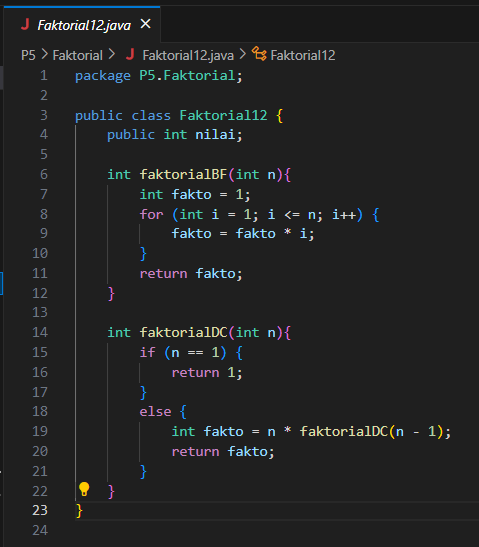
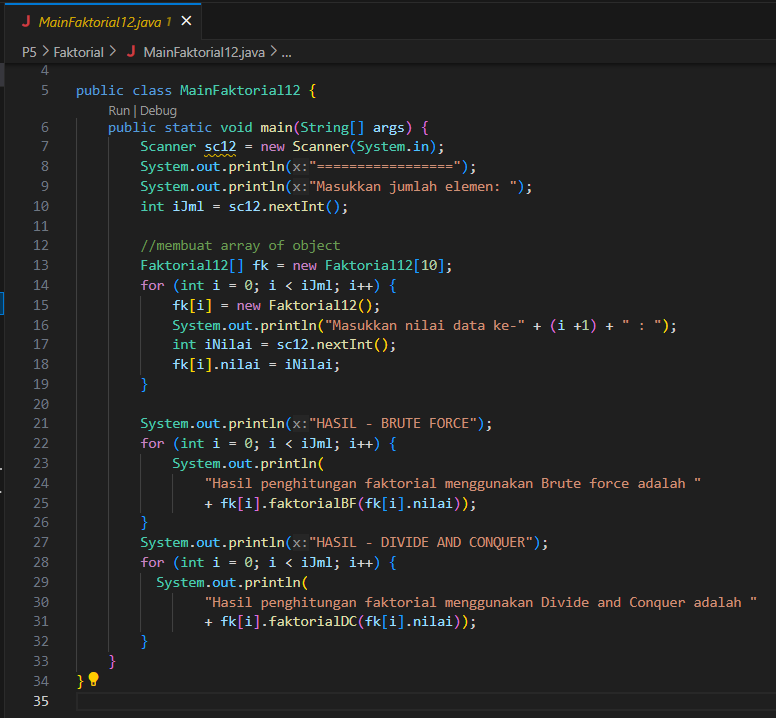
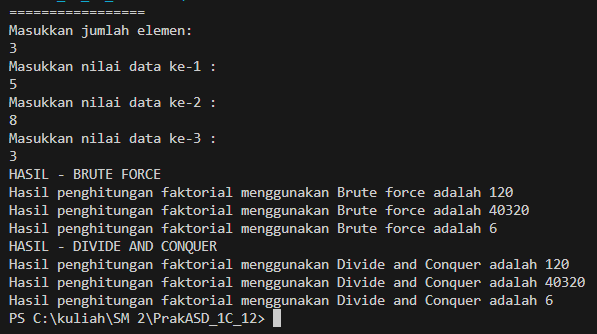
**4.1. Menghitung Nilai Faktorial dengan Algoritma Brute Force dan Divide and Conquer**

