



MODUL PRAKTIKUM

Algoritma dan Pemrograman 2

Searching

ILMU KOMPUTER FPMIPA UPI
TIM ASISTEN PEMROGRAMAN
ANGKATAN 14 (2022)

PENDAHULUAN

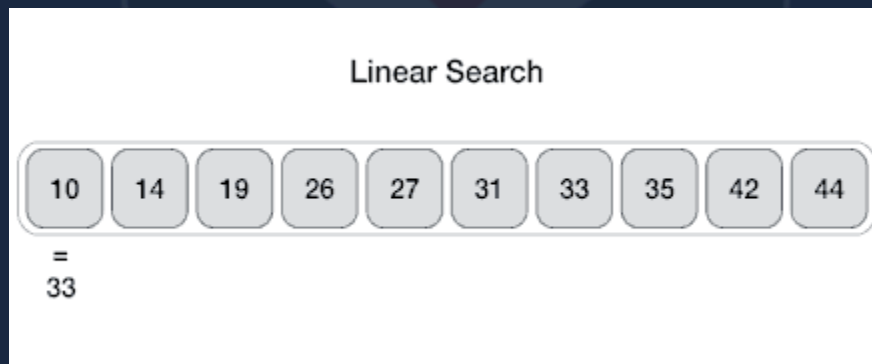
Searching yang jika diartikan ke dalam bahasa Indonesia memiliki arti “mencari”. Dalam konteks pemrograman, searching bermakna proses melakukan pencari suatu nilai dari kumpulan data.

JENIS-JENIS SEARCHING

Pada dasarnya, terdapat **banyak jenis algoritma searching yang ada**. Akan tetapi kita akan mengenali 4 jenis searching dan untuk saat ini cukup mempelajari salah dari 2 jenis searching.

1. **Linear atau Sequential Search.**

Merupakan algoritma yang paling sederhana. Cara kerjanya adalah dengan cara memeriksa apakah setiap elemen merupakan target atau nilai yang ingin dicari dari kumpulan data. Setiap data dicecek secara berurutan dari awal sampai akhir untuk mencari target apakah ditemukan atau tidak. **Apakah algoritma ini pasti menemukan data yang dicari (jika ada)?** Yap, pasti ketemu. **Apakah efisien?** Tidak jika untuk data yang sangat banyak karena pengecekan dilakukan satu persatu.



Visualisasi Sequential Search

2. **Binary Search**

Binary Search merupakan algoritma searching yang efisien untuk **kumpulan data yang sudah terurut** sebelumnya. Cara kerja dari algoritma ini dengan membagi data menjadi dua bagian dan membandingkan elemen yang berada di tengah apakah merupakan target data yang kita cari. Prinsip dari algoritma ini adalah terus membagi sampai pengecekan berada di data terakhir. Dibandingkan dengan Sequential Search, algoritma Binary Search jauh lebih efisien.

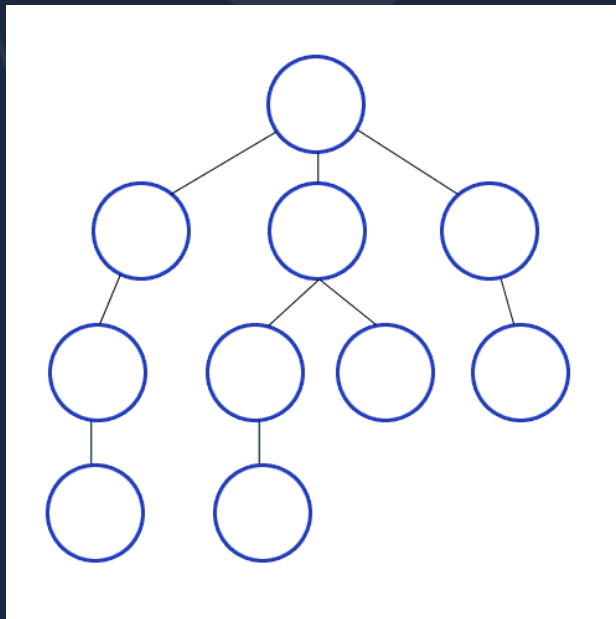
Search for 47

0	4	7	10	14	23	45	47	53
---	---	---	----	----	----	----	----	----

Visualisasi Binary Search

3. DFS (Depth First Search)

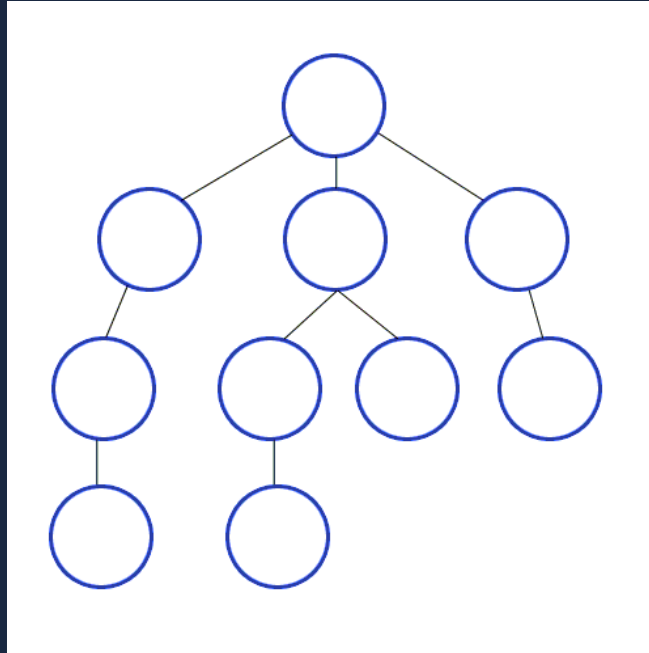
Algoritma searching yang digunakan dalam struktur data seperti graph. Cara kerjanya adalah dengan melakukan pencarian dengan jalur sejauh atau sedalam mungkin sampai node terakhir, lalu baru kembali untuk melakukan pencarian pada jalur cabang lain.



