



NAMA : SALSABILA MAHDA RUNISHA

NIM : 2341720257

NO ABSEN : 25

KELAS : 1F

MATERI : BRUTE FORCE DAN DIVIDE CONQUER

LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

2. 1. Percobaan 1: Menghitung Nilai Faktorial dengan Algoritma Brute Force dan Divide Conquer

```
package PS.Faktorial;

public class Faktorial25 {

    public int nilai;

    int faktorialBF(int n) {
        int fakto = 1;
        for(int i=1; i<=n; i++) {
            // fakto = fakto * i;
            fakto = fakto * i; // perbaikan
        }
        return fakto;
    }

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

package PS.Faktorial;

import java.util.Scanner;

public class Mainfaktorial {

    Run|Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println(x:"-----");
        System.out.println(x:"Masukkan jumlah elemen: ");
        int iJml = sc.nextInt();

        Faktorial25[] fk = new Faktorial25[iJml];
        for(int i=0; i<iJml; i++){
            fk[i] = new Faktorial25();
            System.out.println("masukkan nilai data ke-" + (i+1) + ":");
            int iNilai = sc.nextInt();
            fk[i].nilai = iNilai; //perbaikan
        }

        System.out.println(x:"HASIL - BRUTE FORCE");
        for(int i = 0; i<iJml; i++){
            System.out.println
                ("Hasil perhitungan faktorial menggunakan Brute Force adalah "
                 + fk[i].faktorialBF(fk[i].nilai));
        }

        System.out.println(x:"HASIL - DIVIDE CONQUER");
        for(int i=0; i<iJml; i++){
            System.out.println
                ("Hasil perhitungan faktorial menggunakan Divide and Conquer adalah "
                 + fk[i].faktorialDC(fk[i].nilai));
        }
    }
}
```



NAMA : SALSABILA MAHDA RUNISHA
NIM : 2341720257
NO ABSEN : 25
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

2.1.2 Verifikasi Hasil Percobaan

```
PS D:\PrakASD_1F_25> java -cp .\Program Files\Java\jdk-11.0.2\bin\java.exe -Djava.class.path=.\PrakASD_1F_25\src\PrakASD_1F_25.jar hda\AppData\Roaming\Code\User\workspaceStorage\79c876eb0fa3185450a0db594
-----
Masukkan jumlah elemen:
3
masukkan nilai data ke-1:
5
masukkan nilai data ke-2:
8
masukkan 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 Divide and Conquer adalah 120
Hasil perhitungan faktorial menggunakan Divide and Conquer adalah 40320
Hasil perhitungan faktorial menggunakan Divide and Conquer adalah 6
PS D:\PrakASD_1F_25>
```

Question :

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

Answer :

Pernyataan if menyatakan ketika $n = 1$, maka hasil yang ditampilkan adalah 1, dan ini berfungsi sebagai batas dari perulangan.

Dengan demikian, pernyataan if digunakan untuk menangani base case dari rekursi, sedangkan pernyataan else digunakan untuk menangani rekursi yang terjadi ketika n yang diinputkan lebih besar dari 1.

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

Answer:

Ya memungkinkan, karena jenis looping apapun bisa digunakan pada method faktorialBF(). Disini saya menggunakan perulangan while.

```
5 > Faktorial > J Faktorial25.java > ...
1 package P5.Faktorial;
2
3 public class Faktorial25 {
4
5     public int nilai;
6
7     int faktorialBF(int n) {
8         int fakto = 1;
9         int i = 1;
10        while(i<=n){
11            fakto = fakto*i;
12            i++;
13        }
14        return fakto;
15    }
}

-----
Masukkan jumlah elemen:
1
masukkan nilai data ke-1:
5
HASIL - BRUTE FORCE
Hasil perhitungan faktorial menggunakan Brute Force adalah 120
HASIL - DIVIDE CONQUER
Hasil perhitungan faktorial menggunakan Divide and Conquer adalah 120
PS D:\PrakASD_1F_25>
```



NAMA : SALSABILA MAHDA RUNISHA

NIM : 2341720257

NO ABSEN : 25

KELAS : 1F

MATERI : BRUTE FORCE DAN DIVIDE CONQUER

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

Answer :

Fakto *= i; yaitu operasi yang digunakan untuk menghasilkan hasil perkalian dalam loop, sedangkan **int fakto = n * faktorialDC(n-1)**; yaitu pernyataan yang digunakan dalam rekursi untuk menghitung factorial dengan memanfaatkan submasalah yang lebih kecil untuk mendapatkan solusi dari masalah yang lebih besar.

