Data Structure and Algorithm Practicum
Brute Force and Divide Conquer

Name
Muhammad Baihaqi Aulia Asy'ari

NIM
2241720145

Class
1I

Department
Information Technology

Study Program
D4 Informatics Engineering

2.3 Calculating Factorial Values with Brute Force and Divide and Conquer Algorithms

2.3.1 Practicum

1. Create a new Project, with the name AlgoStruDat / Project Name Equated to last week. Make a package with the name **Week3**, make a new class with the name **Faktorial**.

2. Complete the Faktorial class with the attributes and methods described in the class diagram above:

   (a) Add value attributes

   ```
   public int num;
   ```

   (b) Add method `faktorialBF()`

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

   (c) Add method `faktorialDC()`

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

3. Run the `Faktorial class` y creating a new `MainFaktorial` class.

   (a) In the main function, provide input to input the number of numbers to find the factorial value

   ```
   Scanner sc = new Scanner(System.in);
   System.out.println("================================
    ↪  ================================");
   ```

```
System.out.print("Input the number of elements you want to
↪  count : ");
int elemen = sc.nextInt();
```

(b) Create an Array of Objects on the main function, then input some values that will be factorially calculated

```
Faktorial [] fk = new Faktorial[elemen];
for (int i = 0; i < elemen; i++) {
    fk[i] = new Factorial();
    System.out.print("Input the data value to-"+(i+1)+" : ");
    fk[i].num = sc.nextInt();
}
```

(c) Display the results of calling method `faktorialDC()` dan `faktorialBF()`

```
System.out.println("=================================
↪  ================================");
System.out.println("Factorial Result with Brute Force");
for (int i = 0; i < elemen; i++) {
    System.out.println("Factorial of value"+fk[i].num+" is :
     ↪  "+fk[i].faktorialBF(fk[i].num));
}

System.out.println("=================================
↪  ================================");
for (int i = 0; i < elemen; i++) {
    System.out.println("Factorial of value"+fk[i].num+" is :
     ↪  "+fk[i].faktorialDC(fk[i].num));
}
System.out.println("=================================
↪  ================================");
```

(d) Make sure the program is running well!

1. `Faktorial.java`

```
package Faktorial;

public class Faktorial {
    public int num;
    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;
        }
    }
}
```

2. MainFaktorial.java

```
package Faktorial;

public class MainFaktorial {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
            System.out.println("==================================
            ↪  ================================");
            System.out.print("Input the number of elements you
            ↪  want to count : ");
            int elemen = sc.nextInt();
            Faktorial [] fk = new Faktorial[elemen];
            for (int i = 0; i < elemen; i++) {
                fk[i] = new Factorial();
                System.out.print("Input the data value
                ↪  to-"+(i+1)+" : ");
                fk[i].num = sc.nextInt();
            }
            System.out.println("==================================
            ↪  ================================");
            System.out.println("Factorial Result with Brute
            ↪  Force");
            for (int i = 0; i < elemen; i++) {
                System.out.println("Factorial of value
                ↪  "+fk[i].num+" is :
                ↪  "+fk[i].faktorialBF(fk[i].num));
            }
```

```
                System.out.println("================================
                 ↪  ================================");
                for (int i = 0; i < elemen; i++) {
                    System.out.println("Factorial of value
                     ↪  "+fk[i].num+" is :
                     ↪  "+fk[i].faktorialDC(fk[i].num));
                }
                System.out.println("================================
                 ↪  ================================");
        }
    }
```

2.3.2 Verification of Practicum Results

```
================================================================
Input the number of elements you want to count : 3
Input the data value to-1 : 5
Input the data value to-2 : 8
Input the data value to-3 : 3
================================================================
Factorial Results with Brute Force
Factorial of value 5 : 120
Factorial of value 8 : 40320
Factorial of value 3 : 6
================================================================
Factorial Results with Divide and Conquer
Factorial of value 5 : 120
Factorial of value 8 : 40320
Factorial of value 3 : 6
================================================================
```

```
   Result:
```

```
PS D:\Kuliah>  d:; cd 'd:\Kuliah'; & 'C:\Program
 ↪  Files\Java\jdk-18.0.2.1\bin\java.exe'
 ↪  '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
 ↪  'C:\Users\ASUS\AppData\Roaming\Code\User\workspaceStorage\
 ↪  ce3fcb236261368a6cbd019dc8ddda8b\redhat.java\jdt_ws\
 ↪  Kuliah_28156aa7\bin' 'Faktorial.MainFaktorial'
================================================================
Input the number of elements you want to count : 3
Input the data value to-1 : 5
Input the data value to-2 : 8
Input the data value to-3 : 3
================================================================
Factorial Result with Brute Force
Factorial of value 5 is : 120
Factorial of value 8 is : 40320
Factorial of value 3 is : 6
================================================================
Factorial of value 5 is : 120
Factorial of value 8 is : 40320
Factorial of value 3 is : 6
================================================================
```

2.3.3 Questions

1. Explain the Divide Conquer Algorithm for calculating factorial values!
   with the
2. In the implementation of Factorial Divide and Conquer Algorithm is it complete that consists of 3 stages of divide, conquer, combine? Explain each part of the program code!
3. Is it possible to repeat the factorial BF () method instead of using for? Prove it!
4. Add a check to the execution time of the two types of methods!
5. Prove by inputting elements that are above 20 digits, is there a difference in execution time?