



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

## LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

\*FILE NAME =ABSEN\_NAME \_CLASS\_P1\*

\* Pertemuan mengikuti pertemuan ke berapa

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

\* screenshot code program hasil percobaan\*

Jika percobaan membutuhkan lebih dari 1 class java, maka screenshot dilakukan per class / per method

```
J MainFaktorial26.java 1, U, U PrakASD_1F_26\src\PS\MainFaktorial26.java\...
1  package P5;
2
3  import java.util.Scanner;
4
5  public class MainFaktorial26 {
6      Run | Debug
7      public static void main(String[] args) {
8          Scanner sc26 = new Scanner(System.in);
9          System.out.println(x: "=====");
10         System.out.print(s: "Masukan Jumlah Elemen : ");
11         int ijml = sc26.nextInt();
12
13         Faktorial26[] fk = new Faktorial26[ijml];
14         for (int i = 0; i < ijml; i++) {
15             fk[i] = new Faktorial26();
16             System.out.print("Masukkan Nilai Data Ke-" + (i + 1) + " : ");
17             fk[i].nilai = sc26.nextInt();
18             System.out.println(x: "HASIL - BRUTE FORCE ");
19         }
20         for (int i = 0; i < ijml; i++) {
21             System.out.println(
22                 "Hasil penghitungan faktorial menggunakan Brute Force adalah " + fk[i].faktorialBF(fk[i].nilai));
23         }
24         System.out.println(x: "HASIL - DEVIDE AND CONQUER");
25         for (int i = 0; i < ijml; i++) {
26             System.out.println(
27                 "Hasil penghitungan faktorial menggunakan Devidе Conquer adalah " + fk[i].faktorialDC(fk[i].nilai));
28         }
29     }
30 }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

```
J Faktorial26.java U, U PrakASD_1F_26\src\P5\Faktorial26.java\...
1 package P5;
2
3 public class Faktorial26 {
4     int nilai;
5
6     int faktorialBF(int n) {
7         int fakto = 1;
8         for (int i = 1; i <= n; i++) {
9             fakto = fakto * i;
10        }
11        return fakto;
12    }
13
14    int faktorialDC(int n) {
15        if (n == 1) {
16            return 1;
17        } else {
18            int fakto = n * faktorialDC(n - 1);
19            return fakto;
20        }
21    }
22 }
```

```
=====
Masukan Jumlah Elemen : 3
Masukkan Nilai Data Ke-1: 5
HASIL - BRUTE FORCE
Masukkan Nilai Data Ke-2: 8
HASIL - BRUTE FORCE
Masukkan Nilai Data Ke-3: 3
HASIL - BRUTE FORCE
Hasil penghitungan faktorial menggunakan Brute Force adalah 120
Hasil penghitungan faktorial menggunakan Brute Force adalah 40320
Hasil penghitungan faktorial menggunakan Brute Force adalah 6
HASIL - DEVIDE AND CONQUER
Hasil penghitungan faktorial menggunakan Devide Conquer adalah 120
Hasil penghitungan faktorial menggunakan Devide Conquer adalah 40320
Hasil penghitungan faktorial menggunakan Devide Conquer adalah 6
PS D:\Praktikum Algoritma dan Struktur Data> □
```

### Question :

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

Jawab : Pada bagian **if** digunakan untuk mengecek apakah parameter n sama dengan 1 ( $n == 1$ ). Jika sama (terpenuhi  $n = 1$ ), maka akan langsung mengembalikan nilai 1.

Pada bagian **else** digunakan jika n bukan 1. maka akan dilakukan rekursif dengan mengalikan n dengan pemanggilan fungsi faktorialDC( $n - 1$ ).

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

Jawab : Ya Bisa , Dengan cara menggunakan perulangan while.



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

Hasil Modifikasi :