2.2 Percobaan 2 : Menghitung Hasil Pangkat dengan Algoritma Brute Force dan Divide Conquer

```
P5 > Pangkat > J Pangkat25.java > Pangkat25 > pangkatDC(int, int)
1 package P5.Pangkat;
2
3 public class Pangkat25 {
4     public int nilai, pangkat;
5
6     int pangkatBF(int a, int n){
7         //int hasil = 0;
8         int hasil = 1; //perbaikan
9         for(int i=0; i<n; i++){
10             hasil *= a;
11         }
12         return hasil;
13     }
14
15     int pangkatDC(int a, int n){
16         if(n==1){
17             // return 1;
18             return a; //perbaikan
19         } else{
20             if(n%2==1) // bilangan ganjil
21             {
22                 return (pangkatDC(a,n/2)*pangkatDC(a,n/2)*a);
23             } else{
24                 return (pangkatDC(a,n/2)*pangkatDC(a,n/2));
25             }
26         }
27     }
28 }

P5 > Pangkat > J MainPangkat.java > MainPangkat > main(String[])
1 package P5.Pangkat;
2
3 import java.util.Scanner;
4
5 public class MainPangkat {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9         System.out.println(x:"=====");
10        System.out.println(x:"Masukkan jumlah elemen yang dihitung: ");
11        int elemen = sc.nextInt();
12
13        Pangkat25[] png = new Pangkat25[elemen];
14        for(int i=0; i<elemen; i++){
15            png[i] = new Pangkat25();
16            System.out.println(x:"Masukkan nilai yang hendak dipangkatkan: ");
17            // int nilai = sc.nextInt();
18            png[i].nilai = sc.nextInt(); //perbaikan
19            System.out.println(x:"Masukkan nilai pemangkat:");
20            //int pangkat = sc.nextInt();
21            png[i].pangkat = sc.nextInt(); //perbaikan
22        }
23
24        System.out.println(x:"HASIL PANGKAT - BRUTE FORCE");
25        for(int i=0; i<elemen; i++){
26            System.out.println
27            ("Hasil dari"
28             + png[i].nilai+ " pangkat "
29             + png[i].pangkat+ " adalah "
30             + png[i].pangkatBF(png[i].nilai, png[i].pangkat));
31        }
32
33        System.out.println(x:"HASIL PANGKAT - DIVIDE AND CONQUER");
34        for(int i=0; i<elemen; i++){
35            System.out.println
36            ("Hasil dari"
37             + png[i].nilai+ " pangkat "
38             + png[i].pangkat+ " adalah "
39             + png[i].pangkatDC(png[i].nilai, png[i].pangkat));
40        }
41    }
42 }
43 }
```



NAMA : SALSABILA MAHDA RUNISHA
NIM : 2341720257
NO ABSEN : 25
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

2.2.2 Verifikasi Hasil Percobaan

```
=====
Masukkan jumlah elemen yang dihitung:
2
Masukkan nilai yang hendak dipangkatkan:
6
Masukkan nilai pemangkat:
2
Masukkan nilai yang hendak dipangkatkan:
4
Masukkan nilai pemangkat:
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 D:\PrakASD_1F_25>
```

Question :

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

Answer:

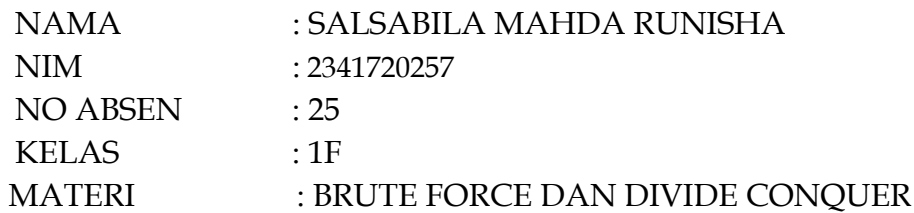
Pada method pangkatBF() operasi menghitung pangkat dilakukan dengan cara brute force yang dilakukan dengan looping dan algoritma brute force nya mendeklarasikan dulu hasil = 1 lalu melakukan looping dengan batas n (pangkatnya) dan dalam perulangan tersebut dilakukan looping dari hasil tadi di kali dengan a (bilangan yang akan dipangkat) dan perulangan akan terus berlanjut hingga $\leq n$ sehingga a akan menghasilkan nilai hasil dari pemangkatnya. Sedangkan pada method pangkatDC() operasi menghitung pangkat dilakukan dengan cara Divide and Conquer yang dilakukan dengan rekursif dan algoritma Divide and Conquer yang dilakukan terbagi dalam 3 tahap, yaitu : Divide, Conquer, Combine.

