Nama: Alvaro Austin

NPM: 2106352180

Kelas: A

Kode Asdos: 3

Dengan ini saya menyatakan bahwa PR ini adalah hasil pekerjaan saya sendiri

1. a) Hasil keluaran dari misteri (3, [2, 10, 6, 5, 9]) dan misteri (7, [2, 10, 6, 5, 9])

kita dapat membuat kode melalui python:

```
def misteri (m, A):
    x = 0
    i = 0
    n = len(A)
    while (i<n):
        temp = abs (A[i] - m)
        if (temp > x):
            x = temp
        i = i+1
    return x
```

hasil dari misteri (3, [2, 10, 6, 5, 9]):

x_0 = x sebelum loop saat itu.

n = 5 → i = 0 → x = 1 temp = 1 , x_0 = 0

i = 1 → x = 7 temp = 7 x_0 = 1

i = 2 → x = 7 temp = 3 x_0 = 7

i = 3 → x = 7 temp = 2 x_0 = 7

i = 4 → x = 7 temp = 6 x_0 = 7

Nilai x akhir adalah 7, sehingga keluarannya 7

hasil dari misteri (7, [2, 10, 6, 5, 9]):

n = 5 → i = 0 → x = 5 temp = 5 x_0 = 0

x = 5 temp = 3 x_0 = 5

x = 5 temp = 1 x_0 = 5

x = 5 temp = 2 x_0 = 5

x = 5 temp = 4 x_0 = 5

Oleh karena itu hasil akhir yg dikeluarkan adalah 5

b) Operasi yg dilakukan algoritma misteri (m, A) adalah operasi untuk mencari nilai maksimum mutlak dari selisih antara elemen Array A dan nilai m. mengiterasi semua elemen dalam array A lalu menghitung selisih mutlak antara elemen ini dan nilai m dan diambil nilai maksimumnya. Sehingga operasi ini

Keyword: Find Absolute maximum difference dalam A[1 ... n] dengan M

c) Loop invariant: nilai x merupakan hasil selisih mutlak terbesar dengan nilai m pada elemen A [1 ... i-1] atau bernilai 0 sebelum iterasi.

↳ Initialization: Sebelum loop dimulai, dapat dilihat bahwa nilai i = 1 dan x = 0, hal ini tentu saja benar karena memenuhi loop invariant karena x bernilai 0. Hal ini juga dapat dibuktikan karena belum ada elemen pada A yg dapat dibandingkan dengan M.

↳ Maintenance: Jika diasumsikan loop invarian benar untuk A [1, ... i-1] dengan x merupakan selisih mutlak terbesar elemen² A dengan nilai m. Kita dapat melakukan observasi bahwa loop melakukan langkah ini:

$temp = \max_{1 \le j \le i-1} |A_j - m|$

- Jika $|A_i - m| \le$ temp maka nilai dari temp (maximum absolute difference yg disimpan) tidak berubah.
- Jika $|A_i - m| >$ temp maka nilai temp dilakukan overwrite menjadi $|A_i - m|$. temp = $|A_i - m|$

Oleh karena itu, pada dua kasus tersebut berlaku bahwa temp = maximum absolute difference yg disimpan $= \max_{1 < j < i-1} |A_j - m|$ berlaku dan menjadi prasyarat untuk loop berikutnya.

Maka, loop invarian juga benar sebelum iterasi sebelumnya.

↳ Termination: Setelah loop berhenti dan berhasil dijalankan (nilai i lebih besar n). Hal ini berarti bahwa i = n+1, maka kita memperoleh bahwa nilai x merupakan hasil selisih mutlak terbesar dengan nilai m pada elemen array A [1 ... n] dan A [1 ... n] merupakan keseluruhan array keseluruhan. Sehingga x memenuhi tujuan dari algoritma diatas yaitu mencari hasil selisih mutlak terbesar antara m dengan elemen pada A. Oleh karena itu, algoritma diatas benar.

Sehingga loop invariant diatas benar dan berlaku pada algoritma diatas.