```
J Faktorial26.java U, M PrakASD_1F_26\src\P5\Faktorial26.java\...
1 package P5;
2
3 public class Faktorial26 {
4     int nilai;
5
6     int faktorialBF(int n) {
7         int fakto = 1;
8         int i = 1;
9         while (i <= n) {
10             fakto = fakto * i;
11             i++;
12         }
13         return fakto;
14     }
15
16     int faktorialDC(int n) {
17         if (n == 1) {
18             return 1;
19         } else {
20             int fakto = n * faktorialDC(n - 1);
21             return fakto;
22         }
23     }
24 }
```

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

Jawab : perbedaan antara  $fakto *= i$ ; dan  $int fakto = n * faktorialDC(n-1)$ ;

Pada  **$fakto *= i$**  didalam **faktorialBF**:

Iterasi dilakukan dari 1 hingga n. Pada setiap iterasi, nilai fakto akan dikalikan dengan nilai iterasi saat itu (i).

Pada  **$fakto = n *$**  didalam **faktorialDC**:

Ketika nilai n dipanggil, metode ini membagi masalah menjadi dua bagian dan menghitung faktorial dari (n-1) kemudian mengalikan hasilnya dengan n. Setelah itu berlanjut hingga mencapai kondisi dasar (basis case) yaitu  $n = 1$ , di mana hasilnya akan dikembalikan.



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

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

*\* screenshot code program hasil percobaan \**

*Jika percobaan membutuhkan lebih dari 1 class java, maka screenshot dilakukan per class / per method*

```
1 package P5.Pangkat;
2
3 public class Pangkat26 {
4     public int nilai, pangkat;
5
6     int pangkatBF(int a, int n) {
7         int hasil = 1;
8         for (int i = 0; i < n; i++) {
9             hasil *= a;
10        }
11        return hasil;
12    }
13
14    int pangkatDC(int a, int n) {
15        if (n == 0) {
16            return 1;
17        } else {
18            int hasil = pangkatDC(a, n / 2);
19            if (n % 2 == 1) {
20                return (hasil * hasil * a);
21            } else {
22                return (hasil * hasil);
23            }
24        }
25    }
26 }
```

```
1 package P5.Pangkat;
2
3 import java.util.Scanner;
4
5 public class MainPangkat26 {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc26 = new Scanner(System.in);
9         System.out.println(x:"=====");
10        System.out.print(s:"Masukan Jumlah Elemen yang dihitung : ");
11        int elemen = sc26.nextInt();
12
13        Pangkat26[] png = new Pangkat26[elemen];
14        for (int i = 0; i < elemen; i++) {
15            png[i] = new Pangkat26();
16            System.out.println(x:"Masukkan Nilai yang hendak dipangkatkan : ");
17            png[i].nilai = sc26.nextInt();
18            System.out.println(x:"Masukkan nilai pemangkat : ");
19            png[i].pangkat = sc26.nextInt();
20        }
21        System.out.println(x:"HASIL PANGKAT - BRUTE FORCE");
22        for (int i = 0; i < elemen; i++) {
23            System.out.println("Hasil dari " + png[i].nilai + " pangkat" + png[i].pangkat + " "
24                + png[i].pangkatBF(png[i].nilai, png[i].pangkat));
25        }
26        System.out.println(x:"HASIL PANGKAT - DEVIDE AND CONQUER");
27        for (int i = 0; i < elemen; i++) {
28            System.out.println("Hasil dari " + png[i].nilai + " pangkat" + png[i].pangkat + " adalah "
29                + png[i].pangkatDC(png[i].nilai, png[i].pangkat));
30        }
31    }
32 }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

```
=====
Masukan 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 36
Hasil dari 4 pangkat 3 64
HASIL PANGKAT - DEVIDE AND CONQUER
Hasil dari 6 pangkat 2 adalah 36
Hasil dari 4 pangkat 3 adalah 64
PS D:\Praktikum Algoritma dan Struktur Data>
```

### Question :

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

Jawab :

- **pangkatBF()** (Brute Force)  
Pada Methods **pangkatBF()** menggunakan metode iteratif untuk menghitung pangkatnya. Pada setiap iterasi, bilangan dasar akan dikalikan dengan dirinya sendiri sebanyak n kali (sesuai dengan pangkat yang diinginkan).
- **pangkatDC()** (Divide and Conquer)  
Pada Method **pangkatDC()** menggunakan metode rekursif yang membagi masalah menjadi submasalah yang lebih kecil. bahwa a pangkat n dapat dipecah menjadi  $(a^{(n/2)}) * (a^{(n/2)})$ , sehingga dapat mengurangi jumlah perkaliannya. Dan Jika pangkat n ganjil maka akan mengalikan hasil dua pangkat  $(a^{(n/2)})$  dengan a tambahan untuk menangani pangkat ganjilnya.

