



NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

*FILE NAME =03_Alexander Agung Raya _1F_P5

* Pertemuan 5

BRUTE FORCE DAN DIVIDE CONQUER

Percobaan 1 :

Faktorial03 :

```
1 package P5;  
2  
3 public class Faktorial03 {  
4     public 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    int faktorialDC(int n) {  
14        if (n==1) {  
15            return 1;  
16        }  
17        else {  
18            int fakto = n * faktorialDC(n-1);  
19            return fakto;  
20        }  
21    }  
22 }  
23
```

MainFaktor03 :

```
1 package P5;  
2 import java.util.Scanner;  
3 public class MainFaktorial03 {  
4     public static void main(String[] args) {  
5         Scanner sc = new Scanner(System.in);  
6         System.out.println("=====");  
7         System.out.println("Masukan Jumlah Elemen :");  
8         int iJml = sc.nextInt();  
9  
10        Faktorial03[] fk = new Faktorial03[10];  
11        for (int i = 0; i < iJml; i++) {  
12            fk[i] = new Faktorial03();  
13            System.out.println("Masukkan nilai data ke-" + (i + 1) + " :");  
14            int iNilai = sc.nextInt();  
15            fk[i].nilai = iNilai;  
16        }  
17  
18        System.out.println("Hasil-BRUTE Force");  
19        for (int i=0; i < iJml; i++) {  
20            System.out.println("Hasil perhitungan Faktorial menggunakan Blue Force adalah : " + fk[i].faktorialBF(fk[i].nilai));  
21        }  
22        System.out.println("Hasil-Divince And conquer");  
23        for (int i=0; i < iJml; i++) {  
24            System.out.println("Hasil perhitungan Faktorial menggunakan Divince And conquer Adalah "+fk[i].faktorialDC(fk[i].nilai));  
25        }  
26    }  
27 }  
28
```



NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

Hasil :

```
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 Blue Force adalah :120  
Hasil perhitungan Faktorial menggunakan Blue Force adalah :40320  
Hasil perhitungan Faktorial menggunakan Blue 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  
ASUS > S Prak 1F 03 > B. Bmaster > B 71 > E
```

Pertanyaan :

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

Jawab :

kalau menggunakan if, maka base case atau kasus dasar akan ditangani secara eksplisit di dalam kondisi if. Misalnya, jika nilai input adalah 0 atau 1, maka hasilnya adalah 1.

kalau menggunakan else, maka base case akan ditangani secara implisit dalam bagian else. Ini berarti bahwa kondisi if menangani kasus rekursif, sementara else menangani kasus dasar. Misalnya, jika nilai input lebih besar dari 1, maka rekursi dilakukan, jika tidak, hasilnya adalah 1.

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

Jawab :

Ya, memungkinkan untuk mengimplementasikan metode faktorial dengan rekursi tanpa menggunakan perulangan for. Salah satu cara untuk melakukan hal ini adalah dengan menggunakan rekursi tanpa batas (tail recursion) dan penerapan aturan rekursif untuk menghitung nilai faktorial. Berikut adalah contoh implementasi metode faktorial tanpa menggunakan perulangan

```
1 int faktorialBF(int n) {  
2     int fakto = 1;  
3     for (int i = 1; i <= n; i++) {  
4         fakto = fakto * i;  
5     }  
6     return fakto;  
7 }
```



NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

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

Jawab :

*fakto *= i; Ini adalah operasi penugasan singkat (short-hand assignment) untuk mengalikan fakto dengan i dan menyimpan hasilnya kembali ke fakto. Ini digunakan dalam perulangan untuk mengakumulasi hasil perkalian.*

*int fakto = n * faktorialDC(n-1); Ini adalah rekursi yang digunakan untuk menghitung nilai faktorial dari n. Ini memanggil fungsi faktorialDC dengan nilai n-1 dan mengalikan hasilnya dengan n. Ini merupakan implementasi dari algoritma Divide Conquer untuk mencari nilai faktorial.*

Percobaan 2 :

Pangkat03 :

```
1 package P5;
2
3 public class Pangkat03 {
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    int pangkatDC(int a, int n) {
14        if (n == 1) {
15            return a;
16        } else {
17            if (n%2 == 1) {
18                return (pangkatDC(a, n/2)*pangkatDC(a, n/2)*a);
19            } else {
20                return (pangkatDC(a, n/2)*pangkatDC(a, n/2));
21            }
22        }
23    }
24 }
25
26
```