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

Answer :

Ya sudah termasuk, tahap combine dalam kode program pangkat ditunjukkan pada sintaks return atau pengembalian nilai dimana hasil penyelesaian masalah sebelumnya direturn semua dan dalam tahap combine dilakukan pemanggilan hasil dari bilangan berpangkat tersebut.

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



- Answer :**

```
package P5.Pangkat;

public class MainPangkat {
    Run | Debug
    public static void main(String[] args) {}

    Pangkat0 [] png = new Pangkat0[3];
    png[0] = new Pangkat0(nilai:5, pangkat:2);
    png[1] = new Pangkat0(nilai:10, pangkat:2);
    png[2] = new Pangkat0(nilai:6, pangkat:2);

    System.out.println("-----");
    System.out.println("Hasil pangkat dengan Brute Force ");
    for(int i=0; i<png.length; i++){
        System.out.println("Nilai "+png[i].nilai+ " pangkat "+png[i].pangkat+ " adalah "+png[i].pangkatBF(png[i].nilai, png[i].pangkat));
    }
    System.out.println("-----");
    System.out.println("Hasil Pangkat Divide and Conquer ");
    for(int i=0; i<png.length; i++){
        System.out.println("Nilai "+png[i].nilai+ " pangkat "+png[i].pangkat+ " adalah "+png[i].pangkatDC(png[i].nilai, png[i].pangkat));
    }
}
```

```
PS D:\PrakASD_1F_25> & 'C:\Program Files\Java\jdk-8.0.60\bin\java.exe' -Xmx1024m -Xms1024m -jar a\jdt_ws\PrakASD_1F_25_3f0e4e3b\bin\PrakASD_1F_25_3f0e4e3b.jar
=====
Hasil pangkat dengan Brute Force
Nilai 5 pangkat 2 adalah 25
Nilai 10 pangkat 2 adalah 100
Nilai 6 pangkat 2 adalah 36
=====
Hasil Pangkat Divide and Conquer
Nilai 5 pangkat 2 adalah 25
Nilai 10 pangkat 2 adalah 100
Nilai 6 pangkat 2 adalah 36
PS D:\PrakASD_1F_25>
```

- Answer :**



NAMA : SALSABILA MAHDA RUNISHA
NIM : 2341720257
NO ABSEN : 25
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

```
PS > Pangkat > J MainPangkat.java > MainPangkat > main(String[] args)
1 package PS.Pangkat;
2
3 import java.util.Scanner;
4
5 public class MainPangkat {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        Pangkat25[] png = new Pangkat25[3];
11        png[0] = new Pangkat25(nilai:5, pangkat:2);
12        png[1] = new Pangkat25(nilai:10, pangkat:2);
13        png[2] = new Pangkat25(nilai:6, pangkat:2);
14
15        int pilih;
16        do {
17            System.out.println("Menu Perhitungan Pangkat");
18            System.out.println("=====");
19            System.out.println("1. Menggunakan Brute Force ");
20            System.out.println("2. Menggunakan Divide and Conquer ");
21            System.out.println("3. Keluar ");
22            System.out.print("Masukkan Pilihan Anda : ");
23            pilih = sc.nextInt();
24
25            switch(pilih) {
26                case 1 :
27                    System.out.println("=====");
28                    System.out.println("Hasil pangkat dengan Brute force ");
29                    for(int i = 0; i < png.length; i++){
30                        System.out.println("Nilai " + png[i].nilai + " pangkat " + png[i].pangkat + " adalah " + png[i].pangkatBf(png[i].nilai, png[i].pangkat));
31                    }
32                    System.out.println("=====");
33                    break;
34                case 2 :
35                    System.out.println("=====");
36                    System.out.println("Hasil pangkat dengan Devide Conquer ");
37                    for(int i = 0; i < png.length; i++){
38                        System.out.println("Nilai " + png[i].nilai + " pangkat " + png[i].pangkat + " adalah " + png[i].pangkatDC(png[i].nilai, png[i].pangkat));
39                    }
40                    System.out.println("=====");
41                    break;
42                case 3 :
43                    break; // Keluar dari loop
44                default :
45                    System.out.println("Input yang anda masukkan tidak valid");
46            }
47        } while (pilih != 3);
48    }
}
```

```
PS D:\PrakASD_1F_25> & 'C:\Program Files\Java\jdk-11.0.2\bin\java.exe' -cp 'PS.Pangkat' MainPangkat
Menu Perhitungan Pangkat
=====
1. Menggunakan Brute Force
2. Menggunakan Divide and Conquer
3. Keluar
Masukkan Pilihan Anda : 1
=====
Hasil pangkat dengan Brute force
Nilai 5 pangkat 2 adalah 25
Nilai 10 pangkat 2 adalah 100
Nilai 6 pangkat 2 adalah 36
=====
Menu Perhitungan Pangkat
=====
1. Menggunakan Brute Force
2. Menggunakan Divide and Conquer
3. Keluar
Masukkan Pilihan Anda : 2
=====
Hasil pangkat dengan Devide Conquer
Nilai 5 pangkat 2 adalah 25
Nilai 10 pangkat 2 adalah 100
Nilai 6 pangkat 2 adalah 36
=====
Menu Perhitungan Pangkat
=====
1. Menggunakan Brute Force
2. Menggunakan Divide and Conquer
3. Keluar
Masukkan Pilihan Anda : 3
PS D:\PrakASD_1F_25>
```



