



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
MATERI : Brute Force dan Divide Conquer

4. 2. Menghitung Nilai Faktorial dengan Algoritma Brute Force dan Divide and Conquer

1. Source Code

```
1 public class faktorial09 {
2     public int nilai;
3
4     int faktorialBF(int n) {
5         int fakto = 1;
6         for (int i = 1; i <= n; i++) {
7             fakto += fakto * i;
8         }
9         return fakto;
10    }
11
12    int faktorialDC(int n) {
13        if (n == 1) {
14            return 1;
15        } else {
16            int fakto = n * faktorialDC(n - 1);
17            return fakto;
18        }
19    }
20 }
```

```
1 public class faktorialMain09 {
2     public static void main(String[] args) {
3         Scanner input = new Scanner(System.in);
4
5         System.out.println("-----");
6         System.out.print("Masukkan bilangan: ");
7         int iJml = input.nextInt();
8
9         faktorial09[] fk = new faktorial09[iJml];
10        for (int i = 0; i < iJml; i++) {
11            fk[i] = new faktorial09();
12            System.out.print("Masukkan nilai data ke-" + (i + 1) + ": ");
13            fk[i].nilai = input.nextInt();
14        }
15
16        System.out.println("\nHASIL - BRUTE FORCE");
17        for (int i = 0; i < iJml; i++) {
18            System.out.println(
19                "Hasil penghitungan faktorial menggunakan Brute Force adalah "
20                + fk[i].faktorialBF(fk[i].nilai));
21        }
22
23        System.out.println("\nHASIL - DIVIDE AND CONQUER");
24        for (int i = 0; i < iJml; i++) {
25            System.out.println(
26                "Hasil penghitungan faktorial menggunakan Divide and Conquer adalah "
27                + fk[i].faktorialDC(fk[i].nilai));
28        }
29    }
30 }
```



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
MATERI : Brute Force dan Divide Conquer

2. Output

```
-----  
Masukkan bilangan: 3  
Masukkan nilai data ke-1: 5  
Masukkan nilai data ke-2: 8  
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 - DIVIDE AND CONQUER  
Hasil penghitungan faktorial menggunakan Divide and Conquer adalah 120  
Hasil penghitungan faktorial menggunakan Divide and Conquer adalah 40320  
Hasil penghitungan faktorial menggunakan Divide and Conquer adalah 6
```

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 adalah base case untuk memberhentikan program
Ketika terpenuhi, sedangkan else merupakan *recursion call* yang menyediakan pengulangan untuk menyerdehanakan masalah.
2. Apakah memungkinkan perulangan pada method faktorialBF() dirubah selain menggunakan for?Buktikan!

Jawab: Bisa menggunakan perulangan *while*

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

3. Jelaskan perbedaan antara **fakto *= i;** dan **int fakto = n * faktorialDC(n-1);** !
Jawab: Perbedaanya adalah **fakto *= i;** termasuk di dalam perulangan iterative sedangkan **int fakto = n * faktorialDC(n-1);** termasuk di dalam perulangan rekursif karena memanggil fungsinya sendiri yaitu faktorialDC.



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
MATERI : Brute Force dan Divide Conquer