Nama: Alvaro Austin
Kelas: A
NPM: 2106752180
Kode Asdos: 3

② Kita dapat menggunakan definisi limit untuk $O(g(n)) = f(n)$ dimana

$$\lim_{n\to\infty}\left(\frac{f(n)}{g(n)}\right) < \infty \implies f(n) = O(g(n))$$

Terdapat order of growth dasar yg dapat kita ketahui:

$$1 < \log(n) < \ln(n) < \lg(n) < \sqrt{n} < n < n\lg n < n^2 < 2^n < n!$$

Oleh karena itu kita dapat mengurutkan berdasarkan pengetahuan kita diatas:

$F_1 = 2\lg(n+50)^{20}$
$F_2 = \ln^2(n+10)$
$F_3 = \sqrt[3]{n+1}$
$F_4 = 0.002n^5 + 5n^3 + 3$
$F_5 = 2^{3n}$
$F_6 = (n-3)!$

Pembuktian bahwa:

- $F_1 = O(F_2)$

$$\lim_{n\to\infty}\left(\frac{2\lg(n+50)^{20}}{\ln^2(n+10)}\right) = \lim_{n\to\infty}\left(\frac{40\lg(n+50)}{\ln^2(n+10)}\right) = \lim_{n\to\infty}\frac{40\ln(n+50)}{\ln^2(n+10)\ln(2)}$$

L'hospital

$$= \lim_{n\to\infty}\frac{40(n+50)^{-1}}{2\ln(n+10)(n+10)^{-1}\ln(2)}$$

$$= \lim_{n\to\infty}\frac{40(n+10)}{2\ln(2)((n+50)(n+10)^{-1} + \ln(n+10))}$$

$$= 0 < \infty$$ //

- $F_2 = O(F_3) \to$ Perlu dicari bahwa $\forall n \geq n_0$, $\ln^2(n+10) \leq c\sqrt[3]{n+1}$, cari n_0 dan c //

$$c \geq \frac{\ln^2(n+10)}{\sqrt[3]{n+1}}$$

maka kita dapat menjabarkan bahwa $f(n) = \frac{\ln^2(n+10)}{\sqrt[3]{n+1}}$

dimana $f'(n) = \frac{-\ln(n+10)((n+10)\ln(n+10) - 6n - 6)}{3(n+1)^{\frac{4}{3}}(n+10)}$

Dapat dilihat untuk $n \geq 500$, $\ln(n+10) > 6$

dan $(n+10)\ln(n+10) - 6n - 6 > 6(n+10) - 6n - 6 = 54 > 0$ //

Sehingga untuk $n \geq 500$, dapat dilihat bahwa $f'(n) < 0$ sehingga $\forall n \geq 500$, $f(n) \leq f(500)$.

$\hookrightarrow \ln(n+10), (n+1)^{\frac{1}{3}}, (n+10)$, bernilai positif.

maka dapat diambil $n_0 = 500$ dan $c = f(500)$, terbukti bawan $\ln^2(n+10) = O(\sqrt[3]{n+1})$

- $F_3 = O(F_4)$

$$\lim_{n\to\infty}\frac{\sqrt[3]{n+1}}{0.002n^5 + 5n^3 + 3}$$

= kita dapat mengambil turunan (L'Hospital)

$$= \lim_{n\to\infty}\frac{(n+1)^{\frac{1}{3}}}{0.002n^5 + 5n^3 + 3} = \lim_{n\to\infty}\frac{\frac{1}{3}(n+1)^{-\frac{2}{3}}}{0.01n^4 + 15n^2}$$

$$= \lim_{n\to\infty}\frac{1}{3(n+1)^{\frac{2}{3}}(0.01n^4 + 15n^2)}$$

= oleh karena itu untuk n sangat tinggi menyebabkan hasil mendekati 0

$$= 0 < \infty$$ //

Terbukti: $\sqrt[3]{n+1} = O(0.002n^5 + 5n^3 + 3)$ //

Nama: Alvaro Austin

Kelas: A

Kode Asdos: 3

NPM: 2106752180

$f(n)$ · $g(n)$

- $F_4 = O(F_5)$

$$\lim_{n\to\infty} \frac{0.002n^5 + 5n^3 + 3}{2^{3n}} \to \left(\frac{0.002n^5}{8^n} + \frac{5n^3}{8^n} + \frac{3}{8^n}\right) \to$$ dimana kita dapat menghapus $\frac{5n^3}{8^n} + \frac{3}{8^n}$ karena operasi yg tidak relevan