Mainpangkat03 :

```
1 package P5;
2 import java.util.Scanner;
3 public class MainPangkat03 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         System.out.println("=====");
7         System.out.println("Masukan Jumlah element yg di hitung : ");
8         int elemen = sc.nextInt();
9
10        Pangkat03 [] png = new Pangkat03[elemen];
11        for (int i = 0; i < elemen; i++) {
12            png[i] = new Pangkat03();
13            System.out.println("Masukan nilai hendak pangkat : ");
14            int nilai = sc.nextInt();
15            png[i].nilai = nilai;
16            System.out.println("Masukan nilai pangkat ");
17            int pangkat = sc.nextInt();
18            png[i].pangkat = pangkat;
19        }
20
21        System.out.println("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 + " Adalah " + png[i].PangkatBF(png[i].nilai,png[i].pangkat));
24        }
25        System.out.println("Hasil pangkat - Divince And Counquer");
26        for (int i = 0; i < elemen; i++) {
27            System.out.println("Hasil dari " + png [i].nilai+ " pangkat " + png [i].pangkat+ " Adalah " + png[i].pangkatDC(png[i].nilai, png[i].pangkat) );
28        }
29    }
30 }
31
```



NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

Hasil :

```
Masukan Jumlah element yg di hitung :
2
Masukan nilai hendak pangkat :
6
Masukan nilai pangkat
2
Masukan nilai hendak pangkat :
4
Masukan nilai pangkat
3
Hasil pangkat - Brute Force
Hasil dari 6Pangkat 2Adalah 36
Hasil dari 4Pangkat 3Adalah 64
Hasil pangkat - Divince And Counquer
Hasil dari 6pangkat 2Adalah 36
Hasil dari 4pangkat 3Adalah 64
ASUS > Prak 1F 03 > @master > 72 >
```

Pertanyaan :

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

Jawab :

PangkatDC untuk menghitung pangkat dengan melakukan perkalian berulang kali secara sederhana

PangkatBF untuk dibagi secara memperkalikan diri sendiri jika n adalah genap maka dilakukan pada $n/2$ dipangkatkan dengan diri sendiri

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

Jawab :

Iya, return (pangkatDC(a, $n/2$)*pangkatDC(a, $n/2$)*a); dan juga return (pangkatDC(a, $n/2$)*pangkatDC(a, $n/2$)); dikalikan untuk ganjil lalu di combine

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

Jawab :

```
1 package P1;
2 public class PangkatBF {
3     public int PangkatBF(int a, int n) {
4         return PangkatBF(a, n, 1);
5     }
6     private int PangkatBF(int a, int n, int k) {
7         if (n == 1) {
8             return k;
9         }
10        return PangkatBF(a, n-1, k*a);
11    }
12 }
13
14 package P1;
15 public class PangkatDC {
16     public int PangkatDC(int a, int n) {
17         return PangkatDC(a, n, 1);
18     }
19     private int PangkatDC(int a, int n, int k) {
20         if (n == 1) {
21             return k;
22         }
23         if (n % 2 == 0) {
24             return PangkatDC(a, n/2, k);
25         }
26         return PangkatDC(a, n/2, k*a);
27     }
28 }
29
30 package P1;
31 public class PangkatDC {
32     public int PangkatDC(int a, int n) {
33         return PangkatDC(a, n, 1);
34     }
35     private int PangkatDC(int a, int n, int k) {
36         if (n == 1) {
37             return k;
38         }
39         if (n % 2 == 0) {
40             return PangkatDC(a, n/2, k);
41         }
42         return PangkatDC(a, n/2, k*a);
43     }
44 }
```




NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

```
1 package P5;  
2  
3 public class Sum03 {  
4     int elemen;  
5     double keutungan[], total;  
6  
7     int sum(int elemen) {  
8         this.elemen = elemen;  
9         this.keutungan = new double[elemen];  
10        this.total = 0;  
11        return elemen;  
12    }  
13  
14    double totalBF(double arr[]) {  
15        for (int i = 0; i < elemen; i++) {  
16            total = total + arr[i];  
17        }  
18        return total;  
19    }  
20  
21    double totalDC(double arr[], int l, int r) {  
22        if (l == r) {  
23            return arr[l];  
24        } else if (l < r) {  
25            int mid = (l + r) / 2;  
26            double lsum = totalDC(arr, l, mid);  
27            double rsum = totalDC(arr, mid + 1, r);  
28            return lsum + rsum;  
29        }  
30        return 0;  
31    }  
32 }  
33
```