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

1. Source Code

```
1 public class pangkat09 {
2     public int nilai, pangkat;
3
4     int pangkatBF(int a, int n) {
5         int hasil = 1;
6         for (int i = 0; i < n; i++) {
7             hasil *= a;
8         }
9         return hasil;
10    }
11
12    int pangkatDC(int a, int n) {
13        if (n == 0) {
14            return 1;
15        } else {
16            if (n % 2 == 1) {
17                return (pangkatDC(a, n/2) * pangkatDC(a, n/2) * a);
18            } else {
19                return (pangkatDC(a, n/2) * pangkatDC(a, n/2));
20            }
21        }
22    }
23 }
```

```
1 public class pangkatMain09 {
2     public static void main(String[] args) {
3         Scanner input = new Scanner(System.in);
4         System.out.println("=====");
5         System.out.print("Masukkan jumlah elemen yang dihitung: ");
6         int elemen = input.nextInt();
7
8         pangkat09[] png = new pangkat09[elemen];
9         for (int i = 0; i < elemen; i++) {
10             png[i] = new pangkat09();
11             System.out.print("Masukkan nilai yang hendak dipangkatkan: ");
12             png[i].nilai = input.nextInt();
13             System.out.print("Masukkan nilai pemangkat: ");
14             png[i].pangkat = input.nextInt();
15         }
16
17         System.out.println("\nHASIL PANGKAT - BRUTE FORCE");
18         for (int i = 0; i < elemen; i++) {
19             System.out.println(
20                 "Hasil penghitungan pangkat menggunakan Brute Force adalah "
21                 + png[i].nilai + " pangkat "
22                 + png[i].pangkat + " = "
23                 + png[i].pangkatBF(png[i].nilai, png[i].pangkat));
24         }
25
26         System.out.println("\nHASIL PANGKAT - DIVIDE AND CONQUER");
27         for (int i = 0; i < elemen; i++) {
28             System.out.println(
29                 "Hasil penghitungan pangkat menggunakan Divide and Conquer adalah "
30                 + png[i].nilai + " pangkat "
31                 + png[i].pangkat + " = "
32                 + png[i].pangkatDC(png[i].nilai, png[i].pangkat));
33         }
34     }
35 }
```

2. Ouput



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
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 penghitungan pangkat menggunakan Brute Force adalah 6 pangkat 2 = 36
Hasil penghitungan pangkat menggunakan Brute Force adalah 4 pangkat 3 = 64

HASIL PANGKAT - DIVIDE AND CONQUER
Hasil penghitungan pangkat menggunakan Divide and Conquer adalah 6 pangkat 2 = 36
Hasil penghitungan pangkat menggunakan Divide and Conquer adalah 4 pangkat 3 = 64
```

Questions:

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

Jawab: method PangkatBf() menggunakan perulangan iterative yaitu menggunakan for sedangkan method PangkatDC() menggunakan perulangan rekursif yaitu memanggil dirinya sendiri.

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

Jawab: Terjadi pada baris return karena menggabungkan perkaliannya

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

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

Jawab: pada file pangkat09 buatlah konstruktor yang berparameter

```
public pangkat09 (int nilai, int pangkat) {
    this.nilai = nilai;
    this.pangkat = pangkat;
}
```

pada file main buat objek pangkat09 menggunakan konstruktor berisi int nilai dan int pangkat

```
pangkat09[] png = new pangkat09[elemen];
for (int i = 0; i < elemen; i++) {
    System.out.print(s:"Masukkan nilai yang hendak dipangkatkan: ");
    int nilai = input.nextInt();
    System.out.print(s:"Masukkan nilai pemangkat: ");
    int pangkat = input.nextInt();

    png[i] = new pangkat09(nilai, pangkat);
}
```

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

Jawab: Penambahan switch-case pada file main



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
MATERI : Brute Force dan Divide Conquer

