

Desain dan Analisis Algoritma

Semester Genap 2023/2024

PR 1 - Analyzing Algorithm and Growth of Function

Deadline: Senin, 19 Februari 2024

Petunjuk Pengerjaan:

- Berkas PR Anda harus dibuat dengan cara ditulis tangan di kertas A4 lalu difoto/di scan dan disimpan sebagai satu berkas PDF (bukan di-*zip*). Mengumpulkan selain tipe file PDF dikenakan penalti sebesar 5 poin.
- Format penamaan berkas PR1 NPM Nama.pdf.

Contoh: PR1 2106123456 John Doe.pdf.

Penamaan berkas yang tidak sesuai dikenakan penalti sebesar 5 poin.

- Tuliskan Nama dan NPM Anda pada bagian kiri atas setiap halaman pada PR Anda.
- Awali berkas PR Anda dengan pernyataan “Dengan ini saya menyatakan bahwa PR ini adalah hasil pekerjaan saya sendiri” disertai tanda tangan pada halaman pertama berkas PR anda. **Tanpa pernyataan ini, PR Anda tidak akan diperiksa.**
- Jika ada, tuliskan Nama Kolaborator pada berkas PR Anda. Perhatikan bahwa walaupun Anda sudah menuliskan nama kolaborator, bukan berarti jawaban Anda boleh sama persis dengan kolab- orator Anda. PR adalah tugas individu, bukan tugas kelompok. Pastikan kolaborasi hanya pada sebatas ide pengerjaan, bukan ketika menulis jawaban.
- Anda harus memberikan penjelasan jawaban pada setiap soal. Bila kurang penjelasan, maka akan dikenakan penalti.
- Anda diperbolehkan menghitung menggunakan kalkulator, namun langkah pengerjaan harus dije- laskan. Tidak boleh menulis nilai akhir saja.
- Pelanggaran peraturan kejujuran akademis akan diproses sesuai peraturan yang sudah dijelaskan di BRP.
- Keterlambatan mengumpulkan PR akan dikenai penalti sebesar 30% dari nilai total selama dikumpulkan **maksimal satu jam setelah batas pengumpulan PR.**

1. Soal 1 (4 + 2 + 12 poin)

Diberikan pseudocode sebuah algoritma yang menerima input bilangan asli n sebagai berikut:

```
MISTERI(n):  
1  if  $n \leq 1$ :  
2    return False  
3  for  $i$  from 2 to  $\lfloor \sqrt{n} \rfloor$ :  
4    if  $n \bmod i == 0$ :  
5      return False  
6  return True
```

- a) Hitunglah keluaran dari MISTERI(15) dan MISTERI(17), kemudian hitung juga berapa kali conditional test di baris 4 dieksekusi!
- b) Apa yang dilakukan oleh fungsi MISTERI tersebut?
- c) Setelah Anda mengetahui bahwa fungsi MISTERI mengerjakan permasalahan X, sekarang coba buktikan kebenaran bahwa fungsi MISTERI memang menyelesaikan permasalahan X. Gunakan loop invariant untuk melakukan pembuktian ini. Jelaskan loop invariant yang Anda pilih, dan tunjukkan bahwa invariant tersebut terpenuhi pada kondisi initialization, maintenance, dan termination!

2. Soal 2 (12 poin)

Diberikan pseudocode fungsi SUM-ARRAY sebagai berikut:

```
SUM-ARRAY(A, n):  
1  sum = 0  
2  for  $i = 1$  to  $n$ :  
3    sum = sum +  $A[i]$   
4  return sum
```

Fungsi SUM-ARRAY menjumlahkan n buah angka pada array $A[1:n]$. Nyatakan loop invariant untuk fungsi SUM-ARRAY, dan gunakan 3 aspek dari loop invariant: initialization, maintenance, dan termination untuk menunjukkan bahwa SUM-ARRAY memang menjumlahkan angka-angka pada $A[1:n]$.

