



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

FILE NAME =ABSEN_NAME_CLASS_P1

* Pertemuan mengikuti pertemuan ke berapa

4.2 Percobaan 1: Menghitung Nilai Faktorial dengan Algoritma Brute Force dan Divide and Conquer

```
public class faktorial18 {  
    3 usages  
    public int nilai;  
    1 usage new *  
    int faktorialBF(int n) {  
        int fakto = 1;  
        for (int i = 1; i <= n; i++) {  
            fakto = fakto * i;  
        }  
        return fakto;  
    }  
    2 usages new *  
    int faktorialDC(int n){  
        if (n == 1) {  
            return 1;  
        } else {  
            int fakto = n * faktorialDC(n - 1);  
            return fakto;  
        }  
    }  
}
```

```
public static void main(String[] args) {  
    Scanner sc18 = new Scanner(System.in);  
    System.out.println("=====");  
    System.out.print("Masukkan jumlah elemen: ");  
    int iJml = sc18.nextInt();  
  
    faktorial18[] fk = new faktorial18[iJml];  
    for (int i = 0; i < iJml; i++) {  
        fk[i] = new faktorial18();  
        System.out.println("Masukkan nilai data ke-" + (i+1) + ": ");  
        fk[i].nilai = sc18.nextInt();  
    }  
    System.out.println("Hasil Brute Force: ");  
    for (int i = 0; i < iJml; i++) {  
        System.out.println("Hasil perhitungan menggunakan Brute Force adalah: " + fk[i].faktorialBF(fk[i].nilai));  
    }  
    System.out.println("Hasil Divide Conquer: ");  
    for (int i = 0; i < iJml; i++) {  
        System.out.println("Hasil perhitungan menggunakan Divide Conquer adalah: " + fk[i].faktorialDC(fk[i].nilai));  
    }  
}
```



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

```
=====
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 menggunakan Brute Force adalah: 120
Hasil perhitungan menggunakan Brute Force adalah: 40320
Hasil perhitungan menggunakan Brute Force adalah: 6
Hasil Divide Conquer:
Hasil perhitungan menggunakan Divide Conquer adalah: 120
Hasil perhitungan menggunakan Divide Conquer adalah: 40320
Hasil perhitungan menggunakan Divide Conquer adalah: 6
```

4.2.3 Pertanyaan

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

perbedaan antara menggunakan if dan else terletak pada cara pemrosesan dan pengelompokan langkah-langkah rekursif

- Penggunaan if dalam kode ini, akan melakukan pengecekan dalam kondisi tertentu untuk menentukan apakah akan melakukan rekursif atau mengembalikan nilai.
- Penggunaan else dalam kode ini, digunakan untuk menjalankan rekursif atau perintah lain ketika if tidak terpenuhi

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

Bisa, dengan menggunakan panggilan rekursif seperti faktorialDC()

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



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

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

fakto *= i;

Menggunakan operator *= untuk mengalikan nilai fakto dengan i.

int fakto = n * faktorialDC(n-1);:

Menggunakan rekursi untuk menghitung faktorial, menghitung faktorial dari n-1, dan mengalikan hasilnya dengan n.

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

```
public static void main(String[] args) {
    Scanner sc18 = new Scanner(System.in);
    System.out.println("=====");
    System.out.print("Masukkan elemen yang akan dihitung: ");
    int element = sc18.nextInt();
    pangkat18[] power = new pangkat18[element];
    for (int i = 0; i < element; i++) {
        power[i] = new pangkat18();
        System.out.print("Masukkan nilai yang akan dipangkatkan: ");
        power[i].nilai = sc18.nextInt();
        System.out.print("Masukkan pangkat: ");
        power[i].pangkat = sc18.nextInt();
    }

    System.out.println("Hasil Pangkat Brute Force: ");
    for (int i = 0; i < element; i++) {
        System.out.println("Hasil dari " + power[i].nilai + " pangkat " + power[i].pangkat + " adalah "
            + power[i].pangkatBF(power[i].nilai, power[i].pangkat));
    }
    System.out.println("Hasil Pangkat Divide and Conquer: ");
    for (int i = 0; i < element; i++) {
        System.out.println("Hasil dari " + power[i].nilai + " pangkat " + power[i].pangkat + " adalah "
            + power[i].pangkatDC(power[i].nilai, power[i].pangkat));
    }
}
```



