LAPORAN TUGAS PRAKTIKUM
JOBSHEET 4
BRUTE FORCE & DIVIDE CONQUER

Disusun Oleh :

DHANISA PUTRI MASHILFA

NIM. 2341720212

TI-1E/07

D-IV TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

Jl. Soekarno Hatta No. 9 Jatimulyo, Kecamatan Lowokwaru, Jatimulyo, Kec. Lowokwaru, Kota Malang, Jawa Timur 6514

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

4.2.1 Praktikum Percobaan 4.2

4.2.2 Hasil Praktikum Percobaan 4.2

```
--------------------
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 Penghitungan Menggunakan Brute Force Adalah 120
Hasil Penghitungan Menggunakan Brute Force Adalah 40320
Hasil Penghitungan Menggunakan Brute Force Adalah 6
HASIL - DIVIDE AND CONQUER
Hasil Penghitungan Menggunakan Divide and Conquer Adalah 120
Hasil Penghitungan Menggunakan Divide and Conquer Adalah 40320
Hasil Penghitungan Menggunakan Divide and Conquer Adalah 6
PS C:\Users\asus\Documents\Semester2\Algoritma dan Struktur Data\Aljabar-Linier-dan-Sturktur-Data\Jobsheet4 (BruteForceDivideConquer)>
```

4.2.3 Pertanyaan

1. Pada base line Algoritma Divide Conquer untuk melakukan pencarian nilai faktorial, jelaskan perbedaan bagian kode pada penggunaan if dan else!

- **If :** digunakan untuk base case pada algoritma yang bertujuan memeriksa nilai input (n) sama dengan 1, apabila nilainya sama maka nilai 1 atau di kembalikan menjadi hasil faktorial.
- **Else :** apabila bukann base case (nilai inputan tidak sama dengan 1) algoritma akan membagi nilai input (n) menjadi n/2 dan n-1, dari ua bagian itu algoritma akan memnanggil dirinya sendiri secara rekfrusif dan hasilnya dikalikan untuk mendapatkan hasil faktorial dari nilai input(n).

2. Apakah memungkinkan perulangan pada method faktorialBF() diubah selain menggunakan for? Buktikan!

Method **faktorialBF()** bisa diubah selain for yaitu menggunakan **do-while loop** atau **while loop**, atau bahkan **rekrusif** untuk mendapatkan hasil yang sama.

```java
public class Faktorial {

    public int nilai;

    int faktorialBf(int n) {
        int fakto = 1;
        int i = 1;
        while (i <= n) {
            fakto = fakto * i;
            i++;
        }
```

```
        return fakto;
    }

    int fakorialDC(int n) {
        if(n==1) {
            return 1;
        }
        else{
            int fakto = n * fakorialDC(n-1);
            return fakto;
        }
    }
}
```

3. Jelaskan perbedaan antara fakto *= i; dan int fakto = n * faktorialDC(n-1); !

Terdapat perbedaan dalam waktu eksekusinya, pada **fakto *=i;** lebih cepat eksekusinya karena merupakan perkalian sederhana. Sedangkan, pada **int fakto = n * faktorialDC(n-1);** prose eksekusinya lebih lambat karena melibatkan pemanggilan fungsi **faktorialDC(n-1);** yang sifatnya rekrusif.

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

4.3.1 Praktikum Percobaan 4.3

4.3.2 Hasil Praktikum Perobaan 4.3

```
==============================
Masukan Jumlah Elemen Yang Akan Dihitung : 2

Masukan Nilai Yang Hendak Dipangkatkan : 6
Masukan Nilai Pemangkatan : 2

Masukan Nilai Yang Hendak Dipangkatkan : 4
Masukan Nilai Pemangkatan : 3

*****HASIL PANGKAT - BRUTE FORCE*****
Hasil dari 6 pangkat 2 adalah 36
Hasil dari 4 pangkat 3 adalah 64

*****HASIL PANGKAT - DIVIDE AND CONQUER*****
Hasil dari 6 pangkat 2 adalah 36
Hasil dari 4 pangkat 3 adalah 64
PS C:\Users\asus\Documents\Semester2\Algoritma dan Struktur D
```

4.2.3 Pertanyaan

1. Jelaskan mengenai perbedaan 2 method yang dibuat yaitu PangkatBF() dan PangkatDC()!

Perbedaannya terdapat dalam method yang digunakan. **PangkatBF()** menggunakan method iteratif (for), sedangkan **PangkatDC()** menggunakan method rekrusif (perhitungan dipecahkan menjadi lebih kecil lagi).