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

Jawab : Ya, Menurut saya Tahap "combine" sudah terjadi pada bagian methods **pangkatDC(int a, int n)** yang ada pada kelas **Pangkat26**, karena terdapat penggabungan hasil submasalah.

```
18         int hasil = pangkatDC(a, n / 2);
19         if (n % 2 == 1) {
20             return (hasil * hasil * a);
21         } else {
22             return (hasil * hasil);
23         }
24     }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

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

Jawab :

```
1 package P5.Pangkat;  
2  
3 public class Pangkat26 {  
4     public int nilai, pangkat;  
5  
6     public Pangkat26(int nilai, int pangkat) {  
7         this.nilai = nilai;  
8         this.pangkat = pangkat;  
9     }  
10  
11     int pangkatBF(int a, int n) {  
12         int hasil = 1;  
13         for (int i = 0; i < n; i++) {  
14             hasil *= a;  
15         }  
16         return hasil;  
17     }  
18  
19     int pangkatDC(int a, int n) {  
20         if (n == 0) {  
21             return 1;  
22         } else {  
23             int hasil = pangkatDC(a, n / 2);  
24             if (n % 2 == 1) {  
25                 return (hasil * hasil * a);  
26             } else {  
27                 return (hasil * hasil);  
28             }  
29         }  
30     }  
31 }
```

```
1 package P5.Pangkat;  
2  
3 import java.util.Scanner;  
4  
5 public class MainPangkat26 {  
6     Run | Debug  
7     public static void main(String[] args) {  
8         Scanner sc26 = new Scanner(System.in);  
9         System.out.println(x: "=====");  
10        System.out.print(s: "Masukan Jumlah Elemen yang dihitung : ");  
11        int elemen = sc26.nextInt();  
12  
13        Pangkat26[] png = new Pangkat26[elemen];  
14        for (int i = 0; i < elemen; i++) {  
15            System.out.println(x: "Masukkan Nilai yang hendak dipangkatkan : ");  
16            int nilai = sc26.nextInt();  
17            System.out.println(x: "Masukkan nilai pemangkat : ");  
18            int pangkat = sc26.nextInt();  
19            png[i] = new Pangkat26(nilai, pangkat);  
20        }  
21        System.out.println(x: "HASIL PANGKAT - BRUTE FORCE");  
22        for (int i = 0; i < elemen; i++) {  
23            System.out.println("Hasil dari " + png[i].nilai + " pangkat " + png[i].pangkat + " " +  
24                png[i].pangkatBF(png[i].nilai, png[i].pangkat));  
25        }  
26        System.out.println(x: "HASIL PANGKAT - DEVIDE AND CONQUER");  
27        for (int i = 0; i < elemen; i++) {  
28            System.out.println("Hasil dari " + png[i].nilai + " pangkat " + png[i].pangkat + " adalah " +  
29                png[i].pangkatDC(png[i].nilai, png[i].pangkat));  
30        }  
31    }  
32 }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

```
=====
Masukan 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 36
Hasil dari 4 pangkat 3 64
HASIL PANGKAT - DEVIDE AND CONQUER
Hasil dari 6 pangkat 2 adalah 36
Hasil dari 4 pangkat 3 adalah 64
PS D:\Praktikum Algoritma dan Struktur Data> █
```

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

Jawab :

```
1 package P5.Pangkat;
2
3 import java.util.Scanner;
4
5 public class MainPangkat26 {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc26 = new Scanner(System.in);
9         System.out.println(x:"=====");
10        System.out.print(s:"Masukan Jumlah Elemen yang dihitung : ");
11        int elemen = sc26.nextInt();
12
13        Pangkat26[] png = new Pangkat26[elemen];
14        for (int i = 0; i < elemen; i++) {
15            System.out.print(s:"Masukkan Nilai yang hendak dipangkatkan : ");
16            int nilai = sc26.nextInt();
17            System.out.print(s:"Masukkan nilai pemangkat : ");
18            int pangkat = sc26.nextInt();
19            png[i] = new Pangkat26(nilai, pangkat);
20        }
21        System.out.println(x:"|===== Pilih metode perhitungan pangkat =====|");
22        System.out.println(x:"| 1. BRUTE-FORCE |");
23        System.out.println(x:"| 2. DIVIDE-CONQUER |");
24        System.out.println(x:"|=====|");
25        System.out.print(s:" Silahkan Masukkan pilihan Anda : ");
26        int pilihan = sc26.nextInt();
27    }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