mainSum03 :

```
1 package P5;  
2  
3 import java.util.Scanner;  
4  
5 public class Main03 {  
6     public static void main(String[] args) {  
7         Scanner sc = new Scanner(System.in);  
8         System.out.println("Masukan jumlah bulan: ");  
9         int bulan = sc.nextInt();  
10        double arr[] = new double[bulan];  
11  
12        System.out.println("Masukan untung ke bulan ke- 1 = ");  
13        arr[0] = sc.nextDouble();  
14  
15        System.out.println("Masukan untung ke bulan ke- 2 = ");  
16        arr[1] = sc.nextDouble();  
17  
18        System.out.println("Masukan untung ke bulan ke- 3 = ");  
19        arr[2] = sc.nextDouble();  
20  
21        System.out.println("Masukan untung ke bulan ke- 4 = ");  
22        arr[3] = sc.nextDouble();  
23  
24        System.out.println("Masukan untung ke bulan ke- 5 = ");  
25        arr[4] = sc.nextDouble();  
26  
27        Sum03 s = new Sum03();  
28        s.sum(bulan);  
29        System.out.println("Total keuntungan perusahaan selama " + bulan + " bulan adalah = " + s.totalDC(arr, 0, arr.length - 1));  
30    }  
31 }
```

Hasil :

```
8.5  
Masukan untung ke bulan ke- 2=  
9.54  
Masukan untung ke bulan ke- 3=  
7.2  
Masukan untung ke bulan ke- 4=  
9.1  
Masukan untung ke 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 = 40.339999999999996
```



NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

Pertanyaan :

1. Mengapa terdapat formulasi return value berikut?Jelaskan!

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

Jawab :

metode totalDC adalah tidak tepat. Seharusnya, itu hanya kembali lsum + rsum tanpa menambahkan arr[mid], karena dalam algoritma divide and conquer, kita sudah membagi array menjadi dua bagian dan menghitung jumlahnya secara rekursif.

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

Jawab :

karena algoritma divide and conquer membagi masalah menjadi submasalah yang lebih kecil

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 :

```
1 package P5.Du03;
2 import java.util.Scanner;
3
4
5 public class Main {
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         System.out.println("Masukkan jumlah perusahaan:");
9         int numCompanies = sc.nextInt();
10
11         // Array 2D untuk menyimpan keuntungan perusahaan
12         double[][] keuntunganPerusahaan = new double[numCompanies][];
13
14         // Input keuntungan perusahaan
15         for (int i = 0; i < numCompanies; i++) {
16             System.out.println("Masukkan jumlah bulan untuk perusahaan " + (i+1) + ":");
17             int numMonths = sc.nextInt();
18             keuntunganPerusahaan[i] = new double[numMonths];
19
20             System.out.println("Masukkan keuntungan per bulan untuk perusahaan " + (i+1) + ":");
21             for (int j = 0; j < numMonths; j++) {
22                 System.out.print("Masukkan keuntungan untuk bulan " + (j+1) + ": ");
23                 keuntunganPerusahaan[i][j] = sc.nextDouble();
24             }
25         }
26
27         // Hitung dan tampilkan total keuntungan perusahaan
28         for (int i = 0; i < numCompanies; i++) {
29             double totalKeuntungan = hitungTotalKeuntungan(keuntunganPerusahaan[i]);
30             System.out.println("Total keuntungan perusahaan " + (i+1) + " adalah: " + totalKeuntungan);
31         }
32
33         // Berikan output penghitungan total keuntungan perusahaan
34         printKeuntunganTotal(keuntunganPerusahaan);
35
36         // Hitung total keuntungan
37         double total = 0;
38         for (double k : keuntungan) {
39             total += k;
40         }
41         return total;
42     }
43 }
```

Hasil :

```
1
Masukkan jumlah bulan untuk perusahaan 1:
2
Masukkan keuntungan per bulan untuk perusahaan 1:
Masukkan keuntungan untuk bulan 1:
5.7
Masukkan keuntungan untuk bulan 2:
10
Total keuntungan perusahaan 1 adalah: 15.7
ASUS ▢ Prak 1F 03 ▢ @master ▢ ~1 ▢
```




NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : BRUTE FORCE DAN DIVIDE CONQUER

Latihan :

