



# Perkenalan Pemrograman Kompetitif Dasar

Tim Olimpiade Komputer Indonesia

# Perkenalan

Selamat datang di topik **Pemrograman Kompetitif Dasar!**

Anda diharapkan telah menguasai materi pemrograman dasar, dan:

- Mengetahui setidaknya satu bahasa pemrograman.
- Mampu membaca dan menulis program.

Mulai dari bab ini, seluruh kode program akan dituliskan dalam **pseudocode**.



# Pseudocode

- Merupakan **bahasa informal** yang mirip dengan bahasa pemrograman untuk mendeskripsikan program.
- Biasa digunakan pada materi pembelajaran algoritma.
- *Pseudocode* sendiri bukanlah bahasa pemrograman sungguhan.



## Contoh Pseudocode

INSERTIONSORT( $A$ )

```
1  for  $i = 2$  to  $A.length$ 
2       $j = i$ 
3      while ( $j > 1$ ) and ( $A[j] < A[j - 1]$ )
4          SWAP( $A[j], A[j - 1]$ )
5           $j = j - 1$ 
```



## Pseudocode (lanj.)

- Memahami *pseudocode* pada awalnya mungkin sulit.
- Seiring berjalannya waktu, Anda akan terbiasa memahami dan menggunakan *pseudocode* dengan mudah.



# Tentang Pemrograman Kompetitif

## Pemrograman Kompetitif

Adalah penyelesaian soal yang terdefinisi dengan jelas dengan menulis program komputer dalam batasan-batasan tertentu (memori dan waktu).



# Tentang Pemrograman Kompetitif (lanj.)

Peserta ditantang untuk:

- Menganalisis permasalahan
- Merancang algoritma solusi
- Menuliskannya dalam bentuk program



## Tentang Pemrograman Kompetitif (lanj.)

- Masalah yang diberikan selalu terdefinisi dengan jelas (*well-defined*).
- Contohnya, batasan masalah dan asumsi yang diperlukan diberikan dengan akurat dan tidak ambigu.
- Solusi ditulis dalam bentuk program program dan memenuhi batas-batas yang ditentukan.
- Batas yang ditentukan: waktu, memori, dan lainnya.



## Contoh Soal Pemrograman Kompetitif

- Terdapat  $N$  ruangan yang dinomori dari 1 hingga  $N$ .
- Ruangan ke- $i$  memiliki sebuah lampu dan sebuah tombol.
- Bila tombol itu ditekan, keadaan lampu pada seluruh ruangan ke- $x$  akan berubah (dari mati menjadi menyala, atau sebaliknya), yang mana  $x$  habis dibagi  $i$ .
- Contoh, bila  $N = 10$  dan tombol di ruangan ke-2 ditekan, maka keadaan lampu pada ruangan 2, 4, 6, 8, dan 10 akan berubah.
- Bila pada awalnya seluruh lampu berada pada keadaan mati, dan tombol pada setiap ruangan ditekan tepat sekali, bagaimanakah keadaan lampu pada ruangan ke- $N$ ?



## Contoh Soal Pemrograman Kompetitif (lanj.)

- Batas waktu: 1 detik.
- Batas memori: 32 MB.
- Nilai  $N$  dijamin selalu memenuhi  $1 \leq N \leq 10^{14}$ .



## Solusi Naif

Salah satu cara penyelesaiannya adalah dengan mensimulasikan skenario pada deskripsi soal:

- Kita cukup memperhatikan lampu pada ruangan ke- $N$  saja.
- Mulai dari ruangan ke-1, dipastikan keadaan lampu pada ruangan ke- $N$  akan berubah ( $N$  habis dibagi 1).
- Lanjut ke ruangan ke-2, periksa apakah 2 habis membagi  $N$ . Bila ya, ubah keadaan lampu ke- $N$ .
- Lanjut ke ruangan ke-3, periksa apakah 3 habis membagi  $N$ . Bila ya, ubah keadaan lampu ke- $N$ .
- ... dan seterusnya hingga ruangan ke- $N$ .