Sehingga $\lim_{n\to\infty} = \frac{0.002n^5}{8^n}$ = Untuk n sangat tinggi operasi pembagian ini akan menghasilkan nilai yg sangat mendekati 0, Oleh karena itu hasil $= 0$

Maka terbukti karena $\frac{f(n)}{g(n)} < \infty$; $\frac{f(n)}{g(n)} = 0$ maka $0 < \infty$

$F_4 = O(F_5)$ terbukti.

- $F_5 = O(F_6)$: $\lim_{n\to\infty} \frac{2^{3n}}{(n-3)!}$

Apabila $x_n = \frac{2^{3n}}{(n-3)!}$ maka:

$$\lim_{n\to\infty} \left|\frac{x_{n+1}}{x_n}\right| = \lim_{n\to\infty} \frac{2^{3n+3} \cdot (n-3)!}{(n-2)!\, 2^{3n}} = \frac{8 \cdot \cancel{2^{3n}} \cdot \cancel{(n-3)!}}{\cancel{2^{3n}}\ \cancel{(n-3)!}\,(n-2)} = \lim_{n\to\infty} \frac{8}{n-2}$$

≈ 0, karena 8 dibagi dengan angka yg sangat besar akan mendekati $0 < \infty$

Oleh karena itu terbukti bahwa $F_5 = O(F_6)$.

Oleh karena itu urutan yg ditemukan sesuai dengan sifat transitif: $F_1 = O(F_2)$ .... $F_{x-1} = O(F_x)$

Urutan yg benar adalah $2\lg(n+50)^{20}$, $\ln^2(n+10)$, $\sqrt[3]{n+1}$, $0.002n^5 + 5n^3 + 3$, 2^{3n}, $(n-3)!$ //

Nama: Alvaro A., NPM: 2106752180, Kelas: A, Kode Asdos: 3

(3) a) $n! \in \Theta((n+1)!)$ atau $(n+1)! \in \Theta(n!)$

Salah, pembuktian 2 kasus.

Kasus 1 $\rightarrow$ $f(n) = n!$, $g(n) = (n+1)!$ $\Big]$ $\lim_{n \to \infty} \left(\frac{n!}{(n+1)!}\right) = \lim_{n \to \infty} \left(\frac{\cancel{n!}}{(n+1)\cancel{n!}}\right) = \lim_{n \to \infty} \left(\frac{1}{n+1}\right)$

$=$ dimana $\lim_{n \to \infty} \frac{1}{n+1}$ untuk n sangat besar akan mendekati 0 dan jika definisi limit maka $\lim_{n \to \infty} \frac{1}{n+1} = 0$ //

Kasus 2 $\rightarrow$ $f(n) = (n+1)!$, $g(n)$ $n!$ $\Big]$ $\lim_{n \to \infty} \left(\frac{(n+1)!}{n!}\right) = \left(\frac{(n+1)\, n!}{n!}\right) \rightarrow \lim_{n \to \infty} (n+1)$, dimana hasil limit $n \to \infty$ dari $n+1$ bernilai ∞.

Dapat dilihat bahwa untuk kasus 1 dan kasus 2 bahwa kasus 1 bernilai 0 dan kasus 2 bernilai ∞ dimana melanggar approach limit yaitu $0 < \lim_{n \to \infty} \left(\frac{f(n)}{g(n)}\right) < \infty \Rightarrow f(n) = \Theta(g(n))$

$\Leftarrow$ Maka dari itu, Pernyataan diatas Salah //

Nama : Alvaro Austin

Kelas: A

Kode Asdos : 3

NPM : 2106752180

3b) Benar, kita dapat membuktikan bahwa $\sum_{j=1}^{n} j^k \in \Theta(n^{k+1})$ melalui approach limit: $F(n) = \sum_{j=1}^{n} j^k$