NAMA : SALSABILA MAHDA RUNISHA

NIM : 2341720257

NO ABSEN : 25

KELAS : 1F

MATERI : BRUTE FORCE DAN DIVIDE CONQUER

2.3 Percobaan 3 : Mengitung Sum Array dengan Algoritma Brute Force dan Divide Conquer

```
1 package PS.SumArray;
2
3 public class Sum25 {
4     int elemen;
5     double keuntungan[], total;
6
7     Sum25(int elemen){
8         this.elemen = elemen;
9         this.keuntungan = new double[elemen];
10        this.total = 0;
11    }
12    double totalBF(double arr[]){
13        for(int i=0; i<elemen; i++){
14            total = total + arr[i];
15        }
16        return total;
17    }
18    double totalDC(double arr[], int l, int r){
19        if(l == r) {
20            return arr[l];
21        } else if(l < r) {
22            int mid = (l + r) / 2;
23            // double lsum = totalDC(arr, l, mid-1);
24            double lsum = totalDC(arr, l, mid-1); //perbaikan
25            //double rsum = totalDC(arr, mid+1, r);
26            double rsum = totalDC(arr, mid + 1, r); //perbaikan
27            return lsum + rsum+arr[mid];
28        }
29        return 0;
30    }
31 }
32 }
```

```
1 package PS.SumArray;
2
3 import java.util.Scanner;
4
5 public class MainSum {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         System.out.println("=====");
9         System.out.println("Program Menghitung Keuntungan Total (Satuan Juta. Misal 5.9)");
10        System.out.print("Masukkan jumlah bulan : ");
11        int elm = sc.nextInt();
12
13        Sum25 sm = new Sum25(elm);
14        System.out.println("=====");
15        for(int i=0; i<sm.elemen; i++){
16            System.out.print("Masukkan untung bulan ke -"+(i+1)+" = ");
17            sm.keuntungan[i] = sc.nextDouble();
18        }
19        System.out.println("=====");
20        System.out.println("Algoritma Brute Force");
21        System.out.println("Total keuntungan perusahaan selama "+ sm.elemen+ " bulan adalah = "+sm.totalBF(sm.keuntungan));
22        System.out.println("=====");
23        System.out.println("Algoritma Divide Conquer");
24        System.out.println("Total keuntungan perusahaan selama "+ sm.elemen+ " bulan adalah = "+sm.totalDC(sm.keuntungan, 0, sm.elemen-1));
25    }
26 }
```

2.3.2 Verifikasi Hasil Percobaan

```
=====
Program Menghitung Keuntungan Total (Satuan Juta. Misal 5.9
Masukkan jumlah bulan : 5
=====
Masukkan untung bulan ke -1 = 8.5
Masukkan untung bulan ke -2 = 9.54
Masukkan untung bulan ke -3 = 7.2
Masukkan untung bulan ke -4 = 9.1
Masukkan untung bulan ke -5 = 6
=====
Algoritma Brute Force
Total keuntungan perusahaan selama 5 bulan adalah = 40.339999999999996
=====
Algoritma Divide Conquer
Total keuntungan perusahaan selama 5 bulan adalah = 40.34
PS D:\PrakASD 1F_25> d:; cd 'd:\PrakASD 1F_25'; & 'C:\Program Files\J
```