```
1 public class pangkatMain09 {
2     public static void main(String[] args) {
3         Scanner input = new Scanner(System.in);
4         char back;
5
6         System.out.println("=====");
7         System.out.print("Masukkan jumlah elemen yang dihitung: ");
8         int elemen = input.nextInt();
9
10        pangkat09[] png = new pangkat09[elemen];
11        for (int i = 0; i < elemen; i++) {
12            System.out.print("Masukkan nilai yang hendak dipangkatkan: ");
13            int nilai = input.nextInt();
14            System.out.print("Masukkan nilai pemangkat: ");
15            int pangkat = input.nextInt();
16
17            png[i] = new pangkat09(nilai, pangkat);
18        }
19
20        do {
21            System.out.println("\nPilih algoritma yang diinginkan: \n1. Brute Force \n2. Divide and Conquer");
22            System.out.print("--> ");
23            char pilih = input.next().charAt(0);
24
25            switch (pilih) {
26                case '1':
27                    System.out.println("HASIL PANGKAT - BRUTE FORCE");
28                    for (int i = 0; i < elemen; i++) {
29                        System.out.println(
30                            "Hasil penghitungan pangkat menggunakan Brute Force adalah "
31                            + png[i].nilai + " pangkat "
32                            + png[i].pangkat + " = "
33                            + png[i].pangkatBF(png[i].nilai, png[i].pangkat));
34                    }
35                    break;
36                case '2':
37                    System.out.println("HASIL PANGKAT - DIVIDE AND CONQUER");
38                    for (int i = 0; i < elemen; i++) {
39                        System.out.println(
40                            "Hasil penghitungan pangkat menggunakan Divide and Conquer adalah "
41                            + png[i].nilai + " pangkat "
42                            + png[i].pangkat + " = "
43                            + png[i].pangkatDC(png[i].nilai, png[i].pangkat));
44                    }
45                    break;
46                default:
47                    System.out.println("Pilihan tidak valid");
48                    break;
49            }
50            System.out.println("=====");
51            System.out.print("Apakah anda ingin kembali ke menu(y/n)? ");
52            back = input.next().charAt(0);
53        } while (back == 'y' || back == 'Y');
54    }
55 }
```

output



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
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

Pilih algoritma yang diinginkan:
1. Brute Force
2. Divide and Conquer
--> 1
HASIL PANGKAT - BRUTE FORCE
Hasil penghitungan pangkat menggunakan Brute Force adalah 6 pangkat 2 = 36
Hasil penghitungan pangkat menggunakan Brute Force adalah 4 pangkat 3 = 64
=====
Apakah anda ingin kembali ke menu(y/n)? y

Pilih algoritma yang diinginkan:
1. Brute Force
2. Divide and Conquer
--> 2
HASIL PANGKAT - DIVIDE AND CONQUER
Hasil penghitungan pangkat menggunakan Divide and Conquer adalah 6 pangkat 2 = 36
Hasil penghitungan pangkat menggunakan Divide and Conquer adalah 4 pangkat 3 = 64
=====
Apakah anda ingin kembali ke menu(y/n)? n
```



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
MATERI : Brute Force dan Divide Conquer

4.4 Menghitung Sum Array dengan Algoritma Brute Force dan Divide Conquer

1. Source Code

```
1 public class Sum09 {
2     int elemen;
3     double keuntungan[], total;
4
5     Sum09(int elemen) {
6         this.elemen = elemen;
7         this.keuntungan = new double[elemen];
8         this.total = 0;
9     }
10
11     double totalBF(double arr[]) {
12         for (int i = 0; i < elemen; i++) {
13             total = total + arr[i];
14         }
15         return total;
16     }
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);
24             double rsum = totalDC(arr, mid + 1, r);
25             return lsum + rsum;
26         }
27         return 0;
28     }
29 }
```

```
1 public class mainSum09 {
2     public static void main(String[] args) {
3         Scanner input = new Scanner(System.in);
4         System.out.println("=====");
5         System.out.println("Program Mengitung Keuntungan Total (Satuan Juta. Misal 5.9)");
6         System.out.print("Masukkan jumlah bulan: ");
7         int elm = input.nextInt();
8
9         Sum09 sum = new Sum09(elm);
10        System.out.println("=====");
11        for (int i = 0; i < sum.elemen; i++) {
12            System.out.print("Masukkan untung bulan ke-" + (i + 1) + " = ");
13            sum.keuntungan[i] = input.nextDouble();
14        }
15
16        System.out.println("=====");
17        System.out.println("Algoritma Brute Force");
18        System.out.println(
19            "Total Keuntungan perusahaan selama " + sum.elemen + " bulan adalah "
20            + sum.totalBF(sum.keuntungan));
21
22        System.out.println("=====");
23        System.out.println("Algoritma Divide and Conquer");
24        System.out.printf(
25            "Total Keuntungan perusahaan selama %d bulan adalah %.2f\n", sum.elemen,
26            sum.totalDC(sum.keuntungan, 0, sum.elemen - 1));
27    }
28 }
```




NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
MATERI : Brute Force dan Divide Conquer

2. Output

```
=====
Program Mengitung 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 and Conquer
Total Keuntungan perusahaan selama 5 bulan adalah 40.34
=====
```

Questions:

1. Mengapa terdapat formulasi return value berikut?Jelaskan!

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

Jawab: Untuk mengembalikan atau melakukan combine nilai penjumlahan dari lsum, lsum dan arr[mid]. Tetapi kode yang benar hanya penjumlahan untuk lsum dan rsum yaitu "return lsum + rsum" karena jika terdapat arr[mid] hasilnya akan ditambah dengan nilai yang ada pada indeks mid.

