



NAMA : HIZKIA ELSADANTA
NIM : 2341720253
KELAS : TI-1G
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

Percobaan 1

```
Minggu5 > BruteForceDivideConquer > Faktorial.java > Faktorial > faktorialDC(int)
1  package Minggu5.BruteForceDivideConquer;
2
3  public class Faktorial {
4
5      int nilai;
6
7      int faktorialBF(int n) {
8          int faktor = 1;
9          for (int i = 1; i <= n; i++) {
10             faktor = faktor * i;
11          }
12          return faktor;
13      }
14
15      int faktorialDC(int n) {
16          if (n == 1) {
17              return 1;
18          } else {
19              int faktor = n * faktorialDC(n-1);
20              return faktor;
21          }
22      }
23  }
24
```

```
1  package Minggu5.BruteForceDivideConquer;
2
3  import java.util.Scanner;
4
5  public class MainFaktorial {
6      Run | Debug
7      public static void main(String[] args) {
8          Scanner kiak = new Scanner(System.in);
9          System.out.println(x:"-----");
10         System.out.println(x:"Masukkan jumlah elemen: ");
11         int jmlE = kiak.nextInt();
12
13         Faktorial[] fk = new Faktorial[10];
14         for (int i = 0; i < jmlE; i++) {
15             fk[i] = new Faktorial();
16             System.out.print("Masukkan nilai data ke-" + (i+1) + ": ");
17             fk[i].nilai = kiak.nextInt();
18         }
19
20         System.out.println(x:"HASIL - BRUTE FORCE");
21         for (int i = 0; i < jmlE; i++) {
22             System.out.println("Hasil perhitungan faktorial menggunakan Brute Force adalah " + fk[i].faktorialBF(fk[i].nilai));
23         }
24
25         System.out.println(x:"HASIL - DIVIDE AND CONQUER");
26         for (int i = 0; i < jmlE; i++) {
27             System.out.println("Hasil perhitungan faktorial menggunakan Divide and Conquer adalah " + fk[i].faktorialDC(fk[i].nilai));
28         }
29
30         kiak.close();
31     }
32 }
33
34
```



NAMA : HIZKIA ELSADANTA
NIM : 2341720253
KELAS : TI-1G
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

```
=====
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 AND 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
=====
```

Pertanyaan

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

Jawab : if ($n == 1$) merupakan base case rekursi, mengembalikan 1 jika $n = 1$, sedangkan else merupakan bagian rekursif, dimana jika $n > 1$ maka menghitung faktorial n dengan memanggil faktorialDC($n-1$) dan mengalikan hasilnya dengan n .

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

Jawab : Bisa

```
int faktorialBF(int n) {
    // int faktor = 1;
    // for (int i = 1; i <= n; i++) {
    //     faktor = faktor * i;
    // }
    // return faktor;
    int faktor = 1;
    int i = 1;
    while (i <= n) {
        faktor = faktor * i;
        i++;
    }
    return faktor;
}
```

```
=====
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 AND 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
=====
```

Output yang dihasilkan tetap sama



NAMA : HIZKIA ELSADANTA
NIM : 2341720253
KELAS : TI-1G
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

3. Jelaskan perbedaan antara faktor $\ast = i$; dan int faktor = $n \ast \text{faktorialDC}(n-1)$; !

Jawab : faktor $\ast = i$; digunakan untuk menghitung faktorial secara iteratif

int faktor = $n \ast \text{faktorialDC}(n-1)$; digunakan untuk menghitung faktorial secara

Percobaan 2

```
package Minggu5.BruteForceDivideConquer;

public class Pangkat {
    int nilai, pangkat;

    int pangkatBF(int a, int n) {
        int hasil = 1;
        for (int i = 0; i < n; i++) {
            hasil *= a;
        }
        return hasil;
    }

    int pangkatDC(int a, int n) {
        if (n == 0) {
            return 1;
        } else {
            if (n % 2 == 1) { // bilangan ganjil
                return (pangkatDC(a, n/2) * pangkatDC(a, n/2) * a);
            } else { // bilangan genap
                return (pangkatDC(a, n/2) * pangkatDC(a, n/2));
            }
        }
    }
}
```

```
1 package Minggu5.BruteForceDivideConquer;
2
3 import java.util.Scanner;
4
5 public class MainPangkat {
6     public static void main(String[] args) {
7         Scanner kia = new Scanner(System.in);
8         System.out.println(x:"=====");
9         System.out.print(s:"Masukkan jumlah elemen yang dihitung: ");
10        int elemen = kia.nextInt();
11
12        Pangkat[] png = new Pangkat[elemen];
13        for (int i = 0; i < elemen; i++) {
14            png[i] = new Pangkat();
15            System.out.print(s:"Masukkan nilai yang hendak dipangkatkan: ");
16            png[i].nilai = kia.nextInt();
17
18            System.out.print(s:"Masukkan nilai pemangkat: ");
19            png[i].pangkat = kia.nextInt();
20        }
21
22        System.out.println(x:"=====");
23        System.out.println(x:"HASIL PANGKAT - BRUTE FORCE");
24        for (int i = 0; i < elemen; i++) {
25            System.out.println("Hasil dari " + png[i].nilai + " pangkat " + png[i].pangkat + " adalah " + png[i].pangkatBF(png[i].nilai, png[i].pangkat));
26        }
27
28        System.out.println(x:"=====");
29        System.out.println(x:"HASIL PANGKAT - DIVIDE AND CONQUER");
30        for (int i = 0; i < elemen; i++) {
31            System.out.println("Hasil dari " + png[i].nilai + " pangkat " + png[i].pangkat + " adalah " + png[i].pangkatDC(png[i].nilai, png[i].pangkat));
32        }
33
34        kia.close();
35    }
36 }
```



