

**Laporan Praktikum**  
**BRUTE FORCE DAN DIVIDE CONQUER**



Disusun oleh:

Tomi Martino Affandi

2341720245

D-IV Teknik Informatika - 1E

Jl. Soekarno Hatta No.9, Jatimulyo, Kec. Lowokwaru, Kota Malang Jawa Timur 65141

Phone : (0341)404424, 404425

E-mail : Polinema.ac.id

## 4.1 Tujuan Praktikum

Setelah melakukan materi praktikum ini, mahasiswa mampu:

1. Mahasiswa mampu membuat algoritma brute force dan divide-conquer
2. Mahasiswa mampu menerapkan penggunaan algoritma brute force dan divide-conquer

## 4.2 Menghitung Nilai Faktorial dengan Algoritma Brute Force dan Divide and Conquer

### 1. Code

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/tree/main/Jobsheet%204>

### 2. Output

```
-----
Masukan jumlah elemen:
3
Masukan nilai data ke-1:
5
Masukan nilai data ke-2:
8
Masukan nilai data ke-3:
3
HASIL - BRUTE FORCE
Hasil perhitungan faktorial menggunakan brute force adalah: 120
Hasil perhitungan faktorial menggunakan brute force adalah: 40320
Hasil perhitungan faktorial menggunakan brute force adalah: 6
HASIL - DIVIDE CONQUER
Hasil perhitungan faktorial menggunakan Devide dan Conquer adalah: 120
Hasil perhitungan faktorial menggunakan Devide dan Conquer adalah: 40320
Hasil perhitungan faktorial menggunakan Devide dan Conquer adalah: 6
PS D:\TUGAS KULIAH\SEMESTER 2\Algoritma & Struktur Data>
```

### Pertanyaan

1. Pada bagian if merupakan base case yaitu untuk berhentinya perulangan, Dimana perulangan akan berhenti Ketika nilai n sama dengan 1 dan akan mengembalikan nilai 1. Sementara Pada bagian else, algoritma mengeksekusi perhitungan faktorial jika kondisi pada if sebelumnya tidak terpenuhi, yang berarti n tidak sama dengan 1.
2. Bisa, contoh menggunakan while:

**Code:**

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/a380a5ccb1b58793c715e5d8cd442de296aaf11f>

**Output:**

```

Masukan jumlah elemen:
3
Masukan nilai data ke-1:
5
Masukan nilai data ke-2:
8
Masukan nilai data ke-3:
3
HASIL - BRUTE FORCE
Hasil perhitungan faktorial menggunakan brute force adalah: 120
Hasil perhitungan faktorial menggunakan brute force adalah: 40320
Hasil perhitungan faktorial menggunakan brute force adalah: 6

```

Contoh menggunakan do while:

**Code:**

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/f8b76fb5e605ce8137327033807801dd388095bc>

**Output:**

```

Masukan jumlah elemen:
3
Masukan nilai data ke-1:
5
Masukan nilai data ke-2:
8
Masukan nilai data ke-3:
3
HASIL - BRUTE FORCE
Hasil perhitungan faktorial menggunakan brute force adalah: 120
Hasil perhitungan faktorial menggunakan brute force adalah: 40320
Hasil perhitungan faktorial menggunakan brute force adalah: 6

```

3. Perbedaannya yaitu, fakto\*=i; akan melakukan operasi perkalian dari bilangan terkecil hingga mendekati n sementara int fakto = n \* faktorialDC(n-1); akan melakukan operasi perkalian dari value n hingga mendekati batas tertentu. Dalam kasus ini batasnya adalah 1

#### 4.3 Menghitung Hasil Pangkat dengan Algoritma Brute Force dan Divide and Conquer

**Code:**

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/57375ec48accdef0ebfdbbe291da14d1e099c58ad>

**Output:**

```

=====
Masukan jumlah elemen yang dihitung:
2
Masukan nilai yang hendak dipangkatkan:
6
Masukan nilai pemangkat:
2
Masukan nilai yang hendak dipangkatkan:
4
Masukan nilai pemangkat:
3
HASIL PANGKAT - BRUTE FORCE:
Hasil dari 6 pangkat 2 adalah: 36
Hasil dari 4 pangkat 3 adalah: 64
HASIL PANGKAT - DIVIDED and CONQUER
Hasil dari 6 pangkat 2 adalah: 36
Hasil dari 4 pangkat 3 adalah: 64
PS D:\TUGAS KULIAH\SEMESTER 2\Algoritma & Struktur Data>

```

### Pertanyaan

1. **pangkatBF()**, Melakukan operasi pangkat dimulai dari kiri ke kanan atau dari pangkat terkecil ke pangkat terbesar  
**pangkatDC()**. Melakukan operasi pangkat dengan memisahkan angka berpangkat menjadi pangkat yang lebih kecil, setelah itu melakukan operasi perkalian pada setiap angka berpangkat yang sudah dipisahkan, setelah hasil nya ketemu akan semua bagian akan digabungkan dan akan menyelesaikan operasi angka berpangkat paling besar
2. Sudah, combine dilakukan pada operasi bilangan berpangkat genap
3. **Code:**  
<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/3b509fcacdc1d4a6d91178bdd00edb4518aebfb7>