Question :

1. Mengapa terdapat formulasi return value berikut? Jelaskan!

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



NAMA : SALSABILA MAHDA RUNISHA
NIM : 2341720257
NO ABSEN : 25
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

Answer :

Karena untuk mereturnkan semua hasil dari penjumlahan keuntungan dari lsum, rsum, dan arr[mid], sehingga dari penjumlahan ketiganya bisa ditemukan berapa banyak keuntungan yang didapat dari Perusahaan tersebut dalam rentang waktu n bulan.

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

Answer :

Karena perhitungan keuntungan pada method totalDC() tersebut membagi bagiannya menjadi left dan right. Maka dari itu membutuhkan variable mid untuk perhitungan semua bagian tengahnya.

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!

Answer :

```
PS > SumArray > Sum25.java > Sum25 > totalBF(double[])  
1 package ...sumarray;  
2  
3 public class Sum25 {  
4     int elemen;  
5     double keuntungan[], total;  
6  
7     // Sum25(int elemen){  
8         // this.elemen = elemen;  
9         // this.keuntungan = new double[elemen];  
10        // this.total = 0;  
11    }  
12  
13    public double totalBF(double arr[]){  
14        double total = 0;  
15        for(int i=0; i<arr.length; i++){  
16            total = total + arr[i];  
17        }  
18        return total;  
19    }  
20    double totalDC(double arr[], int l, int r){  
21        if(l == r){  
22            return arr[l];  
23        } else if(l < r){  
24            int mid = (l + r) / 2;  
25            // double lsum = totalDC(arr, l, mid-1);  
26            double lsum = totalDC(arr, l, mid-1); //perbaikan  
27            //double rsum = totalDC(arr, mid+1, r);  
28            double rsum = totalDC(arr, mid + 1, r); //perbaikan  
29            return lsum + rsum+arr[mid];  
30        }  
31        return 0;  
32    }  
33  
34 }
```

```
SumArray > MainProgram.java > MainProgram > MainProgram  
1 package ...sumarray;  
2  
3 import java.util.Scanner;  
4  
5 public class MainProgram {  
6     Run/Debug  
7     public static void main(String[] args) {  
8         Scanner sc = new Scanner(System.in);  
9  
10        System.out.println("=====");  
11        System.out.println("Program Menghitung Keuntungan Total (Satuan Juta. Misal 5,9)");  
12        System.out.println("Masukkan jumlah perusahaan : ");  
13        int per = sc.nextInt();  
14  
15        int[] peru = new int[per];  
16        for(int i=0; i<per; i++){  
17            peru[i] = new Sum25();  
18            System.out.println("-----");  
19            System.out.println("Perusahaan ke : "+(i+1));  
20            System.out.println("Masukkan jumlah bulan : ");  
21            peru[i].elemen = sc.nextInt();  
22  
23            peru[i].keuntungan = new double[peru[i].elemen];  
24  
25            System.out.println("-----");  
26            for(int j=0; j<peru[i].elemen; j++){  
27                System.out.print("Masukkan untung bulan ke : "+(j+1)+" = ");  
28                peru[i].keuntungan[j] = sc.nextDouble();  
29            }  
30  
31            System.out.println("-----");  
32            Sum25 dat = new Sum25();  
33            for(int i=0; i<peru.length; i++){  
34                System.out.println("Perusahaan ke : "+(i+1));  
35                System.out.println("Jumlah Bulan : "+peru[i].elemen);  
36                System.out.println();  
37                System.out.println("Algoritma Brute Force");  
38                System.out.println("Total keuntungan perusahaan selama "+peru[i].elemen+" bulan adalah : %.2f ", dat.totalBF(peru[i].keuntungan));  
39                System.out.println();  
40                System.out.println("Algoritma Divide Conquer");  
41                System.out.println("Total keuntungan perusahaan selama "+peru[i].elemen+" bulan adalah : %.2f ", dat.totalDC(peru[i].keuntungan, 1,0, peru[i].elemen-1));  
42                System.out.println("=====");  
43            }  
44        }  
45    }  
46 }
```