```
29         switch (pilihan) {  
30             case 1:  
31                 System.out.println(x:"HASIL PANGKAT - BRUTE FORCE");  
32                 for (int i = 0; i < elemen; i++) {  
33                     System.out.println("Hasil dari " + png[i].nilai + " pangkat " + png[i].pangkat + " " +  
34                         + png[i].pangkatBF(png[i].nilai, png[i].pangkat));  
35                 }  
36                 break;  
37             case 2:  
38                 System.out.println(x:"HASIL PANGKAT - DIVIDE AND CONQUER");  
39                 for (int i = 0; i < elemen; i++) {  
40                     System.out.println("Hasil dari " + png[i].nilai + " pangkat " + png[i].pangkat + " adalah " +  
41                         + png[i].pangkatDC(png[i].nilai, png[i].pangkat));  
42                 }  
43                 break;  
44             default:  
45                 System.out.println(x:"Pilihan tidak valid!");  
46         }  
47     }  
48 }
```

```
=====
Masukan 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
|===== Pilih metode perhitungan pangkat =====|
| 1. BRUTE-FORCE                             |
| 2. DIVIDE-CONQUER                          |
|=====|
Silahkan Masukkan pilihan Anda : 1
HASIL PANGKAT - BRUTE FORCE
Hasil dari 6 pangkat 2 36
Hasil dari 4 pangkat 3 64
PS D:\Praktikum Algoritma dan Struktur Data> d:; cd 'd:\Praktikum Algor
C:\Users\HP VICTUS\AppData\Roaming\Code\User\workspaceStorage\9692d2458c
=====
Masukan 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
|===== Pilih metode perhitungan pangkat =====|
| 1. BRUTE-FORCE                             |
| 2. DIVIDE-CONQUER                          |
|=====|
Silahkan Masukkan pilihan Anda : 2
HASIL PANGKAT - DIVIDE AND CONQUER
Hasil dari 6 pangkat 2 adalah 36
Hasil dari 4 pangkat 3 adalah 64
PS D:\Praktikum Algoritma dan Struktur Data> |
```





NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

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

*\* screenshot code program hasil percobaan\**

*Jika percobaan membutuhkan lebih dari 1 class java, maka screenshot dilakukan per class / per method*

```
1  package P5.SumArray;  
2  
3  public class sumArray26 {  
4      int elemen;  
5      double keuntungan[], total;  
6  
7      public sumArray26(int elemen) {  
8          this.elemen = elemen;  
9          this.keuntungan = new double[elemen];  
10         this.total = 0;  
11     }  
12  
13     double totalBF(double arr[]) {  
14         for (int i = 0; i < elemen; i++) {  
15             total = total + arr[i];  
16         }  
17         return total;  
18     }  
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);  
26             double rsum = totalDC(arr, mid + 1, r);  
27             return lsum + rsum;  
28         }  
29         return 0;  
30     }  
31 }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

```
1 package P5.SumArray;  
2  
3 import java.util.Scanner;  
4  
5 public class MainSum26 {  
6     Run | Debug  
7     public static void main(String[] args) {  
8         Scanner sc26 = new Scanner(System.in);  
9         System.out.println(x:"=====");  
10        System.out.println(x:"Program menghitung keuntungan total (satuan juta. Misal 5.9) ");  
11        System.out.print(s:"Masukan Jumlah Bulan : ");  
12        int elm = sc26.nextInt();  
13  
14        sumArray26 sm = new sumArray26(elm);  
15        System.out.println(x:"=====");  
16        for (int i = 0; i < sm.elemen; i++) {  
17            System.out.print("Masukkan Unrung Bulan Ke- " + (i + 1) + " : ");  
18            sm.keuntungan[i] = sc26.nextDouble();  
19        }  
20        System.out.println(x:"=====");  
21        System.out.println(x:"ALGORITMA BRUTE-FORCE");  
22        System.out.println(  
23            "Total Keuntungan perusahaan selama " + sm.elemen + " Bulan adalah = " + sm.totalBF(sm.keuntungan));  
24        System.out.println(x:"=====");  
25        System.out.println(x:"ALGORITMA DIVIDE-CONQUER");  
26        System.out.printf(format:"Total Keuntungan perusahaan selama %d Bulan adalah = %.2f\n", sm.elemen,  
27            sm.totalDC(sm.keuntungan, 1:0, sm.elemen - 1));  
28    }  
}
```

```
=====
Program menghitung keuntungan total (satuan juta. Misal 5.9)
Masukan Jumlah Bulan : 5
=====
Masukkan Unrung Bulan Ke- 1 : 8.5
Masukkan Unrung Bulan Ke- 2 : 9.54
Masukkan Unrung Bulan Ke- 3 : 7.2
Masukkan Unrung Bulan Ke- 4 : 9.1
Masukkan Unrung 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:\Praktikum Algoritma dan Struktur Data> |
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

### Question :

1. Mengapa terdapat formulasi return value berikut?Jelaskan!

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

Jawab : Pada **return lsum + rsum + arr[mid];** yang terdapat dalam methods **totalDC()** adalah implementasi dari Divide and Conquer untuk menghitung total keuntungan perusahaan selama beberapa bulan. Akan tetapi setelah saya coba jalankan Formulasi ini terdapat kesalahan dalam perhitungan total keuntungan menggunakan algoritma divide-conquer. Secara garis besar menurut saya kesalahannya terletak pada bagian penggabungan nilai bagian kiri dan kanan dari array. Karena pada Divide-Conquer seharusnya kita membagi masalah menjadi submasalah yang lebih kecil, dan menyelesaikan masing-masing submasalah dengan rekursif, kemudian menggabungkan nilai akhirnya

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

Jawab : Variabel mid yang ada dalam **totalDC()** digunakan untuk menentukan titik tengah dari array. Dalam Divide and Conquer hal tersebut berguna untuk membagi masalah menjadi submasalah yang lebih kecil dan kemudian menggabungkan solusi dari submasalah-submasalah tersebut.

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!

Jawab :



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

Class Baru **Perusahaan26()**;