### Output:

```

Masukan jumlah elemen yang dihitung: 1
Masukan nilai yang hendak dipangkatkan: 6
Masukan nilai pemangkat: 2
HASIL PANGKAT - BRUTE FORCE:
Hasil dari 6 pangkat 2 adalah: 36
HASIL PANGKAT - DIVIDED and CONQUER
Hasil dari 6 pangkat 2 adalah: 36
HASIL PANGKAT - BRUTE FORCE & DIVIDED and CONQUER
HASIL PANGKAT - BRUTE FORCE: 36
HASIL PANGKAT - DIVIDED and CONQUER: 36
PS D:\TUGAS KULIAH\SEMESTER 2\Algoritma & Struktur Data>

```

4. Code:

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/1afa442e74a765e2844f8d913ce5115a99e50619>

Output:

```

=====
Masukan jumlah elemen yang dihitung: 1
Masukan nilai yang hendak dipangkatkan: 6
Masukan nilai pemangkat: 2
Pilih Metode:
1. Brute Force
2. Divided and Conquer
3. Exit
Input: 1
HASIL PANGKAT - BRUTE FORCE:
Hasil dari 6 pangkat 2 adalah: 36
Pilih Metode:
1. Brute Force
2. Divided and Conquer
3. Exit
Input: 2
HASIL PANGKAT - DIVIDED and CONQUER
Hasil dari 6 pangkat 2 adalah: 36
Pilih Metode:
1. Brute Force
2. Divided and Conquer
3. Exit
Input: 3
anda telah keluar
PS D:\TUGAS KULIAH\SEMESTER 2\Algoritma & Struktur Data>

```

#### 4.4 Menghitung Sum Array dengan Algoritma Brute Force dan Divide and Conquer

Code:

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/6c3f39e79d0f4603cec72454455816c37b075b9e>

Output:

```
=====
Program menghitung keuntungan total(Satuan Juta. misal 5.9)
Masukan jumlah bulan: 5
=====
Masukan untung bulan ke-1= 8.5
Masukan untung bulan ke-2= 9.54
Masukan untung bulan ke-3= 7.2
Masukan untung bulan ke-4= 9.1
Masukan untung bulan ke-5= 6
=====
Algoritma Brute Force
Total keuntungan perusahaan selama 5bulan adalah = 40.339999999999996
=====
Algoritma Divide Conquer
Total keuntungan perusahaan selama 5bulan adalah = 40.34
PS D:\TUGAS KULIAH\SEMESTER 2\Algoritma & Struktur Data> 
```

#### Pertanyaan

1. Berfungsi sebagai combine yaitu menghitung keseluruhan array setelah melakukan divide dan solve
2. Variabel mid pada metode totalDC berfungsi untuk menentukan indeks tengah dari rentang array yang diberikan.
3. **Code:**  
<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/a540b508d75541fa00999a0b0489e53d04c8a02b>  
**Output:**

```

=====
Masukan Jumlah Perusahaan : 2
=====
Program Menghitung Keuntungan Total (Satuan Juta Misal 5.9)
Perusahaan ke-1:
Masukan Jumlah Bulan : 2
=====
Masukan Untung Bulan ke-1 = 9
Masukan Untung Bulan ke-2 = 10
=====
Algoritma Brute Force
Total Keuntungan Perusahaan ke-1 selama 2 bulan adalah = 19.0
=====
Algoritma Divide Conquer
Total Keuntungan Perusahaan ke-1 selama 2 bulan adalah = 19.0
=====
Program Menghitung Keuntungan Total (Satuan Juta Misal 5.9)
Perusahaan ke-2:
Masukan Jumlah Bulan : 6
=====
Masukan Untung Bulan ke-1 = 10
Masukan Untung Bulan ke-2 = 11
Masukan Untung Bulan ke-3 = 12
Masukan Untung Bulan ke-4 = 13
Masukan Untung Bulan ke-5 = 14
Masukan Untung Bulan ke-6 = 15
=====
Algoritma Brute Force
Total Keuntungan Perusahaan ke-2 selama 6 bulan adalah = 75.0
=====
Algoritma Divide Conquer
Total Keuntungan Perusahaan ke-2 selama 6 bulan adalah = 75.0
PS D:\TUGAS KULIAH\SEMESTER 2\Algoritma & Struktur Data>

```

## Latihan Soal

Code:

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/83a2381e1d897ba194119ee69b7041421a24bfa2>

Output

```

Top acceleration tertinggi: 6816
Top acceleration terendah: 3700
Rata-rata top power: 4683.75
PS D:\TUGAS KULIAH\SEMESTER 2\Algoritma & Struktur Data>

```