$g(n) = n^{k+1}$

dimana kita harus membuktikan bahwa $\lim_{n\to\infty} \frac{1^k+2^k+3^k+\dots+n^k}{n^k+n^k+n^k+\dots+n^k}$ memiliki nilai diantara 0 dan ∞

kita dapat membandingkan bahwa

$$\lim_{n\to\infty} \frac{1^k+2^k+\dots+n^k}{n^k+n^k+\dots+n^k} < \lim_{n\to\infty} \frac{n^k+n^k+\dots+n^k}{n^k+n^k+\dots+n^k}$$

$$\lim_{n\to\infty} \frac{1^k+2^k+\dots+n^k}{n^k+n^k+\dots+n^k} < 1$$

↳ dimana nilai limit ini pastinya akan lebih besar dari 0, sehingga ~~ini~~ pernyataan $\sum_{j=1}^{n} j^k \in \Theta(n^{k+1})$ benar.

3c) Salah, buktikan dengan direct proof dengan counter example:

Apabila kita mengasumsikan bahwa pernyataan $2^{F(n)} \in \Omega(2^{g(n)})$ benar maka seharusnya pernyataan $F(n) \in \Omega(g(n))$ juga benar. Mari kita cari nilai $f(n)$ dan $g(n)$ sebagai counter example.

mari pilih $F(n) = \frac{1}{n}$ $g(n) = \frac{1}{n}+1$, dimana kita cek nilai dari $2^{F(n)} \in \Omega(2^{g(n)})$

$$\lim_{n\to\infty} \frac{2^{\frac{1}{n}}}{2^{(\frac{1}{n}+1)}} = \lim_{n\to\infty} = \frac{2^{\frac{1}{n}}}{2^{\frac{1}{n}} \cdot 2} = \frac{1}{2} > 0$$

, sehingga kebenaran dari pernyataan $2^{F(n)} \in \Omega(2^{g(n)})$ benar

Namun mari cek kebenaran bahwa $F(n) \in \Omega(g(n))$ benar atau tidak.

$$\lim_{n\to\infty} \left(\frac{F(n)}{g(n)}\right) = \lim_{n\to\infty} \left(\frac{\frac{1}{n}}{\frac{1}{n}+1}\right) = \lim_{n\to\infty} \left(\frac{\frac{1}{n}}{\frac{1+n}{n}}\right) = \lim_{n\to\infty} \left(\frac{1}{1+n}\right)$$

dimana nilai dari limit ini akan bernilai 0

Oleh karena itu karena $\lim_{n\to\infty} \left(\frac{F(n)}{g(n)}\right) = \lim_{n\to\infty} \left(\frac{\frac{1}{n}}{\frac{1}{n}+1}\right) = 0$

Maka dari itu, karena pernyataan $F(n) \in \Omega(g(n))$ salah maka pernyataan diatas juga salah (karena Benar ∩ salah → Salah)

3D) Benar, $F(n) \in O(g(n))$ maka $[F(n)]^2 \in O([g(n)]^2)$ $2^{F(n)} \in \Omega(2^{g(n)})$ $F(n) \in \Omega(g(n))$

↳ $\lim_{n\to\infty} \frac{F(n)}{g(n)} = 0$ maka,

$$\lim_{n\to\infty} \frac{((F(n))^2}{((g(n))^2} = \lim_{n\to\infty} \left(\frac{F(n)}{g(n)}\right)^2 = \lim_{n\to\infty} (0)^2 < \infty$$

$0 < \infty$

Sehingga pernyataan diatas benar

Nama: Alvaro Austin
Kelas : A
NPM : 2106752180
Kode Asisten: 3

(4) a) Algoritma count(n) melakukan perhitungan dan penyimpanan jumlah dari deret angka 1 hingga j^2 untuk setiap j mulai dari 1 hingga $\sqrt{n}$. Penjabaran matematis : $\sum_{j=1}^{\sqrt{n}} \Big(\underbrace{\sum_{i=1}^{j^2} i}_{\text{disimpan pada sum[j]}} \Big)$ pada setiap iterasi j

Contoh keluaran untuk input n: 16,

Kode python:

```
def count (n) :
    k = math. sqrt (n)
    sum = [0]* int(k)
    For j in range(1, int(k)+1):
        sum[j-1] = 0
        For i in range (1, j**2 +1):
            sum[j-1] = sum[j-1] +1
    return sum
```

untuk n = 16 maka

k = 4.0

sum = [0]* int(4.0) # output : [0,0,0,0]

