

Nama : Annas Rizal Arisandi

Kelas : SE-02-B

NIM : 18104029

Tugas Desain Analisis Algoritma

1. Algoritma strassen dalam bentuk pseudocode

```
StrassenMatrixMultiplication(matrix A, matrix B)

    if size < cross-over point then
        calculate matrix using naive method
    else
        devide matrix A in sub-matrices A11, A12, A21, A22
        devide matrix B in sub-matrices B11, B12, B21, B22

        P1 is calculated as  $(A_{11}+A_{22}) * (B_{11}+B_{22})$  by
            recursive StrassenMatrixMultiplication
        P2 is calculated as  $(A_{21}+A_{22}) * B_{11}$  by
            recursive StrassenMatrixMultiplication
        P3 is calculated as  $A_{11} * (B_{12}-B_{22})$  by
            recursive StrassenMatrixMultiplication
        P4 is calculated as  $A_{22} * (B_{21}-B_{11})$  by
            recursive StrassenMatrixMultiplication
        P5 is calculated as  $(A_{11}+A_{12}) * B_{22}$  by
            recursive StrassenMatrixMultiplication
        P6 is calculated as  $(A_{21}-A_{11}) * (B_{11}+B_{12})$  by
```

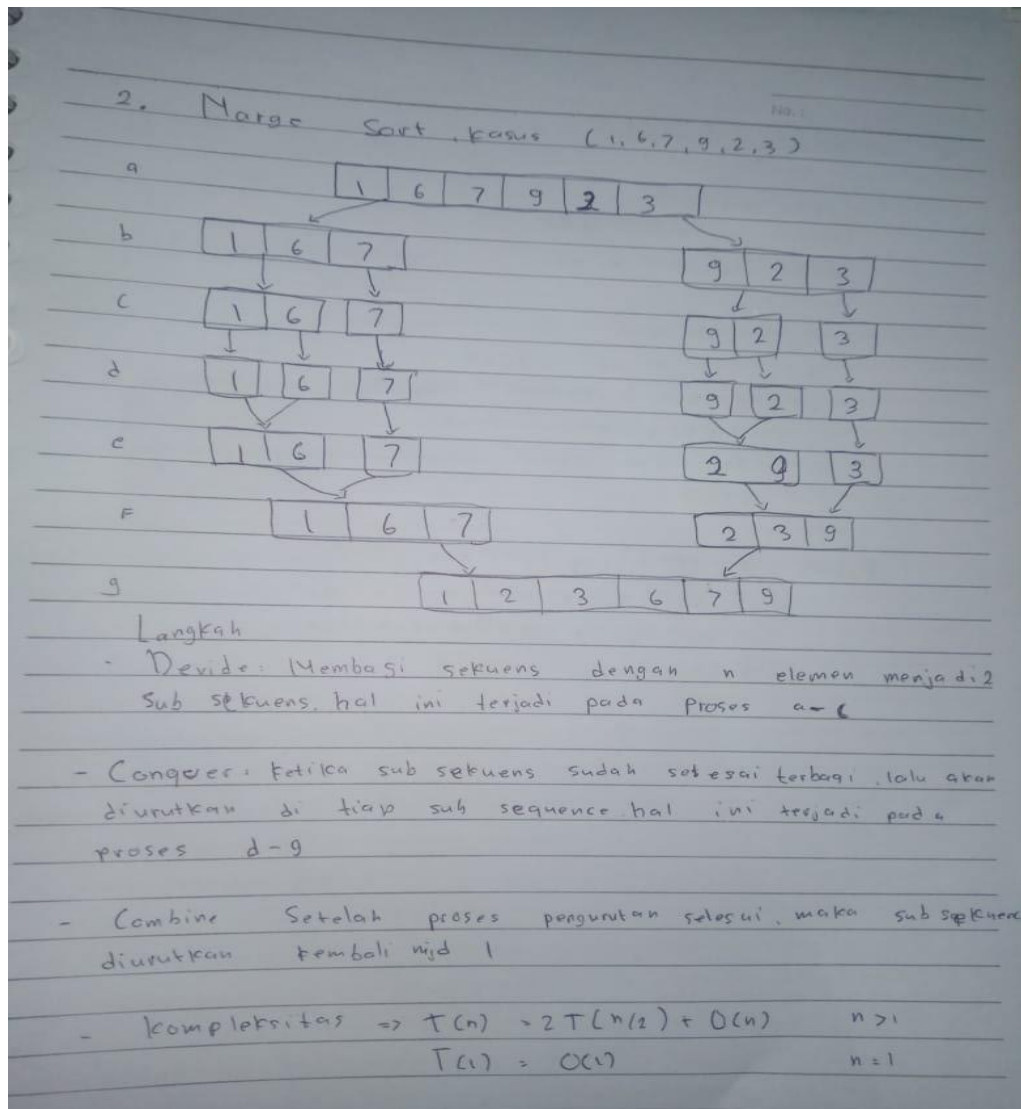
```
recursive StrassenMatrixMultiplication

P7 is calculated as (A12-A22)*(B21+B22) by
recursive StrassenMatrixMultiplication

C11 is calculated as P1+P4-P5+P7
C12 is calculated as P3+P4
C21 is calculated as P2+P4
C22 is calculated as P1+P2+P3+P6

combine sub-matrices C11, C12, C21, C22 in
matrix C and return it as result of function
StrassenMatrixMultiplication
```

