EXERCISE 6.2: ALGORITMA PERKALIAN MATRIKS STRASSEN

dikerjakan di rumah, sebelum perkuliahan pertemuan 6

Aturan pengerjaan tugas:

1. Buatlah kelompok diskusi beranggotakan 3 orang.
2. Kerjakan soal yang ada secara singkat, padat, dan jelas. Anda disarankan mengerjakan soal secara terurut, karena setiap soal terhubung satu sama lain dengan level pemahaman materi yang naik.
3. Tugas boleh diketik/ditulis tangan (pastikan bisa dibaca), boleh menggunakan Bahasa Indonesia/Inggris. Hindari menggunakan tinta merah. Pastikan bisa dibaca dengan baik.
4. Tugas dikumpulkan dalam bentuk *hardcopy*.
5. Setiap anggota kelompok **wajib** memahami hasil diskusi dan solusi yang dituliskan oleh kelompoknya.

Dengan ini, Anda menyatakan bahwa Anda siap menerima segala konsekuensi jika nantinya ditemukan adanya kecurangan dalam pengerjaan tugas ini.

1 Perkalian Matriks

1. (*Perkalian matriks persegi dengan menggunakan matriks blok*)

   Ingat kembali algoritma perkalian matriks persegi dengan strategi brute-force. Kita tahu bahwa kompleksitas algoritma tersebut adalah $\mathcal{O}(n^3)$.

   Perkalian dua matriks persegi dapat dilakukan dengan memecah matriks menjadi matriks blok berukuran 2×2, sebagaimana yang ditampilkan pada ilustrasi berikut.

   Misalkan A dan B adalah matriks yang berukuran $n \times n$. Misalkan $C = A \times B$. Matriks A dan B masing-masing dipartisi menjadi empat submatriks dengan ukuran $\frac{n}{2} \times \frac{n}{2}$.

   $$\underbrace{\left[\begin{array}{c:c} A_{11} & A_{12} \\ \hdashline A_{21} & A_{22} \end{array}\right]}_{A} \times \underbrace{\left[\begin{array}{c:c} B_{11} & B_{12} \\ \hdashline B_{21} & B_{22} \end{array}\right]}_{B} = \underbrace{\left[\begin{array}{c:c} C_{11} & C_{12} \\ \hdashline C_{21} & C_{22} \end{array}\right]}_{C}$$

   C dapat dinyatakan sebagai matriks blok berukuran 2×2 yang juga berukuran $\frac{n}{2} \times \frac{n}{2}$. Komponen matriks C dapat dihitung sebagai berikut:

   - $C_{11} = A_{11} \cdot B_{11} + A_{12} \cdot B_{21}$
   - $C_{12} = A_{11} \cdot B_{12} + A_{12} \cdot B_{22}$
   - $C_{21} = A_{21} \cdot B_{11} + A_{22} \cdot B_{21}$
   - $C_{22} = A_{21} \cdot B_{22} + A_{22} \cdot B_{22}$

   Untuk memperjelas pemecahan matriks persegi menjadi matriks blok, perhatikan contoh berikut.

   Contoh 1.1. *Matriks bujur sangkar dapat dipecah sebagai berikut:*

   $$A = \left[\begin{array}{cc:cc} 1 & 21 & 15 & 7 \\ 11 & 3 & 10 & 31 \\ \hdashline 52 & 31 & 2 & 17 \\ 2 & 9 & 23 & 3 \end{array}\right] \qquad A_{11} = \begin{bmatrix} 1 & 21 \\ 11 & 3 \end{bmatrix} \quad A_{12} = \begin{bmatrix} 15 & 7 \\ 10 & 31 \end{bmatrix}$$

   $$A_{21} = \begin{bmatrix} 52 & 31 \\ 2 & 9 \end{bmatrix} \quad A_{22} = \begin{bmatrix} 2 & 17 \\ 23 & 3 \end{bmatrix}$$

2. Cermati dan kaitkan pseudocode berikut dengan penjelasan di atas. Jelaskan proses yang terjadi pada algoritma berikut.

Algorithm 1 Matrix multiplication

```
procedure MMUL(A, B: matrices, n: integer)
    if n = 1 then                                   ▷ The matrices are of size 1 × 1
        return A * B                                ▷ Scalar multiplication
    else
        SPLIT(A)      ▷ SPLIT adalah prosedur untuk memecah matriks menjadi matriks blok berukuran 2 × 2, dengan setiap blok berukuran n/2 × n/2
        SPLIT(B)
        C11 ← MSUM(MMUL(A11, B11, n/2), MMUL(A12, B21, n/2))
        C12 ← MSUM(MMUL(A11, B12, n/2), MMUL(A12, B22, n/2))
        C21 ← MSUM(MMUL(A21, B11, n/2), MMUL(A22, B21, n/2))
        C22 ← MSUM(MMUL(A21, B12, n/2), MMUL(A22, B22, n/2))
    end if
    return C                                        ▷ C is the union of C11, C12, C21, C22
end procedure
```

Prosedur MSUM yang digunakan dalam MMUL adalah sebagai berikut.