```
1  package P5.SumArray;
2
3  public class Perusahaan26 {
4      int bulan;
5      double[] keuntungan;
6
7      public Perusahaan26(int bulan, double[] keuntungan) {
8          this.bulan = bulan;
9          this.keuntungan = keuntungan;
10     }
11
12     double totalBF() {
13         double total = 0;
14         for (int i = 0; i < bulan; i++) {
15             total += keuntungan[i];
16         }
17         return total;
18     }
19
20     double totalDC() {
21         return totalDC(1:0, bulan - 1);
22     }
23
24     private double totalDC(int l, int r) {
25         if (l == r) {
26             return keuntungan[l];
27         } else if (l < r) {
28             int mid = (l + r) / 2;
29             double lsum = totalDC(l, mid);
30             double rsum = totalDC(mid + 1, r);
31             return lsum + rsum;
32         }
33         return 0;
34     }
35 }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

Main:

```
1 package P5.SumArray;
2
3 import java.util.Scanner;
4
5 public class MainSum26 {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc26 = new Scanner(System.in);
9         System.out.println(x: "=====");
10        System.out.println(x: "Program menghitung keuntungan total (satuan juta. Misal 5.9) ");
11        System.out.print(s: "Masukan Jumlah Perusahaan : ");
12        int noPerusahaan = sc26.nextInt();
13
14        Perusahaan26[] perusahaan = new Perusahaan26[noPerusahaan];
15
16        for (int i = 0; i < noPerusahaan; i++) {
17            System.out.println(x: "=====");
18            System.out.println("Perusahaan " + (i + 1));
19            System.out.print(s: "Masukan Jumlah Bulan: ");
20            int bulan = sc26.nextInt();
21
22            double[] keuntungan = new double[bulan];
23
24            for (int j = 0; j < bulan; j++) {
25                System.out.print("Masukkan Keuntungan Bulan Ke-" + (j + 1) + ": ");
26                keuntungan[j] = sc26.nextDouble();
27            }
28
29            perusahaan[i] = new Perusahaan26(bulan, keuntungan);
30        }
31
32        System.out.println(x: "=====");
33        for (int i = 0; i < noPerusahaan; i++) {
34            Perusahaan26 perusahaanSkrng = perusahaan[i];
35            System.out.println("Perusahaan " + (i + 1));
36            System.out.println(x: "ALGORITMA BRUTE-FORCE");
37            System.out.println("Total Keuntungan perusahaan selama " + perusahaanSkrng.bulan + " Bulan adalah = " +
38                perusahaanSkrng.totalBF());
39            System.out.println(x: "=====");
40            System.out.println(x: "ALGORITMA DIVIDE-CONQUER");
41            System.out.printf(format: "Total Keuntungan perusahaan selama %d Bulan adalah = %.2f\n", perusahaanSkrng.bulan,
42                perusahaanSkrng.totalDC());
43        }
44    }
45 }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

Hasil dari outputnya :