2. Cara kerja merge sort untuk kasus (1,6,7,9,2,3) dan hitung kompleksitasnya



3. Tugas baca : algoritma karatsuba (tahapan, algoritme dalam konsep divide and conquer)

4. Gunakan algoritma karatsuba untuk menghitung (1234 * 4321)

9 Algoritma Karatsuba

$$(1234 \times 4321)$$

$$1234 = 1 \cdot 1000 + 234$$

$$4321 = 4 \cdot 1000 + 321$$

$$z_2 = 1 \times 4 = 4$$

$$z_0 = 234 \times 321 = 75114$$

$$\begin{aligned} z_1 &= ((1+234) \times (4+321)) - z_2 - z_0 \\ &= (235 \times 325) - 4 - 75114 \\ &= 76375 - 4 - 75114 \\ &= 1257 \end{aligned}$$

$$\begin{aligned} \text{Result} &= z_2 \cdot (10^3)^2 + z_1 \cdot (10^3)^1 + z_0 \cdot (10^3)^0 \\ &= 4 \cdot 1000^2 + 1257 \cdot 1000^1 + 75114 \cdot 1000^0 \\ &= 4.000.000 + 1257.000 + 75114 \\ &= 5.332.114 \end{aligned}$$

5. Tuliskan cara kerja dan kompleksitas dari binary search, insertion sort, quick sort, bubble sort

Jawab :

-Binary search

Proses pencarian binary search hanya dapat dilakukan pada sekumpulan data yang sudah diurutkan terlebih dahulu. Prinsip dari binary search terhadap N elemen dapat dijelaskan seperti berikut:

1. Tentukan posisi awal = 0 dan posisi akhir = N-1.
2. Hitung posisi tengah = (posisi awal + posisi akhir)/2.
3. Bandingkan data yang dicari dengan elemen posisi tengah.

- Jika sama maka catat posisi dan cetak kemudian berhenti.

- Jika lebih besar maka akan dilakukan pencarian kembali ke bagian kanan dengan posisi awal = p, posisi tengah +1 dan posisi akhir tetap kemudian ulangi mulai poin 2.

- Jika lebih kecil maka akan dilakukan pencarian kembali ke bagian kiri dengan nilai posisi awal tetap dan nilai posisi akhir = posisi tengah-1 kemudian ulangi mulai poin 2.

Kompleksitas :

$$T(n) = T(n/2) + \theta(1)$$

$$a = 1, b = 2, k = 0,$$

$$a = b^k \rightarrow O(n^k \log n)$$

$$\rightarrow T(n) = O(\lg n)$$

- Insertion sort

Algoritma insertion sort pada dasarnya memilah data yang akan diurutkan menjadi dua bagian, yang belum diurutkan dan yang sudah diurutkan. Elemen pertama diambil dari bagian array yang belum diurutkan dan kemudian diletakkan sesuai posisinya pada bagian lain dari array yang telah diurutkan. Langkah ini dilakukan secara

berulang hingga tidak ada lagi elemen yang tersisa pada bagian array yang belum diurutkan.

Kompleksitas : $T(n) = O(n^2)$

- Quick sort

Algoritma ini berdasar pada pola divide-and-conquer. Berbeda dengan merge sort, algoritma ini hanya mengikuti langkah – langkah sebagai berikut :

1. Divide

Memilah rangkaian data menjadi dua sub-rangkaian $A[p \dots q-1]$ dan $A[q+1 \dots r]$ dimana setiap elemen $A[p \dots q-1]$ adalah kurang dari atau sama dengan $A[q]$ dan setiap elemen pada $A[q+1 \dots r]$ adalah lebih besar atau sama dengan elemen pada $A[q]$. $A[q]$ disebut sebagai elemen pivot. Perhitungan pada elemen q merupakan salah satu bagian dari prosedur pemisahan.

2. Conquer

Mengurutkan elemen pada sub-rangkaian secara rekursif. Pada algoritma quicksort, langkah ”kombinasi” tidak di lakukan karena telah terjadi pengurutan elemen – elemen pada sub-array

Kompleksitas : $T(n) = O(n \log_2 n)$

- Bubble sort

Pengurutan yang dilakukan dengan membandingkan masing-masing item dalam suatu list secara berpasangan, menukar item jika diperlukan, dan mengulaginya sampai akhir list secara berurutan, sehingga tidak ada lagi item yang dapat ditukar.

Kompleksitas : $T(n) = O(n^2)$