NAMA : HIZKIA ELSADANTA
NIM : 2341720253
KELAS : TI-1G
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

```
=====
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
=====
```

Pertanyaan

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

Jawab : pangkatBF(int a, int n) menggunakan pendekatan Brute Force dengan perulangan for untuk mengalikan bilangan a dengan dirinya sendiri sebanyak n kali. Pendekatan ini sederhana namun kurang efisien untuk bilangan pangkat besar, sedangkan pangkatDC(int a, int n) menggunakan pendekatan Divide and Conquer dengan rekursi untuk memecah masalah menjadi submasalah yang lebih kecil. Pendekatan ini memanfaatkan sifat-sifat aljabar dan lebih efisien dalam hal waktu komputasi, terutama untuk bilangan pangkat yang besar.

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

Jawab : Sudah, bisa dilihat pada proses rekursif dalam method pangkatDC(). Tahap combine terjadi di dalam blok if dan else ketika hasil dari panggilan rekursif digunakan untuk menggabungkan solusi.

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

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

Jawab :

```
Pangkat15(int nilai, int pangkat) {
    this.nilai = nilai;
    this.pangkat = pangkat;
}
```

```
Pangkat15[] png = new Pangkat15[elemen];
for (int i = 0; i < elemen; i++) {
    System.out.print(s:"Masukkan nilai yang hendak dipangkatkan: ");
    int nilai = kia.nextInt();

    System.out.print(s:"Masukkan nilai pemangkat: ");
    int pangkat = kia.nextInt();
    png[i] = new Pangkat15(nilai, pangkat);
}
```



NAMA : HIZKIA ELSADANTA
NIM : 2341720253
KELAS : TI-1G
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

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

Jawab :

```
int menu;
do {
    System.out.println(x:"== Pilih method yang ingin dijalankan ==");
    System.out.println(x:"1. Brute Force");
    System.out.println(x:"2. Divide and Conquer");
    System.out.println(x:"Menu : ");
    menu = kia.nextInt();

    switch (menu) {
        case 1:
            System.out.println(x:"=====");
            System.out.println(x:"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(x:"=====");
            System.out.println(x:"HASIL PANGKAT - DIVIDE AND 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;

        default:
            System.out.println(x:"Angka yang anda masukkan tidak valid");
    }
} while (menu < 1 || menu > 2);
```

```
=====
Masukkan jumlah elemen yang dihitung: 2
Masukkan nilai yang hendak dipangkatkan: 3
Masukkan nilai pemangkat: 2
Masukkan nilai yang hendak dipangkatkan: 4
Masukkan nilai pemangkat: 2
== Pilih method yang ingin dijalankan ==
1. Brute Force
2. Divide and Conquer
Menu :
1
=====
HASIL PANGKAT - BRUTE FORCE
Hasil dari 3 pangkat 2 adalah 9
Hasil dari 4 pangkat 2 adalah 16
```



NAMA : HIZKIA ELSADANTA
NIM : 2341720253
KELAS : TI-1G
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