Visualisasi DFS

4. BFS (Breadth-First-Search)

Cara kerja BFS dengan menelusuri setiap level dari node untuk melakukan pencarian data yang dibutuhkan.



Visualisasi BFS

KODE PROGRAM SEQUENTIAL SEARCH

```
1  #include <stdio.h>
2
3  // inisialisasi nilai
4  int n = 10;
5  int arr[] = {4, 8, 0, 2, 6, 2, 8, 9, 10, 15};
6  int targetValue = 9; // angka yang ingin dicari
7
8  // Utility: Mencari sebuah angka pada sebuah array
9  // Returns value: 0 berarti data yang dicari tidak ada, 1 berarti data yang dicari ada
10
11 int sequentialSearch(int n, int arrOfInt[], int target)
12 {
13     int found = 0; // penanda apakah angka yang dicari sudah ditemukan
14     int i = 0;
15
16     while (i < n && found == 0) // proses pencarian sampai ketemu, jika ketemu berhenti loop data
17     {
18         // cek apakah elemen dengan index ke i pada array arrOfInt adalah elemen yang dicari
19         if (arrOfInt[i] == target)
20         {
21             found = 1; // penanda ketemu
22         }
23         else // jika elemen dengan index ke i pada array arrOfInt bukan merupakan elemen yang dicari
24         {
25             i++; // lanjut ke data berikutnya
26         }
27     }
28
29     return found;
30 }
31
32 int main()
33 {
34     // panggil fungsi sequential search dan cek apakah data yang dicari ketemu?
35     if (sequentialSearch(n, arr, targetValue) == 0) // jika ketemu
36     {
37         printf("Data tidak ditemukan!\n");
38     }
39     else // jika tidak
40     {
41         printf("Data ditemukan!\n");
42     }
43 }
```

KODE PROGRAM BINARY SEARCH

```
1  #include <stdio.h>
2
3  // inisialisasi nilai
4  int n = 15;
5  int arr[] = {4, 8, 0, 2, 6, 2, 8, 1, 13, 22, 4, 9, 10, 15, 3};
6  int targetValue = 9; // angka yang ingin dicari
7
8  // Utility: Mencari sebuah angka pada sebuah array
9  // Returns value: 0 berarti data yang dicari tidak ada, 1 berarti data yang dicari ada
10
11 int binarySearch(int n, int arrOfInt[], int targetValue)
12 {
13     int i = 0, j = n - 1, mid; // penanda rentang data, dan data tengah yang akan dicocokkan dengan target
14     int found = 0; // penanda hasil pencarian
15
16     // jika found masih bernilai 0 dan i kurang dari sama dengan j, lakukan:
17     while (found == 0 && i <= j)
18     {
19         // mengambil data tengah dari pembagian rentang data
20         mid = (i + j) / 2;
21
22         // cek data tengah pada rentang data i ke j apakah sama dengan data yang ingin dicari?
23         if (arrOfInt [mid] == targetValue)
24         {
25             // jika ya maka ubah nilai found jadi 1
26             found = 1;
27         }
28         // jika tidak, cek apakah nilai data tengah lebih besar dari data yang dicari?
29         else
30         {
31             if (arrOfInt [mid] >= targetValue)
32             {
33                 // jika ya maka penanda rentang data kanan (j)
34                 // dipindahkan ke posisi data tengah lalu sebelah kirinya
35                 j = mid - 1;
36             }
37             else
38             {
39                 // jika tidak maka penanda rentang data kiri (i)
40                 // dipindahkan ke posisi data tengah lalu sebelah kanannya
41                 i = mid + 1;
42             }
43         }
44     }
45
46     return found;
47 }
48
49 int main()
50 {
51     // panggil fungsi binary search dan cek apakah data yang dicari ketemu?
52     if (binarySearch(n, arr, targetValue) == 0) // jika ketemu
53     {
54         printf("Data tidak ditemukan!\n");
55     }
56     else // jika tidak
57     {
58         printf("Data ditemukan!\n");
59     }
60 }
```

LATIHAN

BESOK YAH :D



PENUTUP

Penulis mengucapkan terimakasih yang sebesar-besarnya kepada semua elemen seperti teman-teman penulis, penulis, dan juga dosen penulis yang telah mendukung berjalannya sesi praktikum pada mata kuliah Algoritma dan Pemrograman 2 ini. Semoga apa yang telah kita laksanakan dapat menjadi berkah dan memberi manfaat bagi kita semua pada kemudian hari.

DAFTAR PUSTAKA

Tim Asisten Praktikum Algoritma dan Pemrograman 2 Angkatan 13 (2021).

Searching. Modul Praktikum Algoritma dan Pemrograman 2. Bandung, Universitas Pendidikan Indonesia.

Aninisa (2023).

Algoritma Search. Fakultas Ilmu Komputer dan Teknologi Informasi. Medan, Universitas Muhammadiyah Sumatera Utara.