Algorithm 2 Sum of two matrices

procedure MSUM(A, B: matrices, n: integer)
    for $i \leftarrow 1$ **to** n **do**
        for $j \leftarrow 1$ **to** n **do**
            $C[i,j] \leftarrow A[i,j] + B[i,j]$
        end for
    end for
end procedure

3. Berapakah kompleksitas waktu algoritma perkalian matriks persegi tersebut? Apakah ada peningkatan dibandingkan dengan strategi brute-force?
   (*Hint: Gunakan Teorema Master*)

2 Algoritma perkalian matriks Strassen

1. (*Analisis algoritma perkalian matriks dengan pemecahan menjadi matriks blok*)

   Jika kalkulasi Anda benar, Anda akan menemukan bahwa tidak ada perbedaan yang signifikan pada komplesitas waktu algoritma di atas dibandingkan dengan strategi brute-force.

   Ide Volker Strassen adalah untuk mengurangi banyaknya 'perkalian' dalam prosedur. Karena biaya 'perkalian' lebih 'mahal' daripada 'penambahan' (lihat `https://www.wikiwand.com/en/Computational_complexity_of_mathematical_operations`).

   Operasi yang terjadi pada algoritma sebelumnya terdiri dari 8 perkalian dan 4 penjumlahan:

   - $C_{11} = A_{11} \cdot B_{11} + A_{12} \cdot B_{21}$
   - $C_{12} = A_{11} \cdot B_{12} + A_{12} \cdot B_{22}$
   - $C_{21} = A_{21} \cdot B_{11} + A_{22} \cdot B_{21}$
   - $C_{22} = A_{21} \cdot B_{22} + A_{22} \cdot B_{22}$

   Strassen memodifikasi persamaan di atas untuk menguranginya menjadi 7 perkalian tetapi dengan konsekuensi terdapat lebih banyak penjumlahan. Dapatkah Anda memperkirakan mengapa teknik ini berpotensi memperbaiki kompleksitas waktunya?

2. (*Modifikasi oleh Strassen*)

   Modifikasi perkalian matriks oleh Strassen adalah sebagai berikut:

   - $M_1 = (A_{11} - A_{22})(B_{21} + B_{22})$
   - $M_2 = (A_{11} + A_{22})(B_{11} + B_{22})$
   - $M_3 = (A_{11} - A_{21})(B_{11} + B_{12})$
   - $M_4 = (A_{11} + A_{12})B_{22}$
   - $M_5 = A_{11}(B_{12} - B_{22})$
   - $M_6 = A_{22}(B_{21} - B_{11})$
   - $M_7 = (A_{21} + A_{22})B_{11}$

Sehingga:

- $C_{11} = M_1 + M_2 - M_4 + M_6$
- $C_{12} = M_4 + M_5$
- $C_{21} = M_6 + M_7$
- $C_{22} = M_2 - M_3 + M_5 - M_7$

Dalam pseudocode, teknik ini dapat dituliskan sebagai berikut.

Algorithm 3 Matrix multiplication

procedure STRASSEN(A, B: matrices, n: integer)
    if $n = 1$ **then return** $A * B$ ▷ *Scalar multiplication*
    else
        SPLIT(A)
        SPLIT(B)
        $M_1 \leftarrow$ STRASSEN($A_{12} - A_{22}, B_{21} + B_{22}, \frac{n}{2}$)
        $M_2 \leftarrow$ STRASSEN($A_{11} + A_{22}, B_{11} + B_{22}, \frac{n}{2}$)
        $M_3 \leftarrow$ STRASSEN($A_{11} - A_{21}, B_{11} + B_{12}, \frac{n}{2}$)
        $M_4 \leftarrow$ STRASSEN($A_{11} + A_{12}, B_{22}, \frac{n}{2}$)
        $M_5 \leftarrow$ STRASSEN($A_{11}, B_{12} - B_{22}, \frac{n}{2}$)
        $M_6 \leftarrow$ STRASSEN($A_{22}, B_{21} - B_{11}, \frac{n}{2}$)
        $M_7 \leftarrow$ STRASSEN($A_{21} + A_{22}, B_{11}, \frac{n}{2}$)
        $C_{11} \leftarrow M_1 + M_2 - M_4 + M_6$
        $C_{12} \leftarrow M_4 + M_5$
        $C_{21} \leftarrow M_6 + M_7$
        $C_{22} \leftarrow M_2 - M_3 + M_5 - M_7$
    end if
    return C ▷ *C is the union of $C_{11}, C_{12}, C_{21}, C_{22}$*
end procedure

Cermati dan analisis proses yang terjadi pada algoritma di atas. Jelaskan!

3. (*Kompleksitas waktu algoritma Strassen*)

   Gunakan Teorema Master untuk menganalisis kompleksitas waktu algoritma Strassen di atas. apakah ada perbedaan dengan hasil yang diperoleh dibandingkan dengan algoritma perkalian matriks yang sebelumnya?