Percobaan 3

```
package P4.BruteForceDivideConquer;

public class Sum15 {

    int elemen;
    double keuntungan[], total;

    Sum15(int elemen) {
        this.elemen = elemen;
        this.keuntungan = new double[elemen];
        this.total = 0;
    }

    double totalBF(double arr[]) {
        for (int i = 0; i < elemen; i++) {
            total = total + arr[i];
        }
        return total;
    }

    double totalDC(double arr[], int l, int r) {
        if (l == r) {
            return arr[l];
        } else if (l < r) {
            int mid = (l + r) / 2;
            double lsum = totalDC(arr, l, mid-1);
            double rsum = totalDC(arr, mid+1, r);
            return lsum + rsum + arr[mid];
        }
        return 0;
    }
}
```

```
package P4.BruteForceDivideConquer;

import java.util.Scanner;

public class MainSum15 {

    public static void main(String[] args) {
        Scanner kia = new Scanner(System.in);
        System.out.println(x:"=====");
        System.out.println(x:"Program Menghitung Keuntungan Total (Satuan Juta. Misal 5.9)");
        System.out.print(s:"Masukkan jumlah bulan : ");
        int elm = kia.nextInt();

        Sum15 sm = new Sum15(elm);
        System.out.println(x:"=====");
        for (int i = 0; i < sm.elemen; i++) {
            System.out.print("Masukkan untung bulan ke- " + (i+1) + " = ");
            sm.keuntungan[i] = kia.nextDouble();
        }

        System.out.println(x:"=====");
        System.out.println(x:"Algoritma Brute Force");
        System.out.println("Total keuntungan perusahaan selama " + sm.elemen + " bulan adalah = " + sm.totalBF(sm.keuntungan));
        System.out.println(x:"=====");
        System.out.println(x:"Algoritma Divide Conquer");
        System.out.println("Total keuntungan perusahaan selama " + sm.elemen + " bulan adalah = " + Math.round(sm.totalDC(sm.keuntungan, 1:0, sm.elemen-1) * 100.0) / 100.0 );
        kia.close();
    }
}
```



NAMA : HIZKIA ELSADANTA
NIM : 2341720253
KELAS : TI-1G
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

```
=====
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
=====
```

Pertanyaan

1. Mengapa terdapat formulasi return value berikut? Jelaskan!

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

Jawab : Formulasi tersebut merupakan implementasi dari algoritma Divide and Conquer untuk menghitung total dari elemen-elemen array secara rekursif. Seperti yang sudah diketahui, cara kerja Divide and Conquer adalah dengan memecah masalah menjadi submasalah yang lebih kecil, kemudian setiap submasalah diselesaikan secara rekursif, setelah itu hasil dari semua submasalah digabung untuk mendapatkan solusi akhir. Oleh karena itu, formulasi diatas sedang menggabungkan total setiap submasalah (sum bagian kiri(lsum) + sum bagian kanan (rsum) dan arr mid) untuk menghitung total keseluruhan dari seluruh array yang dikirim kedalam fungsi.

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

Jawab : Variabel mid dibutuhkan karena variabel tersebut digunakan untuk menentukan bagian tengah (middle) dari array saat ini yang sedang diproses. Dengan menggunakan variabel mid kita dapat membagi array menjadi dua bagian (kiri dan kanan), sehingga algoritma secara efisien menangani setiap submasalah kemudian digabungkan untuk mendapatkan solusi akhir. Oleh karena itu variabel mid sangat penting dalam algoritma Divide Conquer pada kode 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 : HIZKIA ELSADANTA
NIM : 2341720253
KELAS : TI-1G
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

```
System.out.print(s:"Banyak Perusahaan : ");
int banyak = kia.nextInt();
System.out.println();

for (int i = 0; i < banyak; i++) {
    System.out.println("Peusahaan ke-" + (i+1));
    System.out.println(x:"=====");
    System.out.println(x:"Program Menghitung Keuntungan Total (Satuan Juta. Misal 5.9)");
    System.out.print(s:"Masukkan jumlah bulan : ");
    int elm = kia.nextInt();

    Sum15 sm = new Sum15(elm);
    System.out.println(x:"=====");
    for (int j = 0; j < sm.elemen; j++) {
        System.out.print("Masukkan untung bulan ke- " + (j+1) + " = ");
        sm.keuntungan[j] = kia.nextDouble();
    }

    System.out.println(x:"=====");
    System.out.println(x:"Algoritma Brute Force");
    System.out.println("Total keuntungan perusahaan selama " + sm.elemen + " bulan adalah = " + sm.totalBF(sm.keuntungan));
    System.out.println(x:"=====");
    System.out.println(x:"Algoritma Divide Conquer");
    System.out.println("Total keuntungan perusahaan selama " + sm.elemen + " bulan adalah = " + Math.round(sm.totalDC(sm.keuntungan, 1:0, sm.elemen-1) * 100.0) / 100.0 );
    System.out.println();
}
kia.close();
```

```
Banyak Perusahaan : 2

Peusahaan ke-1
=====
Program Menghitung Keuntungan Total (Satuan Juta. Misal 5.9)
Masukkan jumlah bulan : 5
=====
Masukkan untung bulan ke- 1 = 4
Masukkan untung bulan ke- 2 = 6
Masukkan untung bulan ke- 3 = 7
Masukkan untung bulan ke- 4 = 8
Masukkan untung bulan ke- 5 = 9
=====
Algoritma Brute Force
Total keuntungan perusahaan selama 5 bulan adalah = 34.0
=====
Algoritma Divide Conquer
Total keuntungan perusahaan selama 5 bulan adalah = 34.0

Peusahaan ke-2
=====
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
```

Link github :

https://github.com/Kiaakk/Algoritma_Struktur_Data_1G_15.git