

---

## EXERCISE 4: DIVIDE/DECREASE/TRANSFORM-AND-CONQUER

dikerjakan saat pertemuan tatap muka

---

### Aturan pengerjaan tugas:

1. Buatlah kelompok diskusi beranggotakan 6 orang (lintas kelas).
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. Jika menggunakan tulis tangan, harap discan (tidak difoto), kemudian dikompresi untuk memperkecil ukuran file. Tulis jawaban pada satu file pdf.
4. Pengumpulan tugas melalui e-learning Undiksha. Format penamaan ada di e-learning.
5. Setiap anggota kelompok **wajib** memahami hasil diskusi dan solusi yang dituliskan oleh kelompoknya.
6. Di akhir perkuliahan, **setiap mahasiswa** wajib memberikan ulasan tentang hasil diskusi kelompoknya dalam bentuk video **mandiri (tidak berkelompok)** berdurasi  $\pm 10$  menit yang diunggah di Youtube. Ulasan memuat hasil diskusi kelompok, apa yang Anda pelajari, apa yang Anda tidak pahami, dan hal-hal lain yang Anda pandang perlu.

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

# 1 Polynomials multiplication

Dalam latihan ini, kita menyelidiki pendekatan divide-and-conquer untuk mengalikan dua polinomial dengan derajat<sup>1</sup> yang sama, yaitu  $n$  (ini mirip dengan “perkalian matriks” dan “perkalian bilangan besar” yang sudah dibahas sebelumnya).

## 1. (Algoritma naif untuk perkalian polinomial)

Diberikan dua polinomial berderajat  $n$  sebagai berikut. Tujuan kita adalah menghitung  $A(x)B(x)$ .

$$A(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$
$$B(x) = b_0 + b_1x + b_2x^2 + \dots + b_nx^n$$

Cara naif untuk melakukan perkalian polinomial adalah dengan *metode perkalian langsung*, seperti pada contoh berikut:

### Example

$$A(x) = 1 + 2x + 3x^2$$
$$B(x) = 3 + 2x + 2x^2$$

$$A(x)B(x) = (1 + 2x + 3x^2)(3 + 2x + 2x^2) = 3 + 8x + 15x^2 + 10x^3 + 6x^4 \quad \square$$

(a) Selesaikan perkalian polinomial berikut menggunakan algoritma naif.

$$A(x) = 2 + 5x + 3x^2 + x^3 - x^4$$
$$B(x) = 1 + 2x + 2x^2 + 3x^3 + 6x^4$$

(b) Tulis pseudocode untuk algoritma naif untuk mengalikan dua polinomial  $A(x)$  dan  $B(x)$  yang memiliki derajat  $n$ .

(c) Hitung kompleksitas algoritma naif Anda. Nyatakan dengan notasi asimtotik!

## 2. (Polynomials multiplication divide-and-conquer algorithm)

Diberikan dua polinomial dengan derajat  $n$  sebagai berikut. Tujuan kita adalah menghitung  $A(x)B(x)$ .

$$A(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$
$$B(x) = b_0 + b_1x + b_2x^2 + \dots + b_nx^n$$

Bagaimana kita melakukan perkalian polinomial dengan Divide-and-Conquer? Algoritmanya adalah sebagai berikut.

- Partisi  $A(x)$  menjadi  $A_0(x)$  dan  $A_1(x)$ , masing-masing berisi  $n/2$  suku:

$$A_0(x) = a_0 + a_1x + a_2x^2 + \dots + a_{\lceil n/2 \rceil - 1}x^{\lceil n/2 \rceil - 1}$$
$$A_1(x) = a_{\lceil n/2 \rceil} + a_{\lceil n/2 \rceil + 1}x + a_{\lceil n/2 \rceil + 2}x^2 + \dots + a_{n - \lceil n/2 \rceil}x^{n - \lceil n/2 \rceil}$$

Sehingga

$$A(x) = A_0(x) + A_1(x)x^{\lceil n/2 \rceil}$$

<sup>1</sup>Derajat suatu polinomial adalah pangkat tertinggi dari variabel pada suku-suku polinomial tersebut (misa: polinomial  $3x^3 + 2x - 4$  memiliki derajat 3).