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

Jawab: variable mid pada method TotalDC() berfungsi untuk menentukan titik tengah dari rentang elemen yang akan diproses. Dengan itu elemen bisa dibagi menjadi 2 bagian yang lebih sederhana dengan jumlah elemen yang mendekati pada setiap bagian.

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: Pada file main terdapat perubahan untuk menghitung beberapa perusahaan dan menyimpannya kemudian pemanggilan method brute force dan divide and conquer.



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
MATERI : Brute Force dan Divide Conquer

```
1 public class mainSum09 {
2     public static void main(String[] args) {
3         Scanner input = new Scanner(System.in);
4         System.out.println("=====");
5         System.out.println("Program Mengitung Keuntungan Total (Satuan Juta. Misal 5.9)");
6
7
8         System.out.println("=====");
9         System.out.print("Masukkan jumlah perusahaan: ");
10        int jmlPerusahaan = input.nextInt();
11
12        Sum09[] perusahaan = new Sum09[jmlPerusahaan];
13
14        for (int i = 0; i < perusahaan.length; i++) {
15            System.out.print("\nMasukkan jumlah bulan untuk perusahaan ke- " + (i + 1) + ": ");
16            int elm = input.nextInt();
17
18            perusahaan[i] = new Sum09(elm);
19
20            System.out.println("=====");
21            for (int j = 0; j < perusahaan[i].elemen; j++) {
22                System.out.print("Masukkan untung bulan ke- " + (j + 1) + " perusahaan ke- " + (i + 1) + "= ");
23                perusahaan[i].keuntungan[j] = input.nextDouble();
24            }
25        }
26
27        System.out.println("=====");
28        System.out.println("\nAlgoritma Brute Force");
29        for (int i = 0; i < jmlPerusahaan; i++) {
30            System.out.println(
31                "Total Keuntungan perusahaan ke- " + (i + 1) + " selama "
32                + perusahaan[i].elemen + " bulan adalah "
33                + perusahaan[i].totalBF(perusahaan[i].keuntungan));
34        }
35
36        System.out.println("=====");
37        System.out.println("Algoritma Divide and Conquer");
38        for (int i = 0; i < jmlPerusahaan; i++) {
39            System.out.printf(
40                "Total Keuntungan perusahaan ke-%d selama %d bulan adalah %.2f\n", (i + 1), perusahaan[i].elemen,
41                perusahaan[i].totalDC(perusahaan[i].keuntungan, 0, (perusahaan[i].elemen - 1)));
42        }
43    }
44 }
45
```

=====

Program Mengitung Keuntungan Total (Satuan Juta. Misal 5.9)

=====

Masukkan jumlah perusahaan: 2

Masukkan jumlah bulan untuk perusahaan ke-1: 5

=====

Masukkan untung bulan ke-1 perusahaan ke-1= 8.5
Masukkan untung bulan ke-2 perusahaan ke-1= 9.54
Masukkan untung bulan ke-3 perusahaan ke-1= 7.2
Masukkan untung bulan ke-4 perusahaan ke-1= 9.1
Masukkan untung bulan ke-5 perusahaan ke-1= 6

Masukkan jumlah bulan untuk perusahaan ke-2: 4

=====

Masukkan untung bulan ke-1 perusahaan ke-2= 10
Masukkan untung bulan ke-2 perusahaan ke-2= 6.1
Masukkan untung bulan ke-3 perusahaan ke-2= 8.9
Masukkan untung bulan ke-4 perusahaan ke-2= 7

