0; i < n; i++) {
23        cout << nilai[i] << " ";
24    }
25    cout << endl;
26
27    // Linear Search dari Belakang
28    int posisi = -1;
29    for (int i = n - 1; i ≥ 0; i--)
30    {
31        if (nilai[i] == cari) {
32            posisi = i;
33            break;
34        }
35    }
36    cout << "Mencari nilai " << cari << " ..." << endl;
37    if (posisi ≠ -1)
38        cout << "Ditemukan di indeks ke-" << posisi << endl;
39    else
40        cout << "Tidak ditemukan." << endl;
41
42    return 0;
43 }
```

Hasil Output :

- Hasil Bubble Sort (Descending): 78 85 90 88 92 72 95 70 65 83
Mencari nilai 88 ...
Ditemukan di indeks ke-3

Penjelasan :

Karena menggunakan algoritma Bubble Sort Descending, maka data akan diurutkan dari nilai terbesar ke terkecil. Prosesnya dilakukan dengan membandingkan dua elemen bersebelahan dan menukar jika urutannya salah. Setelah proses pengurutan selesai, pencarian dilakukan dengan metode Linear Search dari belakang, yaitu memeriksa elemen array mulai dari posisi terakhir hingga ke awal untuk menemukan nilai 88.

Cara Kerjanya:

Langkah	Elemen Dibandingkan	Hasil Setelah Langkah Ini
1	85 & 72	90, 85, 72, 65, 88, 92, 78, 95, 70, 83
2	88 & 92	90, 85, 88, 92, 78, 95, 72, 70, 83, 65
3	95 & 92	95, 92, 90, 88, 85, 83, 78, 72, 70, 65
4	92 & 90	95, 92, 90, 88, 85, 83, 78, 72, 70, 65
5	90 & 88	95, 92, 90, 88, 85, 83, 78, 72, 70, 65

Keterangan:

Pada setiap langkah, dua elemen dibandingkan dan ditukar jika elemen kiri lebih kecil dari elemen kanan (karena pengurutan descending). Setelah beberapa iterasi, nilai terbesar akan "menggelembung" ke posisi paling depan hingga seluruh data tersusun dari terbesar ke terkecil.

Penjelasan Pencarian (Linear Search dari Belakang):

Setelah data diurutkan secara descending menjadi:

95, 92, 90, 88, 85, 83, 78, 72, 70, 65

Langkah pencarian dimulai dari elemen terakhir menuju awal array.

Prosesnya:

- Bandingkan indeks ke-9 (65) dengan nilai 88 → tidak sama
- Bandingkan indeks ke-8 (70) → tidak sama
- Bandingkan indeks ke-7 (72) → tidak sama
- Bandingkan indeks ke-6 (78) → tidak sama
- Bandingkan indeks ke-5 (83) → tidak sama
- Bandingkan indeks ke-4 (85) → tidak sama
- Bandingkan indeks ke-3 (88) → sama! Nilai ditemukan

Hasil:

Nilai 88 ditemukan pada indeks ke-3.