```
=====
Program menghitung keuntungan total (satuan juta. Misal 5.9)
Masukan Jumlah Perusahaan : 3
=====
Perusahaan 1
Masukan Jumlah Bulan: 2
Masukkan Keuntungan Bulan Ke-1: 9.8
Masukkan Keuntungan Bulan Ke-2: 8.9
=====
Perusahaan 2
Masukan Jumlah Bulan: 3
Masukkan Keuntungan Bulan Ke-1: 7.5
Masukkan Keuntungan Bulan Ke-2: 8.5
Masukkan Keuntungan Bulan Ke-3: 9.5
=====
Perusahaan 3
Masukan Jumlah Bulan: 4
Masukkan Keuntungan Bulan Ke-1: 7.0
Masukkan Keuntungan Bulan Ke-2: 8.0
Masukkan Keuntungan Bulan Ke-3: 9.0
Masukkan Keuntungan Bulan Ke-4: 10
=====
Perusahaan 1
ALGORITMA BRUTE-FORCE
Total Keuntungan perusahaan selama 2 Bulan adalah = 18.700000000000003
=====
ALGORITMA DIVIDE-CONQUER
Total Keuntungan perusahaan selama 2 Bulan adalah = 18.70
Perusahaan 2
ALGORITMA BRUTE-FORCE
Total Keuntungan perusahaan selama 3 Bulan adalah = 25.5
=====
ALGORITMA DIVIDE-CONQUER
Total Keuntungan perusahaan selama 3 Bulan adalah = 25.50
Perusahaan 3
ALGORITMA BRUTE-FORCE
Total Keuntungan perusahaan selama 4 Bulan adalah = 34.0
=====
ALGORITMA DIVIDE-CONQUER
Total Keuntungan perusahaan selama 4 Bulan adalah = 34.00
PS D:\Praktikum Algoritma dan Struktur Data> |
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

#### 4.5 Latihan Praktikum

1. Sebuah showroom memiliki daftar mobil dengan data sesuai tabel 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:

- top\_acceleration tertinggi menggunakan Divide and Conquer!*
- top\_acceleration terendah menggunakan Divide and Conquer!*
- Rata-rata top\_power dari seluruh mobil menggunakan Brute Force!*

Jawab :

Class Mobil26();

```
J Mobil26.java U, M PrakASD_1F_26\src\P5\Latihan\Mobil26.java\...
1  package P5.Latihan;
2
3  public class Mobil26 {
4      String Merk, Tipe;
5      int Tahun, topAcceleration, topPower;
6
7      public Mobil26(String Merk, String Tipe, int Tahun, int topAcceleration, int topPower) {
8          this.Merk = Merk;
9          this.Tipe = Tipe;
10         this.Tahun = Tahun;
11         this.topAcceleration = topAcceleration;
12         this.topPower = topPower;
13     }
14 }
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

Class Showroom26();

```
J Showroom26.java U, M PrakASD_1F_26\src\P5\Latihan\Showroom26.java\...
1  package P5.Latihan;
2
3  public class Showroom26 {
4      Mobil26[] jnsMobil;
5
6      public Showroom26(Mobil26[] jnsMobil) {
7          this.jnsMobil = jnsMobil;
8      }
9
10     // Method mencari top_acceleration tertinggi menggunakan Divide and Conquer
11     public int TopAccelerationTertinggi(int start, int end) {
12         if (start == end)
13             return jnsMobil[start].topAcceleration;
14
15         int mid = (start + end) / 2;
16         int lMax = TopAccelerationTertinggi(start, mid);
17         int rMax = TopAccelerationTertinggi(mid + 1, end);
18
19         return Math.max(lMax, rMax);
20     }
21
22     // Method mencari top_acceleration terendah menggunakan Divide and Conquer!
23     public int TopAccelerationTerendah(int start, int end) {
24         if (start == end)
25             return jnsMobil[start].topAcceleration;
26
27         int mid = (start + end) / 2;
28         int lMin = TopAccelerationTerendah(start, mid);
29         int rMin = TopAccelerationTerendah(mid + 1, end);
30
31         return Math.min(lMin, rMin);
32     }
33
34     // Method mencari Rata-rata top_power seluruh mobil menggunakan Brute Force!
35     public double RataTopPower() {
36         int sum = 0;
37         for (Mobil26 car : jnsMobil) {
38             sum += car.topPower;
39         }
40         return (double) sum / jnsMobil.length;
41     }
42 }
43 }
```





NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer

Class Main26());