=====

Algoritma Brute Force

Total Keuntungan perusahaan ke-1 selama 5 bulan adalah 40.339999999999996
Total Keuntungan perusahaan ke-2 selama 4 bulan adalah 32.0

=====

Algoritma Divide and Conquer

Total Keuntungan perusahaan ke-1 selama 5 bulan adalah 40.34
Total Keuntungan perusahaan ke-2 selama 4 bulan adalah 32.00



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
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:

- 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!

```
1 public class showroom09 {
2     public String merk, tipe, tahun;
3     public int top_acceleration;
4     public double top_power;
5
6     public showroom09(String merk, String tipe, String tahun, int top_acceleration, double top_power) {
7         this.merk = merk;
8         this.tipe = tipe;
9         this.tahun = tahun;
10        this.top_acceleration = top_acceleration;
11        this.top_power = top_power;
12    }
13
14    // high itu r dan l itu low
15    int[] findMinMax(showroom09[] arr, int l, int r) {
16        int max, min;
17
18        if (l == r) {
19            return new int[] { arr[l].top_acceleration, arr[l].top_acceleration };
20        } else if (l < r) {
21            int mid = (l + r) / 2;
22            int[] left = findMinMax(arr, l, mid);
23            int[] right = findMinMax(arr, mid + 1, r);
24
25            max = Math.max(left[1], right[1]);
26            min = Math.min(left[0], right[0]);
27            return new int[] { min, max };
28        }
29        return new int[] { 0, 0 };
30    }
31
32    public void tampilInfoAcceleration(showroom09[] arr, int acceleration) {
33        for (showroom09 mobil : arr) {
34            if (mobil.top_acceleration == acceleration) {
35                System.out.println(mobil.merk + " " + mobil.tipe + " Tahun " + mobil.tahun
36                    + " dengan top acceleration sebesar " + mobil.top_acceleration);
37                break;
38            }
39        }
40    }
41
42    double averagePower(showroom09[] arr) {
43        double sum = 0;
44        for (showroom09 mobil : arr) {
45            sum += mobil.top_power;
46        }
47        return sum / arr.length;
48    }
49 }
```



NAMA : Duta Kurnia Ardhani
NIM : 2341720139
NO ABSEN : 09
KELAS : 1C
MATERI : Brute Force dan Divide Conquer

```
1 public class mainShowroom09 {  
2     public static void main(String[] args) {  
3         showroom09[] showroom = new showroom09[8];  
4  
5         showroom[0] = new showroom09("BMW", "M2 Coupe", "2016", 6816, 728);  
6         showroom[1] = new showroom09("Ford", "Fiesta ST", "2014", 3921, 575);  
7         showroom[2] = new showroom09("Nissan", "370Z", "2009", 4360, 657);  
8         showroom[3] = new showroom09("Subaru", "BRZ", "2014", 4058, 609);  
9         showroom[4] = new showroom09("Subaru", "Impreza WRX STI", "2013", 6255, 703);  
10        showroom[5] = new showroom09("Toyota", "AE86 Trueno", "1986", 3700, 553);  
11        showroom[6] = new showroom09("Toyota", "86/GT86", "2014", 4180, 609);  
12        showroom[7] = new showroom09("Volkswagen", "Golf GTI", "2014", 4180, 631);  
13  
14  
15        int[] minMax = showroom[0].findMinMax(showroom, 0, showroom.length - 1);  
16        System.out.println("Mobil dengan acceleration terendah:");  
17        showroom[0].tampilInfoAcceleration(showroom, minMax[0]);  
18        System.out.println("\nMobil dengan acceleration tertinggi:");  
19        showroom[0].tampilInfoAcceleration(showroom, minMax[1]);  
20  
21        double averagePower = showroom[0].averagePower(showroom);  
22        System.out.println("\nRata-rata top power semua mobil: " + averagePower);  
23    }  
24 }
```

Mobil dengan acceleration terendah:

Toyota AE86 Trueno Tahun 1986 dengan top acceleration sebesar 3700

Mobil dengan acceleration tertinggi:

BMW M2 Coupe Tahun 2016 dengan top acceleration sebesar 6816

Rata-rata top power semua mobil: 633.125