**Belajar Algoritma Dari Nol Part 1**

Algoritma adalah kumpulan himpunan struktur untuk menyelesaikan tugas. Algoritma dipilih karena dapat bekerja cepat dan dapat memecahkan masalah atau keduanya.

1. **Binary Search**

* Misalkan kita diminta mencari huruf di sebuah kitas, dan kata itu diawali dengan K.
* Kita dapat memulai membalik kertas kitas dimulai dari halaman awal hingga kita mendapatkan kumpulan abjad K.
* Tapi akan lebih mudah menemukan kumpulan abjad K, ketika kita memulai membuka halaman kitas dari tengah, karena kita tahu abjad K akan berada di pertengahan kitas.
* Pencarian abjad kitas merupakan permasalahan pencarian, solusi dari permasalahan ini adalah dengan Binary Search, dimana inputan dari algoritma ini berisikan kumpulan elemen yang terurut dari sebuah list.
* Jika elemen yang kita cari berada di dalam list, Binary Search akan mengembalikan dimana elemen berada, dilain kemungkinan jika tidak menemukan elemen maka Binary Search akan mengembalikan ke nilai null.

1. **Case Study**

Berpikirlah untuk menebak sebuah angka dari rentang 1 hingga 100, kita harus mencoba menebak angka yang diberikan se-sedikit mungkin. Setiap tebakan, akan diberikan clue, tebakan akan diberi clue, *too low,* *too high,* dan *correct.*

**Solution:**

**Simple Search Way**

* Salah satu solusi yang dapat kita lakukan adalah dengan menebak satu persatu angka dari rentang 1 hingga 100 : 1, 2, 3, 4, 5,...., 100
* Simple search dapat dikatakan Stupid search jika dalam versi lebih baiknya.
* Dengan setiap tebakan, kita hanya dapat mengeliminasi 1 angka saja.
* Jika tebakan yang benar adalah angka 99, maka kita harus menebak 99 kemungkinan untuk tebakan yang benar.

**Binary Search**

* Ini adalah solusi yang lebih baik
* Dimulai dari 50
* Too low! berarti kita telah mengeliminasi sebagian dari deretan angka (51, 52, 53, ...., 100)
* Dan sekarang kita tahu jika dari rentang 1 hingga 50 adalah too low. Maka tebakan kedua adalah 75.
* Too high! Tetapi kita telah mengeliminasi setengah dari angka yang tersisa.
* **Dengan binary search, kita dapat menebak angka mulai dari tengah dan mengeliminasi setengah dari angka setiap tebakan.**
* Selanjutnya 63, (karena merupakan nilai tengah antara 50 dan 75)
* Too high! Dan terakhir kita menebak **57** **tebakan benar.**

1. **Binary Search Performance Explained**

* Ini Binary Search. Kita baru saja mempelajari algoritma pertamamu!
* Ini adalah beberapa angka yang kita dapat selalu eliminasi
* 100 item-> 50 -> 25 -> 13 -> 7 -> 4 -> 2 -> 1 (7 steps)
* Eliminasi setengah angka dengan Binary Search
* Kita dapat menebak maksimum 7 tebakan karena kita telah mengeliminasi banyak angka dengan setiap tebakan.
* Misalkan kita mencari sebuah kata di dalam kamus. Dimana kamus itu terdiri 240.000 kata.
* Di kasus terburuknya, berapa kira-kira banyak steps yang diambil?
* Simple search: 240.000 steps
* Binary search: 240K-> 12OK -> 60K -> 30K -> 15K -> 7.5K -> 375K -> 1875 -> 938 -> 464 -> 235 -> 118 -> 59 -> 30 -> 15 -> 8 -> 4 -> 2 -> 1
* Jadi binary search akan mengambil 18 steps, perbedaan yang sangat besar
* **Umumnya, untuk sebuah list of *n*, binary search akan mengambil log2n step untuk menjalankan case terburuk, sedangkan simple search akan mengambil n steps.**

1. **Logarithms**

* Logs adalah kebalikan dari perpangkatan

|  |  |  |
| --- | --- | --- |
| 10­2 = 100 | <-> | Log10100 = 2 |
| 103 = 1000 | <-> | Log101000 = 3 |
| 23 = 8 | <-> | Log28 = 3 |
| 24 = 16 | <-> | Log216 = 4 |
| 25 = 32 | <-> | Log232 = 5 |

* Di buku, ketika berbicara running time in Big O notasi (dijelaskan sedikit kemudian), log selalu berarti log2.
* Ketika kita mencari untuk sebuah elemen menggunakan simple search, di case terburuk mengharuskan kita untuk melihat setiap elemen.
* Jadi untuk list angka 8, maka kita harus memeriksa paling banyak 8 angka.
* Untuk binary search, kita harus mengecek log n elemen di case terburuknya.
* Untuk list dari 8 elemen, log 8 = 3, karena 23 = 8. Jadi untuk list angka 8, kita harus mengecek paling banyak 3 angka.
* Untuk sebuah list dari 1024 elemen, log 1024 = 10, karena 210 == 1024. Jadi list untuk 1024 angka, setidaknya kita harus memeriksa paling banyak 10 angka.