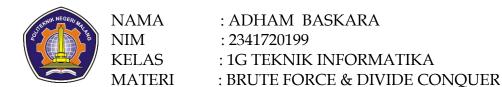
4.2 Menghitung Nilai Faktorial dengan Algoritma Brute Force dan Divide and Conquer

```
package P5.BruteForceDivideConquer.src.minggu5;

Comment Code | Improve Code
public class Faktorial {
    public int nilai;

    public int faktorialBF(int n){
        int fakto = 1;
        for(int i=1; i<= n; i++){
            | fakto = fakto*i;
        }
        return fakto;
    }

    public int faktorialDC(int n){
        if (n==1){
            return 1;
        }
        else{
            int fakto = n*faktorialDC(n-1);
            return fakto;
        }
    }
}</pre>
```



4.2.2 Verifikasi Hasil Percobaan

```
Masukkan jumlah elemen:
3
Masukkan nilai data ke-1:
5
Masukkan nilai data ke-2:
8
Masukkan nilai data ke-3:
3
HASIL - ERUTE FORCE
Hasil perhitungan faktorial menggunakan Brute Force adalah120
Hasil perhitungan faktorial menggunakan Brute Force adalah40320
Hasil perhitungan faktorial menggunakan Brute Force adalah40320
Hasil perhitungan faktorial menggunakan Brute Force adalah40820
Hasil perhitungan faktorial menggunakan Brute Force adalah6
HASIL - DIVIDE AND CONQUER
Hasil perhitungan faktorial menggunakan Divide and Conquer adalah 40320
Hasil perhitungan faktorial menggunakan Divide and Conquer adalah 40320
Hasil perhitungan faktorial menggunakan Divide and Conquer adalah 6
PS D:\SEMESTER 2\Praktikum Algoritma\left(a Algoritma_6 StrukturData_02> []
```

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!

Jawab: Singkatnya, if digunakan untuk menemukan base case dan menghentikan rekursi, sementara else melakukan langkah rekursif untuk menghitung faktorial.

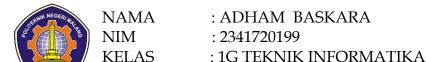
2. Apakah memungkinkan perulangan pada method faktorialBF() diubah selain menggunakan for? Buktikan!

Jawab:

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

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

Jawab: yg pertama menggunakan loop, sedangkan yg kedua menggunakan pendekatan rekursif utk membagi menjadi masalah kecil-kecil



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

```
package P5.BruteForceDivideConquer.src.minggu5;

CommentCode|ImproweCode
public class Pangkat {
   public int nilai, pangkat;

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

public int pangkatDC(int a, int n){
   if (n==0){
      return 1;
   }else{
      if(n%2==1)//bitangan ganjit
      return(pangkatDC(a,n/2)*pangkatDC(a,n/2)*a);
      else//bitangan genap
      return(pangkatDC(a,n/2)*pangkatDC(a,n/2));
   }
}</pre>
```

4.3.2 Verifikasi Hasil Percobaan

```
Masukkan nilai yang akan dipangkatkan ke-1 : 6
Masukkan nilai pemankat ke-1 : 2
Masukkan nilai pemankat ke-1 : 2
Masukkan nilai pemankat ke-2 : 4
Masukkan nilai pemankat ke-2 : 4
Masukkan nilai pemankat ke-2 : 3
HASIL PANRAT - ERUIF FURGE
Nilai 6 pangkat 2 adalah 36
Nilai 4 pangkat 3 adalah 36
HASIL PANRAT - DIVIDE CONQUER
Nilai 6 pangkat 2 adalah 36
Nilai 6 pangkat 2 adalah 36
Nilai 6 pangkat 2 adalah 36
Nilai 6 pangkat 3 adalah 36
```

: BRUTE FORCE & DIVIDE CONQUER

4.3.3 Pertanyaan

MATERI

1. Jelaskan mengenai perbedaan 2 method yang dibuat yaitu PangkatBF() dan PangkatDC()! *Jawab:* yg pertama menggunakan brute force, sedangkan yg kedua menggunakan divide conquer

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

Jawab:

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

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

Jawab:

```
package PS. BruteforceOrdadeConquer.src.minggud;

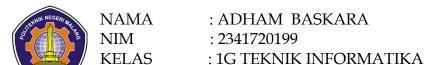
Import JavantilaGenoms;

Commentoes Import code
public Cast Parghat (
    public fast files, parghat;

// fonstruntor untuk forsionises ricked dan pangket
public Parghat (first files, first pangkat) (
    this public fast files, first pangkat) (
    this public first pangkat) (
    int teap = pangkat) (
    int teap = pangkat) (
    int pang
```

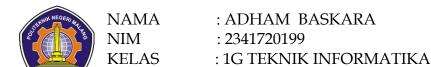
```
The most of seas will interaction

Comment Column and C
```



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

Jawab:



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

```
package P5.BruteforceDivideConquer.src.minggu5;

CommantCode|ImproxeCode
public class Sum {
    int elemen;
    double keuntungan[], total;

public Sum(int elemen){
        this.elemen = elemen;
        this.elemen = elemen;
        this.total = 0;

}

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

public double totalDC(double arr[], int 1, int r){
    if(l = r){
        int mid = (1 + r) / 2;
        double !sum + totalDC(arr, l, mid);
        double rsum + totalDC(arr, mid + 1, r);
        return lsum + rsum;
    }
    return 0;
}</pre>
```

```
package PS.BruteForceDivideConquer.src.singgus;

import java.util.Scanner;

CommentCosi | Import Cost |

Upublic class MainSum (

Upublic class MainSum (

Upublic static void main(String[] args) (

Scanner sc = new Scanner(System.in);

System.out.println(sc**newsem.emphitume keuntungan total (Satuan juta, misal 5.9)*);

System.out.println(sc**newsem.emphitume very season selama * + sm.elemen + *bulan adalah = * + sm.total0f(sm.keuntungan, li0, sm.elemen - 1));

System.out.println(sc**lagoritam Divide and conquer**);

System.out.println(sc**lagoritam Divide and conquer**);
```

4.4.2 Verifikasi Hasil Percobaan

```
Program memoghitump keuntungan total (Satuan juta, misal 5.9)

Masabkan jumlah bulan:

5

8.5

Masabkan untung bulan ke - 1=
8.5

Masabkan untung bulan ke - 2=
9.54

Masabkan untung bulan ke - 3=
7.2

Masabkan untung bulan ke - 4=
9.1

Masabkan untung bulan ke - 5=
6

Masabkan untung bulan ke - 5=
6

Company bulan ke - 5=
6

Masabkan untung bulan ke - 5=
6

Total keuntungan perusahaan selama Sulan adalah = 40.3399999999996

PS D:\SEMESTER 2\Fraktikum Algoritum/16_Algoritum_2_StruktunData_02
```



: ADHAM BASKARA

: 2341720199

: 1G TEKNIK INFORMATIKA

: BRUTE FORCE & DIVIDE CONQUER

4.4.3 Pertanyaan

1. Mengapa terdapat formulasi return value berikut? Jelaskan!

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

Jawab: menggabungkan keuntungan total dari dua bagian array yang dibagi secara rekursif (bagian kiri dan kanan)

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

Jawab: digunakan utk menentukan titik tengah saat membagi array menjadi 2 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:



NAMA : ADHAM BASKARA

: 2341720199

: 1G TEKNIK INFORMATIKA

MATERI: BRUTE FORCE & DIVIDE CONQUER