*Untuk soal 3 dan 4, asumsikan model komputasi yang digunakan adalah Random Access Machine, dengan running time untuk assignment = bernilai c_1 , penjumlahan + bernilai c_2 , perkalian * bernilai c_3 , dan perbandingan == bernilai c_4 . Intruksi di luar keempat operasi di atas tidak perlu diikutsertakan dalam perhitungan.*

3. Soal 3 (3 + 7 + 2 + 8 poin)

Diberikan pseudocode sebuah algoritma yang menerima input bilangan asli n sebagai berikut:

MISTERI2(n):

```
1  x = 0
2  for i = 1 to n:
3      y = 1
4      for j = 1 to i:
5          y = y * j
6          x = x + y
7  return x
```

- a) Apa yang dilakukan algoritma tersebut? Anda dapat memberikan jawaban berupa persamaan atau kalimat deskripsi. Berikan keluaran jika $n = 5$ untuk mendukung jawaban Anda.
- b) Hitunglah running time total $T(n)$ algoritma tersebut dengan menghitung jumlah dieksekusinya tiap baris dalam pseudocode.
- c) Berdasarkan running time yang didapat, tentukan kompleksitas waktu algoritma tersebut dalam notasi big Θ .
- d) Modifikasi algoritma tersebut sehingga mampu menyelesaikan masalah yang sama, namun dengan kompleksitas waktu yang lebih rendah. Berapa kompleksitas waktu dalam notasi big Θ yang Anda dapatkan? Lakukan langkah b dan c untuk mendapatkan jawaban Anda.

4. Soal 4 (4 + 4 + 7 poin)

Diberikan pseudocode algoritma linear search yang berfungsi mencari elemen x dalam array $A[1..n]$ sebagai berikut:

LINEAR-SEARCH(A, x):

```
1  for i = 1 to n
2      if A[i] == x:
3          return i
4  return -1
```

- a) Secara umum, input seperti apa yang merupakan worst case? Hitunglah running time total $T(n)$ pada kondisi tersebut.
- b) Secara umum, input seperti apa yang merupakan best case? Hitunglah running time total $T(n)$ pada kondisi tersebut.

- c) Hitunglah running time total $T(n)$ pada kondisi average case, dengan asumsi x memiliki kemungkinan kemunculan yang sama untuk setiap posisi di A .

5. Soal 5 ($7 + 5 + 5 + 5 + 6 + 7 = 35$)

- a. Jawablah pertanyaan ini sesuai dengan NPM Anda:

- **Untuk NPM Genap**

Tentukanlah mana fungsi yang secara asimptotik lebih besar diantara $\lfloor \lg(n) \rfloor!$ atau $\lg(n!)$.

- **Untuk NPM Ganjil**

Tentukanlah mana fungsi yang secara asimptotik lebih besar diantara $\lg(\lg^* n)$ atau $\lg^*(\lg n)$

- b. Buktikan bila benar atau berikan *counterexample* jika salah untuk pernyataan:

“Jika diketahui bahwa $f(n) = O(g(n))$ dan $f(n) = \Theta(h(n))$, maka akan berlaku bahwa $g(n) = \Omega(h(n))$ ”

- c. Buktikan bila benar atau berikan *counterexample* jika salah untuk pernyataan:

“Jika diketahui $f(n) = O(g(n))$ dan $\lg(g(n)) \geq 0$ serta $f(n) \geq 1$ untuk semua nilai n yang cukup besar, maka akan berlaku bahwa $\lg(f(n)) = O(\lg(g(n)))$ ”

- d. Tunjukkan bahwa jika $k^3 \lg(k) = \Theta(n^3)$, maka berlaku bahwa $k^3 = \Theta\left(\frac{n^3}{\lg(n)}\right)$.

- e. Misalkan terdapat suatu fungsi:

$$p(n) = \sum_{i=0}^d a_i n^i$$

yang mana $a_d > 0$, dengan $p(n)$ merupakan suatu polinomial berderajat d , dengan suatu konstan k . Gunakan definisi formal dari notasi asimptotik untuk membuktikan bahwa:

“Jika $k = d$, maka akan berlaku bahwa $p(n) = \Theta(n^k)$ ”

- f. Buktikan secara formal bahwa berlaku $4^n = \omega(3^n)$. Berikan observasi dan justifikasi yang tepat bahwa ketidaksamaan berlaku untuk semua pemilihan nilai c terhadap n_o yang dipilih!