## Solusi Naif (lanj.)

- Setelah selesai disimulasikan, periksa keadaan lampu ruangan ke- $N$  dan cetak jawabannya.
- Kompleksitas solusi ini adalah  $O(N)$ , dan hanya akan bekerja untuk nilai  $N$  yang kecil.
- Untuk  $N$  yang lebih besar, misalnya  $N = 10^9$ , kemungkinan besar diperlukan waktu lebih dari 1 detik.
- Solusi ini tidak akan mendapatkan nilai penuh, atau bahkan 0, tergantung skema penilaian yang digunakan.



## Solusi yang Lebih Baik

- Dengan sedikit observasi, yang sebenarnya perlu dilakukan adalah menghitung banyaknya pembagi dari  $N$ .
- Apabila banyaknya pembagi ganjil, berarti pada akhirnya lampu di ruangan ke- $N$  akan menyala (mengapa?).
- Bila genap, berarti lampu di ruangan ke- $N$  akan mati.



## Solusi yang Lebih Baik (lanj.)

- Untuk menghitung banyaknya pembagi dari  $N$  dengan efisien, lakukan faktorisasi prima terlebih dahulu.
- Misalkan  $N = 12$ , maka faktorisasi primanya adalah  $2^2 \times 3$ .
- Untuk menjadi pembagi dari 12, suatu bilangan hanya boleh:
  - Memiliki faktor 2 maksimal sebanyak 2.
  - Memiliki faktor 3 maksimal sebanyak 1.
  - Tidak boleh memiliki faktor lainnya.



## Solusi yang Lebih Baik (lanj.)

Sebagai contoh, berikut daftar seluruh pembagi dari 12:

- $1 = 2^0 \times 3^0$
- $2 = 2^1 \times 3^0$
- $3 = 2^0 \times 3^1$
- $4 = 2^2 \times 3^0$
- $6 = 2^1 \times 3^1$
- $12 = 2^2 \times 3^1$



## Solusi yang Lebih Baik (lanj.)

- Banyaknya pembagi dari 12 sebenarnya sama saja dengan banyaknya kombinasi yang bisa dipilih dari  $\{2^0, 2^1, 2^2\}$  dan  $\{3^0, 3^1\}$ .
- Banyaknya kombinasi sama dengan mengalikan banyaknya elemen pada tiap-tiap himpunan.
- Sehingga, banyaknya cara adalah  $3 \times 2 = 6$  cara.
- Dengan demikian, banyaknya pembagi dari 12 adalah 6.



## Solusi yang Lebih Baik (lanj.)

- Cara ini juga berlaku untuk nilai  $N$  yang lain.
- Misalnya  $N = 172.872 = 2^3 \times 3^2 \times 7^4$ .
- Berarti banyak pembagiannya adalah  $4 \times 3 \times 5 = 60$ .



## Solusi yang Lebih Baik (lanj.)

- Secara umum, banyaknya pembagi dari:

$$N = a_1^{p_1} \times a_2^{p_2} \times a_3^{p_3} \times \dots \times a_k^{p_k}$$

adalah:

$$(1 + p_1) \times (1 + p_2) \times (1 + p_3) \times \dots \times (1 + p_k)$$

- Sehingga, cukup faktorkan  $N$ , lalu hitung banyak pembaginya dengan rumus tersebut.
- Faktorisasi bilangan dapat diimplementasikan dalam  $O(\sqrt{N})$ .
- 1 detik cukup untuk  $N$  sampai  $10^{14}$ .



# Ajang Pemrograman Kompetitif

- Olimpiade Sains Nasional (**OSN**) bidang komputer/informatika merupakan kompetisi tingkat nasional di Indonesia.
- International Olympiad in Informatics (**IOI**) merupakan kompetisi tingkat internasional bagi siswa SMA dari seluruh dunia.
- Untuk mahasiswa, terdapat ACM International Collegiate Programming Contest (**ACM-ICPC**) yang pesertanya terdiri dari tim-tim beranggotakan tiga orang.



# Manfaat Pemrograman Kompetitif

- Mengasah kemampuan memecahkan masalah.
- Bertemu dengan teman-teman sehoobi.
- Biasanya, wawancara untuk masuk ke perusahaan teknologi terkemuka menggunakan soal-soal pemrograman kompetitif.



Dan Tentunya...

Menantang dan menyenangkan!



# Penutup

- Sebagai pemanasan, silakan mengerjakan soal-soal latihan yang diberikan.

