Rangkuman

INTRODUCTION TO ALGORITHMS

Algoritma adalah sekumpulan instruksi untuk menyelesaikan suatu tugas.

Binary Search adalah suatu algoritma dengan input berupa list yang terdiri dari sejumlah elemen yang terurut. Binary search dapat membantu dalam pencarian menjadi lebih cepat. Ketika kata yang dicari terdapat dalam list yang terurut, maka binary reaserch akan mengembalikan posisi kata tersebut. Jika tidak ditemukan, maka binary reaserch akan mengembalikan nilai kosong (null).

Contohnya adalah ketika sedang mencari sebuah kata dalam kamus, misalnya kata want(ingin) yang berawalan dengan huruf w. Di dalam kamus, setiap kata telah diurutkan sesuai abjad. Binary search akan mencari kata tersebut dalam kamus dan ketika telah ditemukan, maka akan mengembalikan posisi kata terebut. Jika tidak di temukan maka akan mengembalikan nilai null (kosong).

Simple search adalah algoritma pencarian yang mencari elemen dengan memeriksa elemen list satu-persatu. Ini akan membuat pencarian lebih lama. Misalnya saya harus menebak sebuah angka dari rentang 1 sampai 50, Simple Search akan mengeliminasi angka secara berurutan satu persatu sampai menemukan angka yang benar. Sedangkan dengan Binary Search, pencarian angka dapat dilakukan dengan menebak sebuah angka terlebih dahulu dan akan mendapatkan informasi apabila angka tebakan terlalu tinggi atau terlalu rendah, sehingga kita bisa mengeliminasi atau memotong angka yang sudah pasti tidak termasuk dalam terlalu tinggi atau terlalu rendah tersebut.

**Logaritma**

Logaritma adalah kebalikan dari perpangkatan.

10³ = 1000 dalam bentuk logaritma adalah log10 1000 = 3

Ketika mencari sebuah elemen dengan simple search, maka kemungkinan terburuk kita harus mencari di setiap elemen sampai mendapatkan elemen yang dicari. Sedangkan, binary search, kita harus mencari log n elemen dalam kemungkinan terburuk.

**Binary Search in Phyton**

Kode simpel yang digunakan adalah list. List digunakan untuk menyimpan sekumpulan nilai dalam sebuah list. Posisi pertama dalam list memiliki index 0, posisi kedua index 1, dst.

Fungsi Binary search memiliki sebuah list terurut dan sebuah item. Jika item terdapat dalam list, maka fungsi akan mengembalikan posisi item. Caranya yaitu dengan membagi list menjadi 2 bagian dan mengecek data di urutan yang dipilih apakah sama atau tidak dengan data yang dicari. Kemudian mengecek masing-masing data yang sudah dibagi 2 bagian tadi apakah terdapat data yang dicari. Ulang cara tersebut hingga data yang dicari ditemukan. Dengan Binary Search, jika nilai atau data yang dicari ada pada urutan terakhir, tidak harus memeriksa keseluruhan data.

**Running Time**

Secara umum, kita selalu memilih untuk menggunakan algoritma yang efisien agar mengoptimalkan waktu dan ruang. Dengan Binary Search, jika list berisi 100 item, butuh paling buruk 7 kali pengecekan. Jika sebuah list berisi 4 miliar item, maka paling buruk ada 32 kali pengecekan. Binary Search menerapkan logarithmic time (log time).

A white paper with text and words

Description automatically generated with medium confidence

**Big O Notation**

Big O Notation adalah notasi khusus yang bisa digunakan untuk mengetahui seberapa cepat suatu algoritma bekerja. Perbedaan waktu bekerja antara Simple Search dan Binary Search :

A diagram of a simple search

Description automatically generated

Jika Simple Search membutuhkan 100 miliseconds untuk 100 elemen, Binary Searh membutuhkan 7 miliseconds untuk 100 elemen. Yang harus menjadi perhatian, kedua algoritma tersebut tidak berkembang dengan kecepatan yang sama. Untuk 1.000.000.000 elemen, Simple Search membutuhkan 1 miliar miliseconds atau 11 hari sedangkan Binary Search hanya membutuhkan 32 miliseconds untuk bekerja. Jadi, seiring dengan berkembangannya elemen list menjadi lebih besar, Binary Search bekerja lebih cepat dibandingkan Simple Search.

Big O Notation memberitahu seberapa cepat sebuah algoritma. Misalnya, terdapat sebuah list dengan size n. Jika Simpel Search melakukan pengecekan elemen satu-persatu, maka akan membutuhkan n operasi. Run time dalam Big O Notation-nya adalah O(n). Big O Notation tidak memberitahu kecepatan algoritma dalam detik, melainkan membuat pengguna membandingkan jumlah operasi. Dimana Big O Notation memberitahu seberapa cepat algoritma berkembang.

A white text box with black text

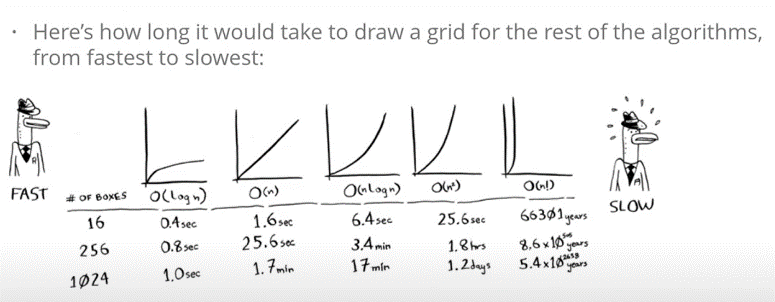
Description automatically generated

Big O Notation membutuhkan log n operasi untuk mengecekan list dengan ukuran n. Maka running time Binary Searh diekspresikan sebagai O(log n). Secara umum, Big O Notation dituliskan sebagai berikut:

A black text and arrows

Description automatically generated with medium confidence

Nilai log yang digunakan pada Binary Search adalah log basis 2. Misalnya, jika terdapat list dengan ukuran 1.024, maka log 1.024 = 10 atau ditbutuhkan waktu 1 detik.



Sumbu x merepresentasikan number of element.

Sumbu y merepresentasikan angka notasinya.