NAMA : SALSABILA MAHDA RUNISHA
NIM : 2341720257
NO ABSEN : 25
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

```
PS D:\PrakASD_1F_25> & 'C:\Program Files\Java\jdk-20\bin\java  
a\jdt_ws\PrakASD_1F_25_3f0e4e3b\bin' 'P5.SumArray.MainSum'  
=====
Program Menghitung Keuntungan Total (Satuan Juta. Misal 5,9)
Masukkan jumlah perusahaan : 3
=====
Perusahaan ke -1
Masukkan Jumlah Bulan : 2
=====
Masukkan untung bulan ke -1 = 10
Masukkan untung bulan ke -2 = 15
=====
Perusahaan ke -2
Masukkan Jumlah Bulan : 1
=====
Masukkan untung bulan ke -1 = 9.5
=====
Perusahaan ke -3
Masukkan Jumlah Bulan : 2
=====
Masukkan untung bulan ke -1 = 9.8
Masukkan untung bulan ke -2 = 20
=====
Perusahaan ke-1
Jumlah Bulan : 2

Algoritma Brute Force
Total keuntungan perusahaan selama 2 bulan adalah : 25.00
Algoritma Divide Conquer
Total keuntungan perusahaan selama 2 bulan adalah : 25.00
=====
Perusahaan ke-2
Jumlah Bulan : 1

Algoritma Brute Force
Total keuntungan perusahaan selama 1 bulan adalah : 9.50
Algoritma Divide Conquer
Total keuntungan perusahaan selama 1 bulan adalah : 9.50
=====
Perusahaan ke-3
Jumlah Bulan : 2

Algoritma Brute Force
Total keuntungan perusahaan selama 2 bulan adalah : 29.80
Algoritma Divide Conquer
Total keuntungan perusahaan selama 2 bulan adalah : 29.80
=====
PS D:\PrakASD_1F_25> |
```

2.4 Latihan Praktikum

1. sebuah showroom memiliki daftar mobil dengan data sesuai table di bawah ini

merk	tipe	tahun	top_acceleration	top_power
BMW	M2 Coupe	2016	6816	728
Ford	Fiesta ST	2014	3921	575
Nissan	370Z	2009	4360	657
Subaru	BRZ	2014	4058	609
Subaru	Impreza WRX STI	2013	6255	703
Toyota	AE86 Trueno	1986	3700	553
Toyota	86/GT86	2014	4180	609
Volkswagen	Golf GTI	2014	4180	631

Tentukan:

- a) top_acceleration tertinggi menggunakan Divide and Conquer!
- b) top_acceleration terendah menggunakan Divide and Conquer!
- c) Rata-rata top_power dari seluruh mobil menggunakan Brute Force!



NAMA : SALSABILA MAHDA RUNISHA
NIM : 2341720257
NO ABSEN : 25
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