```
J Main26.java U,M PraksASD_1F_26(src\PS\Latihan\Main26.java\...
1 package P5.Latihan;
2
3 public class Main26 {
4     Run | Debug
5     public static void main(String[] args) {
6
7         Mobil26[] jnsMobil = {
8             new Mobil26(Merk:"BMW", Tipe:"M2 coupe", Tahun:2016, topAcceleration:6816, topPower:728),
9             new Mobil26(Merk:"Ford", Tipe:"Fiesta ST", Tahun:2014, topAcceleration:3921, topPower:575),
10            new Mobil26(Merk:"Nissan", Tipe:"370Z", Tahun:2009, topAcceleration:4360, topPower:657),
11            new Mobil26(Merk:"Subaru", Tipe:"BRZ", Tahun:2014, topAcceleration:4058, topPower:609),
12            new Mobil26(Merk:"Subaru", Tipe:"Impreza WRX STI", Tahun:2013, topAcceleration:6255, topPower:703),
13            new Mobil26(Merk:"Toyota", Tipe:"AE86 Trueno", Tahun:1986, topAcceleration:3700, topPower:553),
14            new Mobil26(Merk:"Toyota", Tipe:"86/GT86", Tahun:2014, topAcceleration:4180, topPower:609),
15            new Mobil26(Merk:"Volkswagen", Tipe:"Golf GTI", Tahun:2014, topAcceleration:4180, topPower:631)
16        };
17
18        Showroom26 showroom = new Showroom26(jnsMobil);
19
20        System.out.println(x:"=====");
21        System.out.printf(format:"| %-12s | %-18s | %-6s | %-17s | %-9s |\n", ...args:"Brand", "Type", "Year", "Top Acceleration",
22            "Top Power");
23        System.out.println(x:"=====");
24        for (Mobil26 NamMobil : jnsMobil) {
25            System.out.printf(format:"| %-12s | %-18s | %-6d | %-17d | %-9d |\n", NamMobil.Merk, NamMobil.Tipe, NamMobil.Tahun,
26                NamMobil.topAcceleration, NamMobil.topPower);
27        }
28        System.out.println(x:"=====");
29
30        // Hasil dari top_acceleration tertinggi menggunakan Divide and Conquer
31        int highestAcceleration = showroom.TopAccelerationTertinggi(start:0, jnsMobil.length - 1);
32        System.out.println("Top acceleration tertinggi : " + highestAcceleration);
33        // Hasil dari top_acceleration terendah menggunakan Divide and Conquer
34        int lowestAcceleration = showroom.TopAccelerationTerendah(start:0, jnsMobil.length - 1);
35        System.out.println("Top acceleration terendah : " + lowestAcceleration);
36        // Hasil Rata-rata top_power dari seluruh mobil menggunakan Brute Force!
37        double RataRataPower = showroom.RataTopPower();
38        System.out.println("Rata-rata top power : " + RataRataPower);
39        System.out.println(x:"=====");
40    }
```

Hasil Outputnya :

```
PS D:\Praktikum Algoritma dan Struktur Data> & 'C:\Program Files\Java\jdk-20\bin
aceStorage\9692d2458cdf9930daf158c5124dae24\redhat.java\jdt_ws\Praktikum Algoritma

=====
| Brand          | Type              | Year | 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        |
=====
Top acceleration tertinggi : 6816
Top acceleration terendah : 3700
Rata-rata top power : 633.125
=====
PS D:\Praktikum Algoritma dan Struktur Data>
```



NAMA : SATRIO WISNU ADI PRATAMA  
NIM : 2341720219  
NO ABSEN : 26  
KELAS : 1F  
MATERI : Brute-Force dan Divide-Conquer