2. Apakah tahap combine sudah termasuk dalam kode tersebut?Tunjukkan!

Sudah termasuk dalam kode,

```
if(n%2==1){ //bilangan hanjil
                return(pangkatDC(a, n/2) * pangkatDC(a, n/2)*a);
            } else {
                return (pangkatDC(a, n/2) * pangkatDC(a, n/2));
            }
```

4. Modifikasi kode program tersebut, anggap proses pengisian atribut dilakukan dengan konstruktor.

```
public class Pangkat {

    public int nilai, pangkat;

    public Pangkat(int nilai, int pangkat){
        this.nilai = nilai;
        this.pangkat = pangkat;
    }
}
```

5. Tambahkan menu agar salah satu method yang terpilih saja yang akan dijalankan menggunakan switch-case!

```
switch (pilih) {
            case 1:
                System.out.println("Hasil pangkat BRUTE FORCE");
                for (int i = 0; i < elemen; i++) {
                    System.out.println("Hasil dari " +
                            png[i].nilai + " pangkat " +
                            png[i].pangkat + " adalah " +
                            png[i].pangkatBF(png[i].nilai, png[i].pangkat));
                }
                break;
            case 2:
                System.out.println("Hasil pangkat Divide Conquer");
                for (int i = 0; i < elemen; i++) {
                    System.out.println("Hasil dari " +
                            png[i].nilai + " pangkat " +
                            png[i].pangkat + " adalah " +
                            png[i].pangkatDC(png[i].nilai, png[i].pangkat));
                }
                break;
```

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

4.4.1 Praktikum Percobaan 4.4

4.4.2 Hasil Praktikum Percobaan 4.4

```
==============================================
Program Menghitung Keuntungan Total (Satuan Juta, Mulai 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 5 Bulan Adalah 40.339999999999996
==============================================
ALGORITMA DIVIDE AND CONQUER
Total Keuntungan Perusahaan Selama 5 Bulan Adalah 40.34
```

4.4.3 Pertanyaan

1. Mengapa terdapat formulasi return value berikut? Jelaskan!

```
return lsum+rsum+arr[mid];
```

Struktur algoritma Divide and Conquer digunakan untuk mengimplementasikan method **PangkatDC().** Formulasi diatas digunakan untuk mengembalikan nilai total dari pangkat dalam array untuk konteks method **PangkatC().**

2. Kenapa dibutuhkan variable mid pada method TotalDC()?

Sebagai penanda dari letak tengah rentan suatu elemen yang terdapat dalam method **TotalDC()**

3. Program perhitungan keuntungan suatu perusahaan ini hanya untuk satu perusahaan saja. Bagaimana cara menghitung sekaligus keuntungan beberapa bulan untuk beberapa perusahaan.(Setiap perusahaan bisa saja memiliki jumlah bulan berbeda-beda)? Buktikan dengan program!

```
============================================================
Program menghitung keuntungan total(satuan JUTA. Misal 5.9)
Masukkan jumlah perusahaan: 2
============================================================
Masukkan jumlah bulan perusahaan 1 : 3
============================================================
Masukkan untung bulan ke 1 : 9
Masukkan untung bulan ke 2 : 8.90
Masukkan untung bulan ke 3 : 7.50
============================================================
Masukkan jumlah bulan perusahaan 2 : 5
============================================================
Masukkan untung bulan ke 1 : 3
Masukkan untung bulan ke 2 : 2.99
Masukkan untung bulan ke 3 : 5.5
Masukkan untung bulan ke 4 : 4.70
Masukkan untung bulan ke 5 : 5.89
ALGORITMA BRUTE FORCE
Total Keuntungan Perusahaan 1 Selama 3 Bulan Adalah 25.4

Total Keuntungan Perusahaan 2 Selama 5 Bulan Adalah 22.080000000000002

ALGORITMA DIVIDE AND CONQUER
Total Keuntungan Perusahaan 1 Selama 3 Bulan Adalah 25.4

ALGORITMA DIVIDE AND CONQUER
Total Keuntungan Perusahaan 2 Selama 5 Bulan Adalah 22.08
```

4.5 LATIHAN PRAKTIKUM

4.5.1 Hasil Dari Latihan Praktikum 4.5

```
**************************DATA SHOWROOM**************************
------------------------------------------------------------------
Mobil Dengan Acceslaration Tertinggi Adalah :
Merk              : BMW
Tipe              : M2 Coupe
Tahun             : 2016
Top Acceslation   : 6816
Top Power         : 728
------------------------------------------------------------------
Mobil Dengan Acceslaration Terendah Adalah :
Merk              : Toyota
Tipe              : MAE86 Trueno
Tahun             : 1986
Top Acceslation   : 3700
Top Power         : 553
------------------------------------------------------------------
Rata - Rata Top Power Dari Seluruh Mobil Di Showroom Adalah : 4683.75
******************************************************************
```