Mobildata03 :

```
1 package PS_Mobil;
2
3 public class MobilData03 {
4     String merk, tipe, tahun;
5     int top_acceleration, top_power;
6
7     public MobilData03(String merk, String tipe, String tahun, int top_acceleration, int top_power) {
8         this.merk = merk;
9         this.tipe = tipe;
10        this.tahun = tahun;
11        this.top_acceleration = top_acceleration;
12        this.top_power = top_power;
13    }
14
15    public static int CariTopOC(MobilData03[] arr, int l, int r) {
16        if (l == r)
17            return l;
18        int mid = (l + r) / 2;
19        int lmid = CariTopOC(arr, l, mid);
20        int rmid = CariTopOC(arr, mid + 1, r);
21        if (arr[lmid].top_acceleration < arr[rmid].top_acceleration) {
22            return rmid;
23        } else {
24            return lmid;
25        }
26    }
27
28    public static int CariTopRendahOC(MobilData03[] arr, int l, int r) {
29        if (l == r)
30            return l;
31        int mid = (l + r) / 2;
32        int lmid = CariTopRendahOC(arr, l, mid);
33        int rmid = CariTopRendahOC(arr, mid + 1, r);
34        if (arr[lmid].top_acceleration > arr[rmid].top_acceleration) {
35            return lmid;
36        } else {
37            return rmid;
38        }
39    }
40
41    public static double RataTopPower03(MobilData03[] arrMobil) {
42        int sum = 0;
43        for (int i = 0; i < arrMobil.length; i++) {
44            sum += arrMobil[i].top_power;
45        }
46        return (double) sum / arrMobil.length;
47    }
48 }
```

Maincars03 :

```
1 package PS_Mobil;
2 import java.util.Scanner;
3
4 public class MainCars03 {
5     public static void main(String[] args) {
6         MobilData03 mobil[] = new MobilData03[8];
7         mobil[0] = new MobilData03("Mercedes", "C200", "2016", 6816, 7780);
8         mobil[1] = new MobilData03("Ford", "Fiesta 3", "2014", 3921, 575);
9         mobil[2] = new MobilData03("Nissan", "Jetta", "2009", 4360, 657);
10        mobil[3] = new MobilData03("Subaru", "WRX", "2014", 4055, 609);
11        mobil[4] = new MobilData03("Subaru", "Impreza WRX STi", "2011", 6255, 783);
12        mobil[5] = new MobilData03("Toyota", "Auris", "2016", 3700, 553);
13        mobil[6] = new MobilData03("Toyota", "Prius", "2014", 4100, 609);
14        mobil[7] = new MobilData03("Volkswagen", "Golf GTI", "2014", 4100, 633);
15
16        System.out.println("Pilih opsi:");
17        System.out.println("1. Mobil dengan top acceleration tertinggi");
18        System.out.println("2. Mobil dengan top acceleration terendah");
19
20        Scanner scanner = new Scanner(System.in);
21        int pilihMobil = scanner.nextInt();
22
23        switch (pilihMobil) {
24            case 1:
25                int tinggi = MobilData03.CariTopOC(mobil, 0, mobil.length - 1);
26                System.out.println("Mobil dengan Top tertinggi : ");
27                System.out.println("Merek : " + mobil[tinggi].merk);
28                System.out.println("Tipe : " + mobil[tinggi].tipe);
29                System.out.println("Tahun : " + mobil[tinggi].tahun);
30                System.out.println("Top acceleration : " + mobil[tinggi].top_acceleration);
31                System.out.println("Top Power : " + mobil[tinggi].top_power);
32                System.out.println("Rata-Rata Top Power : " + MobilData03.RataTopPower03(mobil));
33                break;
34            case 2:
35                int rendah = MobilData03.CariTopRendahOC(mobil, 0, mobil.length - 1);
36                System.out.println("Mobil dengan Top terendah : ");
37                System.out.println("Merek : " + mobil[rendah].merk);
38                System.out.println("Tipe : " + mobil[rendah].tipe);
39                System.out.println("Tahun : " + mobil[rendah].tahun);
40                System.out.println("Top acceleration : " + mobil[rendah].top_acceleration);
41                System.out.println("Top Power : " + mobil[rendah].top_power);
42                System.out.println("Rata-Rata Top Power : " + MobilData03.RataTopPower03(mobil));
43                break;
44            default:
45                System.out.println("Pilihan tidak valid");
46        }
47        scanner.close();
48    }
49 }
```

Hasil :

```
Pilih opsi:
1. Mobil dengan top acceleration tertinggi
2. Mobil dengan top acceleration terendah
1
Mobil dengan Top tertinggi :
Merek :BMW
Tipe :M2 Coupe
Tahun :2016
top_acceleration :6816
top Power :7780
Rata-Rata Top Power : 633.125
ASUS  D Prk 1F 03  D Master = 71  3  1
```