- Demikian pula,  $B(x)$  dapat dipartisi menjadi  $B_0(x)$  dan  $B_1(x)$ , sehingga:

$$B(x) = B_0(x) + B_1(x)x^{\lceil n/2 \rceil}$$

Dengan demikian:

$$A(x)B(x) = A_0(x)B_0(x) + (A_0(x)B_1(x) + A_1(x)B_0(x))x^{\lceil n/2 \rceil} + A_1(x)B_1(x)x^{2\lceil n/2 \rceil}$$

**Tugas:** Selesaikan perkalian polinomial berikut menggunakan algoritma yang dijelaskan di atas. Tuliskan langkah-langkahnya dengan jelas!

$$A(x) = 2 + 5x + 3x^2 + x^3 - x^4$$

$$B(x) = 1 + 2x + 2x^2 + 3x^3 + 6x^4$$

### 3. (DnC-based polynomials multiplication: pseudocode and time complexity)

Algoritma divide-and-conquer yang dijelaskan pada pertanyaan 3 dapat ditulis dalam pseudocode sebagai berikut:

---

**Algorithm 1** Polynomials multiplication (divide-and-conquer, version 1)

---

```

1: procedure POLYMUL( $A, B$ : polynomials,  $n$ : integer)
2:   declaration
3:      $A_0, A_1, B_0, B_1$ : polynomials
4:      $s$ : integer
5:   end declaration
6:   if  $n = 0$  then return  $A * B$  ▷ scalar multiplication
7:   else
8:      $s \leftarrow \lceil n/2 \rceil$ 
9:      $A_0 \leftarrow a_0 + a_1x + a_2x^2 + \dots + a_{s-1}x^{s-1}$ 
10:     $A_1 \leftarrow a_sx^s + a_{s+1}x^{s+1} + a_{s+2}x^{s+2} + \dots + a_nx^{n-s}$ 
11:     $B_0 \leftarrow b_0 + b_1x + b_2x^2 + \dots + b_{s-1}x^{s-1}$ 
12:     $B_1 \leftarrow b_sx^s + b_{s+1}x^{s+1} + b_{s+2}x^{s+2} + \dots + b_nx^{n-s}$ 
13:    return POLYMUL( $A_0, B_0, s$ ) + POLYMUL( $A_0, B_1, s$ ) + POLYMUL( $A_1, B_0, s$ ) *  $x^s$  + POLYMUL( $A_1, B_1, s$ ) *  $x^{2s}$ 
14:  end if
15: end procedure

```

---

- Periksa apakah pseudocode tersebut sesuai dengan perhitungan matematis yang dijelaskan di pertanyaan 3.
- Tuliskan fungsi kompleksitas waktu dari algoritma divide-and-conquer pada pertanyaan 3 dalam rumus rekursif. Menggunakan Teorema Master, hitung kompleksitas waktu asimptotik!

### 4. (DnC-based polynomials multiplication: improvement)

Sekarang kita ingin memodifikasi algoritma divide-and-conquer untuk perkalian polinomial yang diberikan pada pertanyaan 2. Kita akan mengurangi jumlah perkalian yang dilakukan dalam algoritma. Pada soal 2, kita mengetahui formula berikut:

$$A(x)B(x) = A_0(x)B_0(x) + (A_0(x)B_1(x) + A_1(x)B_0(x))x^{\lceil n/2 \rceil} + A_1(x)B_1(x)x^{2\lceil n/2 \rceil}$$

Terdapat 4 perkalian dan 3 penjumlahan polinomial dengan derajat  $n$ . Kita akan mengurangi jumlah perkalian menjadi 3, tetapi dengan konsekuensi jumlah penjumlahan bertambah.