For j in range (1,5):

↳ j=1 → sum[0] = 1
j=2 → sum[1] = 10
j=3 → sum[2] = 45
j=4 → sum[3] = 136

Sehingga outputnya untuk n=16 adalah [1, 10, 45, 136]

Nama : Alvaro Austin , Kelas : A , NPM : 2106752180, Kode asisten : 3

(b) Count (n):

$k = \sqrt{n}$

Count (n):	Operation Cost	
1. $k = \sqrt{n}$	c_1	1
2. let sum [i ... k] be a new array	c_2	1
3. For j , 1 to k	c_3	$\sqrt{n}+1 : k+1$
4. sum[j] = 0	c_4	$\sqrt{n} : k$
5. For i = 1 to j^2	c_5	$\sum_{i=1}^{\sqrt{n}} (i^2+1) :$ $\sum_{i=1}^{k} (i^2+1) : \frac{k(k+1)(2k+1)}{6} + k$ $= k(\frac{1}{3}k^2 + \frac{1}{2}k + \frac{7}{6})$
6. sum[j] = sum[j] + 1	c_6	$\sum_{i=1}^{\sqrt{n}} (i^2) . \sum_{i=1}^{k} : \frac{k(k+1)(2k+1)}{6} = \frac{1}{3}k^3 + \frac{1}{2}k^2 + \frac{1}{6}k$ $= k(\frac{1}{3}k^2 + \frac{1}{2}k + \frac{1}{6})$
7. endfor	-	
8. endfor	-	
9. return sum[1 ... k]	c_7	1

Total running time dari algoritma count() adalah

$$c_1 + c_2 + c_3 k + c_3 + c_4 k + c_5 (k(\tfrac{1}{3}k^2 + \tfrac{1}{2}k + \tfrac{7}{6})) + c_6 (k(\tfrac{1}{3}k^2 + \tfrac{1}{2}k + \tfrac{1}{6})) + c_7$$

$$= k^3 (\tfrac{1}{3}c_5 + \tfrac{1}{3}c_6) + k^2 (\tfrac{1}{2}c_5 + \tfrac{1}{2}c_6) + k(c_3 + c_4 + \tfrac{7}{6}c_5 + \tfrac{1}{6}c_6) + (c_1 + c_2 + c_3 + c_7)$$

karena $k = \sqrt{n}$, maka subsitutsi

$$T(n) = n\sqrt{n}(\tfrac{1}{3}c_5 + \tfrac{1}{3}c_6) + n(\tfrac{1}{2}c_5 + \tfrac{1}{2}c_6) + \sqrt{n}(c_3 + c_4 + \tfrac{7}{6}c_5 + \tfrac{1}{6}c_6) + (c_1 + c_2 + c_3 + c_7)$$

Basically $T(n) = a n\sqrt{n} + bn + c\sqrt{n} + d$

Name: Alvaro Austin, kelas: A, NPM: 2106735480, kode asisten: 3

(c) Tight Bound.

$$\lim_{n\to\infty} \frac{T(n)}{n\sqrt{n}} = \lim_{n\to\infty} \left(\left(\frac{1}{3}c_5 + \frac{1}{3}c_6\right) + \left(\frac{1}{\sqrt{n}}\right)\left(\frac{1}{2}c_5 + \frac{1}{2}c_6\right) + \left(\frac{1}{n}\right)\left(c_3 + c_4 + \frac{7}{6}c_5 + \frac{1}{6}c_6\right) + \left(\frac{1}{n\sqrt{n}}\right)\left(c_1 + c_2 + c_3 + c_9\right)\right).$$

$$= \lim_{n\to\infty} \frac{1}{3}c_5 + \frac{1}{3}c_6$$

Karena

$$0 < \frac{1}{3}c_5 + \frac{1}{3}c_6 < \infty$$

maka $0 < \lim_{n\to\infty} \frac{T(n)}{n\sqrt{n}} < \infty$

dan dapat disimpulkan bahwa $T(n) \in \Theta(n\sqrt{n})$ //