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

```
public class pangkat18 {  
    5 usages  
    public int nilai, pangkat;  
  
    1 usage new *  
    int pangkatBF(int a, int n) {  
        int hasil = 1;  
        for (int i = 0; i < n; i++) {  
            hasil *= a;  
        }  
        return hasil;  
    }  
  
    2 usages new *  
    int pangkatDC(int a, int n) {  
        if (n==0) {  
            return 1;  
        } else {  
            int nilai = pangkatDC(a, n: n/2);  
            if (n%2==0) {  
                return nilai * nilai;  
            } else {  
                return nilai * nilai * a;  
            }  
        }  
    }  
}
```

```
=====  
Masukkan elemen yang akan dihitung: 2  
Masukkan nilai yang akan dipangkatkan: 6  
Masukkan pangkat: 2  
Masukkan nilai yang akan dipangkatkan: 4  
Masukkan pangkat: 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 6  
Hasil dari 4 pangkat 3 adalah 1
```

4.3.3 Pertanyaan

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

Pangkat BF()

Menggunakan operasi *= untuk mengalikan nilai hasil dengan a

PangkatDC()

Menggunakan rekursif nilai tersebut dibagi 2 dan disimpan menghitung pangkat dari masing-masing bagian secara rekursif, dan kemudian menggabungkan hasilnya.



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

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

Tahap "combine" terjadi saat kita menggabungkan hasil pangkat dari kedua bagian yang terbagi menjadi hasil akhir.

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

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

```
public pangkat18(int nilai, int pangkat){  
    this.nilai = nilai;  
    this.pangkat = pangkat;  
}  
  
power[i] = new pangkat18( nilai: 0, pangkat: 0);
```



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

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

```
=====
1. Brute Force
2. Divide Conquer
=====
Pilih: 1
Hasil Pangkat Brute Force:
Hasil dari 6 pangkat 2 adalah 36
Hasil dari 4 pangkat 3 adalah 64
```

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

```
int element;
4 usages
double keuntungan[], total;
1 usage new *
sum18( int element){
    this.element = element;
    this.keuntungan = new double[element];
    this.total = 0;
}
1 usage new *
double totalBF (double array[]){
    for (int i = 0; i < element; i++) {
        total = total + array[i];
    }
    return total;
}
```



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

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

```
Scanner sc18 = new Scanner(System.in);
System.out.println("=====");
System.out.println("Program menghitung keuntungan total (satuan jutam misal5,9)");
System.out.print("Masukkan jumlah bulan: ");
int jmlbln = sc18.nextInt();

sum18 sm = new sum18(jmlbln);
System.out.println("=====");
for (int i = 0; i < sm.element; i++) {
    System.out.print("Masukkan untung bulan ke - " + (i+1) + ": ");
    sm.keuntungan[i] = sc18.nextDouble();
}
System.out.println("=====");
System.out.println("Algoritma Brute Force");
System.out.println("Total keuntungan perusahaan selama " + sm.element + " bulan adalah: " + sm.totalBF(sm.keuntungan));
System.out.println("=====");
System.out.println("Algoritma Divide and Conquer");
System.out.println("Total keuntungan perusahaan selama " + sm.element + " bulan adalah: " + sm.totalDC(sm.keuntungan, 0, sm.element-1));
```

```
Program menghitung keuntungan total (satuan jutam misal5,9)
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: 40.339999999999996

Process finished with exit code 0
```



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

Pertanyaan

1. Mengapa terdapat formulasi return value berikut?Jelaskan!

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

Untuk mengembalikan value hasil dari penjumlahan tersebut

2. Kenapa dibutuhkan variable mid pada method TotalDC()?
Untuk membagi array menjadi dua bagian yang lebih kecil, menentukan batas indeks untuk rekursi, dan mengakses nilai dari elemen tengah array.
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!

```
public static void main(String[] args) {  
    Scanner sc18 = new Scanner(System.in);  
    System.out.println("=====");  
    System.out.println("Program menghitung keuntungan total (satuan jutam misal5  
    System.out.print("Masukkan berapa perusahaan: ");  
    int jmlperusahaan = sc18.nextInt();  
  
    int[] jmlblnPerusahaan = new int[jmlperusahaan];  
    double[] totalPerusahaanBF = new double[jmlperusahaan];  
    double[] totalPerusahaanDC = new double[jmlperusahaan];  
    sum18 sm = null;  
    System.out.println("=====");  
    for (int i = 0; i < jmlperusahaan; i++) {  
        System.out.println("Perusahaan ke -" + (i + 1));  
        System.out.print("Masukkan jumlah bulan: ");  
        jmlblnPerusahaan[i] = sc18.nextInt();  
        sm = new sum18(jmlblnPerusahaan[i]);  
        for (int j = 0; j < sm.element; j++) {  
            System.out.print("Masukkan untung bulan ke - " + (j + 1) + ": ");
```




NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

```
sm.keuntungan[i] = sc.nextInt();  
}  
totalPerusahaanBF[i] = sm.totalBF(sm.keuntungan);  
totalPerusahaanDC[i] = sm.totalDC(sm.keuntungan, l: 0, r: sm.element - 1);  
  
system.out.println("=====");  
for (int i = 0; i < jmlPerusahaan; i++) {  
    System.out.println("Perusahaan ke -" + (i + 1));  
    System.out.println("Algoritma Brute Force");  
    System.out.println("Total keuntungan perusahaan selama " + jmlBlnPerusahaan[i] + " bulan adalah: " + totalPerusahaanBF[i]);  
    System.out.println("=====");  
    System.out.println("Algoritma Divide and Conquer");  
    System.out.println("Total keuntungan perusahaan selama " + jmlBlnPerusahaan[i] + " bulan adalah: " + totalPerusahaanDC[i]);  
    System.out.println("=====");  
}
```

```
Masukkan berapa perusahaan: 2  
=====  
Perusahaan ke -1  
Masukkan jumlah bulan: 2  
Masukkan untung bulan ke - 1: 4  
Masukkan untung bulan ke - 2: 5  
Perusahaan ke -2  
Masukkan jumlah bulan: 2  
Masukkan untung bulan ke - 1: 6  
Masukkan untung bulan ke - 2: 7  
=====  
Perusahaan ke -1  
Algoritma Brute Force  
Total keuntungan perusahaan selama 2 bulan adalah: 9.0  
=====  
Algoritma Divide and Conquer  
Total keuntungan perusahaan selama 2 bulan adalah: 9.0  
=====  
Perusahaan ke -2  
Algoritma Brute Force  
Total keuntungan perusahaan selama 2 bulan adalah: 13.0  
=====  
Algoritma Divide and Conquer  
Total keuntungan perusahaan selama 2 bulan adalah: 13.0  
=====
```

4.5 Latihan Praktikum



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

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:

```
1 usage new *
2 public mobil18(int[] acc, double[] pwr) {
3     this.acc = acc;
4     this.pwr = pwr;
5 }
6
7 1 usage new *
8 public int maxAccell(int[] acc) {
9     return maxAcc(acc, 0, acc.length - 1);
10 }
11
12 3 usages new *
13 public int maxAcc(int[] acc, int l, int r) {
14     if (l == r) {
15         return acc[l];
16     }
17     int mid = (l + r) / 2;
18     int maxAccl = maxAcc(acc, l, mid);
19     int maxAccr = maxAcc(acc, mid + 1, r);
20     return Math.max(maxAccl, maxAccr);
21 }
```

a.



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

```
1 usage new *  
✓ public int minAccl(int[] acc) {  
✓     return minAcc(acc, l: 0, r: acc.length - 1);  
    }  
  
3 usages new *  
✓ public int minAcc(int[] acc, int l, int r) {  
✓     if (l == r) {  
        return acc[l];  
    }  
    int mid = (l + r) / 2;  
    int minAccl = minAcc(acc, l, mid);  
    int minAccr = minAcc(acc, l: mid + 1, r);  
  
    return Math.min(minAccl, minAccr);  
    }  
b.
```



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

```
usage new *  
  
public double avgpwr(double[] pwr) {  
    double penjumlahan = 0;  
    int pembagian = 0;  
    for (int i = 0; i < pwr.length; i++) {  
        penjumlahan += pwr[i];  
        pembagian++;  
    }  
    if (pembagian != 0) {  
        return penjumlahan / pembagian;  
    } else {  
        return 0.0;  
    }  
}
```

c.

```
public static void main(String[] args) {  
    int[] acc = {6816, 3921, 4360, 4058, 6255, 3700, 4180, 4180};  
    double[] pwr = {728, 575, 657, 609, 703, 553, 609, 631};  
  
    mobil18 mobil = new mobil18(acc, pwr);  
    int max = mobil.maxAcell(acc);  
    int min = mobil.minAcell(acc);  
    double avg = mobil.avgpwr(pwr);  
  
    System.out.println("Akselerasi mobil tertinggi adalah: " + max);  
    System.out.println("Akselerasi mobil terendah adalah: " + min);  
    System.out.println("Rata-rata tenaga mobil adalah: "+ avg);  
}
```

output:



NAMA : Muhammad Khalid Atthoriq
NIM : 2341720114
NO ABSEN : 18
KELAS : 1C
MATERI : Array of Object

```
Akselerasi mobil tertinggi adalah: 6816  
Akselerasi mobil terendah adalah: 3700  
Rata-rata tenaga mobil adalah: 633.125
```