Definisikan:

$$\begin{aligned} Y(x) &= (A_0(x) + A_1(x)) \times (B_0(x) + B_1(x)) \\ U(x) &= A_0(x)B_0(x) \\ Z(x) &= A_1(x)B_1(x) \end{aligned}$$

Maka:

$$Y(x) - U(x) - Z(x) = A_0(x)B_1(x) + A_1(x)B_0(x)$$

sehingga:

$$\begin{aligned} A(x)B(x) &= A_0(x)B_0(x) + (A_0(x)B_1(x) + A_1(x)B_0(x))x^{\lceil n/2 \rceil} + A_1(x)B_1(x)x^{2\lceil n/2 \rceil} \\ &= U(x) + (Y(x) - U(x) - Z(x))x^{\lceil n/2 \rceil} + Z(x)x^{2\lceil n/2 \rceil} \end{aligned}$$

Perhatikan bahwa dalam algoritma ini, hanya ada tiga perkalian, yaitu  $U(x)$ ,  $Y(x)$ , dan  $Z(x)$ .

**Tugas:** Selesaikan perkalian polinomial berikut menggunakan algoritma yang dijelaskan di atas. Tuliskan langkah-langkahnya dengan jelas!

$$\begin{aligned} A(x) &= 2 + 5x + 3x^2 + x^3 - x^4 \\ B(x) &= 1 + 2x + 2x^2 + 3x^3 + 6x^4 \end{aligned}$$

### 5. (DnC poly-multiplication improvement: pseudocode)

Algoritma divide-and-conquer yang dijelaskan pada pertanyaan 3 dapat ditulis dalam pseudocode sebagai berikut:

---

#### Algorithm 2 Polynomials multiplication (divide-and-conquer, version 2)

---

```

1: procedure POLYMUL2( $A, B$ : polynomials,  $n$ : integer)
2:   declaration
3:      $A_0, A_1, B_0, B_1, U, Y, Z$ : polynomials
4:      $s$ : integer
5:   end declaration
6:   if  $n = 0$  then
7:     return  $A * B$  ▷ scalar multiplication
8:   else
9:      $s \leftarrow \lceil n/2 \rceil$ 
10:     $A_0 \leftarrow a_0 + a_1x + a_2x^2 + \dots + a_{s-1}x^{s-1}$ 
11:     $A_1 \leftarrow a_sx^s + a_{s+1}x^{s+1} + a_{s+2}x^{s+2} + \dots + a_nx^{n-s}$ 
12:     $B_0 \leftarrow b_0 + b_1x + b_2x^2 + \dots + b_{s-1}x^{s-1}$ 
13:     $B_1 \leftarrow b_sx^s + b_{s+1}x^{s+1} + b_{s+2}x^{s+2} + \dots + b_nx^{n-s}$ 
14:     $Y \leftarrow \text{POLYMUL2}(A_0 + A_1, B_0 + B_1, s)$ 
15:     $U \leftarrow \text{POLYMUL2}(A_0, B_0, s)$ 
16:     $Z \leftarrow \text{POLYMUL2}(A_1, B_1, s)$ 
17:    return  $U + (Y - U - Z) * x^s + Z * x^{2s}$ 
18:  end if
19: end procedure

```

---

**Tugas:** Selesaikan perkalian polinomial berikut dengan algoritma POLYMUL2. Tuliskan langkah-langkahnya dengan jelas!

$$A(x) = 2 + 5x + 3x^2 + x^3 - x^4$$

$$B(x) = 1 + 2x + 2x^2 + 3x^3 + 6x^4$$

6. Setelah menghitung hasil  $A(x)B(x)$ , periksa apakah ketiga algoritma berbeda di atas memberikan hasil yang sama. Jika tidak, jelaskan alasannya!