Jawab :

```
package PS.Latihan4;

public class Mobil125 {
    String merk, tipe;
    int tahun;
    double top_acceleration, top_power;

    public Mobil125(String merk, String tipe, int tahun, double top_acceleration, double top_power) {
        this.merk = merk;
        this.tipe = tipe;
        this.tahun = tahun;
        this.top_acceleration = top_acceleration;
        this.top_power = top_power;
    }

    public static Mobil125 cariTopAccelerationTertinggi(Mobil125[] mobilis, int rendah, int tinggi) {
        if (rendah == tinggi) {
            return mobilis[rendah];
        }
        int mid = (rendah + tinggi) / 2;

        Mobil125 leftMax = cariTopAccelerationTertinggi(mobilis, rendah, mid);
        Mobil125 rightMax = cariTopAccelerationTertinggi(mobilis, mid + 1, tinggi);

        if (leftMax.top_acceleration > rightMax.top_acceleration) {
            return leftMax;
        } else {
            return rightMax;
        }
    }

    public static Mobil125 cariTopAccelerationTerendah(Mobil125[] mobilis, int rendah, int tinggi) {
        if (rendah == tinggi) {
            return mobilis[rendah];
        }
        int mid = (rendah + tinggi) / 2;

        Mobil125 leftMin = cariTopAccelerationTerendah(mobilis, rendah, mid);
        Mobil125 rightMin = cariTopAccelerationTerendah(mobilis, mid + 1, tinggi);

        if (leftMin.top_acceleration < rightMin.top_acceleration) {
            return leftMin;
        } else {
            return rightMin;
        }
    }

    public static double cariRataTopPower(Mobil125[] mobilis) {
        double totalPower = 0;
        for (Mobil125 mobil : mobilis) {
            totalPower += mobil.top_power;
        }
        return totalPower / mobilis.length;
    }
}
```

```
package PS.Latihan4;

public class MainMobil {
    public static void main(String[] args) {

        Mobil125 mb1 = new Mobil125(merk:"BMW",tipe:"M2 Coupe",tahun:2016, top_acceleration:6816, top_power:728);
        Mobil125 mb2 = new Mobil125(merk:"Ford", tipe:"Fiesta ST", tahun:2014, top_acceleration:3921, top_power:575);
        Mobil125 mb3 = new Mobil125(merk:"Nissan", tipe:"370Z", tahun:2009, top_acceleration:4360, top_power:657);
        Mobil125 mb4 = new Mobil125(merk:"Subaru", tipe:"BRZ", tahun:2014, top_acceleration:4058, top_power:609);
        Mobil125 mb5 = new Mobil125(merk:"Subaru", tipe:"Impreza WRX STI", tahun:2013, top_acceleration:6255, top_power:703);
        Mobil125 mb6 = new Mobil125(merk:"Toyota", tipe:"AE86 Trueno", tahun:1986, top_acceleration:3700, top_power:553);
        Mobil125 mb7 = new Mobil125(merk:"Toyota", tipe:"86/GT86", tahun:2014, top_acceleration:4180, top_power:609);
        Mobil125 mb8 = new Mobil125(merk:"Volkswagen", tipe:"Golf GTI", tahun:2014, top_acceleration:4180, top_power:631);

        Mobil125[] mobilis = new Mobil125[8];

        mobilis[0] = mb1;
        mobilis[1] = mb2;
        mobilis[2] = mb3;
        mobilis[3] = mb4;
        mobilis[4] = mb5;
        mobilis[5] = mb6;
        mobilis[6] = mb7;
        mobilis[7] = mb8;

        System.out.println(x:"=====");
        Mobil125 accTertinggi = Mobil125.cariTopAccelerationTertinggi(mobilis, rendah:0, mobilis.length - 1);
        System.out.println(x:"Mobil dengan top_acceleration tertinggi: ");
        System.out.println("Merk : " + accTertinggi.merk);
        System.out.println("Tipe : " + accTertinggi.tipe);
        System.out.println("Tahun : " + accTertinggi.tahun);
        System.out.println(x:"=====");

        Mobil125 accTerendah = Mobil125.cariTopAccelerationTerendah(mobilis, rendah:0, mobilis.length - 1);
        System.out.println(x:"Mobil dengan top_acceleration terendah: ");
        System.out.println("Merk : " + accTerendah.merk);
        System.out.println("Tipe : " + accTerendah.tipe);
        System.out.println("Tahun : " + accTerendah.tahun);
        System.out.println(x:"=====");

        double rataTopPower = Mobil125.cariRataTopPower(mobilis);
        System.out.println("Rata - rata top_power seluruh mobil : " + rataTopPower);
        System.out.println(x:"=====");
    }
}
```



NAMA : SALSABILA MAHDA RUNISHA
NIM : 2341720257
NO ABSEN : 25
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

```
Latihan.MainMobil'  
=====  
Mobil dengan top_acceleration tertinggi:  
Merk : BMW  
Tipe : M2 Coupe  
Tahun : 2016  
=====  
Mobil dengan top_acceleration terendah:  
Merk : Toyota  
Tipe : AE86 Trueno  
Tahun : 1986  
=====  
Rata - rata top_power seluruh mobil : 633.125  
=====  
PS D:\PrakASD_1F_25>
```

Bukti perhitungan rata - rata sudah valid :

E10 : ✕ ✓ <i>fx</i> =AVERAGE(E2:E9)					
	A	B	C	D	E
1	merk	tipe	tahun	top_acceleration	top_power
2	BMW	M2 Coupe	2016	6816	728
3	Ford	Fiesta ST	2014	3921	575
4	Nissan	370z	2009	4360	657
5	Subaru	BRZ	2014	2058	609
6	Subaru	Impreza WRX STI	2013	6255	703
7	Toyota	AE86 Trueno	1986	3700	553
8	Toyota	86/GT86	2014	4180	609
9	Volkswagen	Golf GTI	2014	4180	631
10					633.125
11					