** ****

****

**Question :**

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

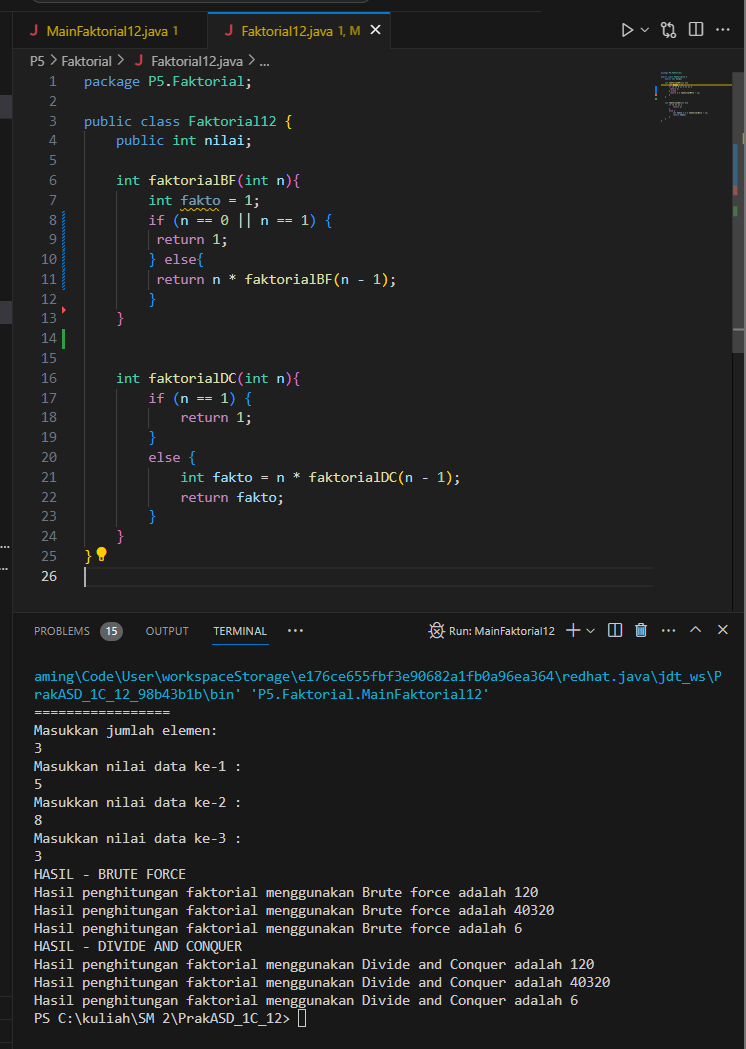
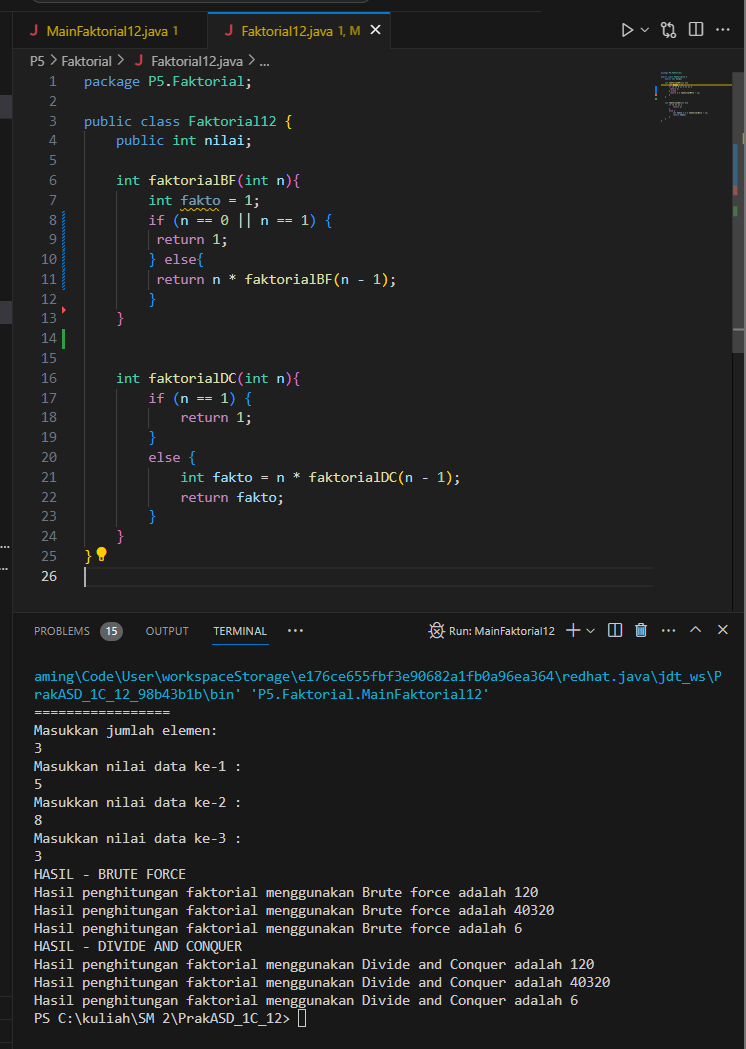
Jawab :

Pada “If” dilakukan pengecekan apakah n == 1, jika terpenuhi akan mengembalikan nilai 1.

Pada “else” dilakukan ketika kondisi if tidak terpenuhi dan pada “else” melakukan perhitungan nilai faktorial dari n dikalikan dengan nilai faktorial dari n – 1, kemudian dilakukan sampai mencapai n == 1

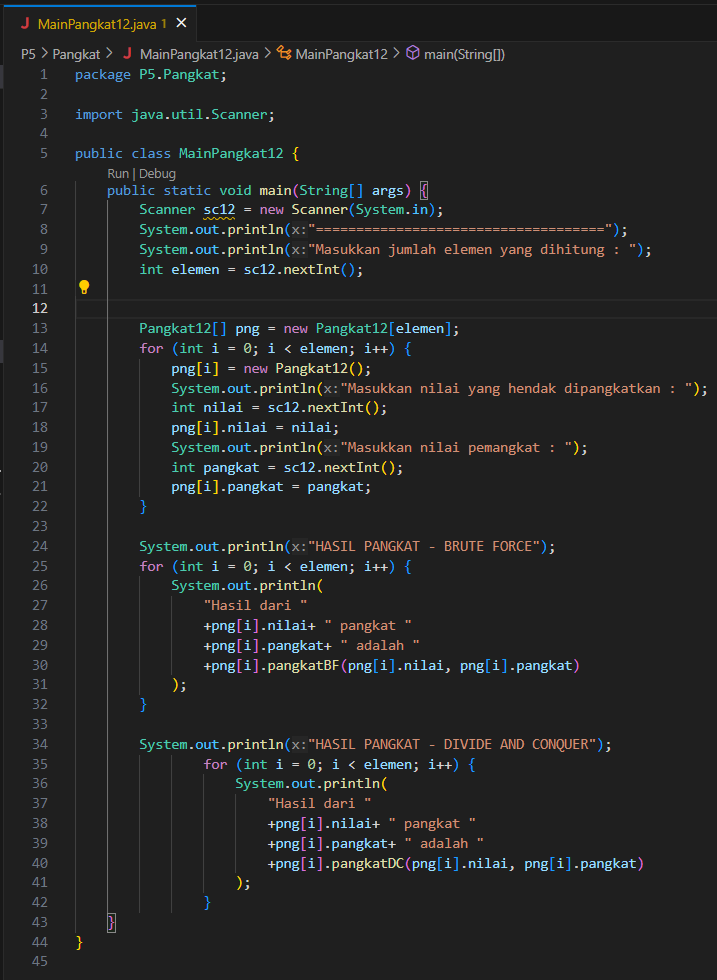
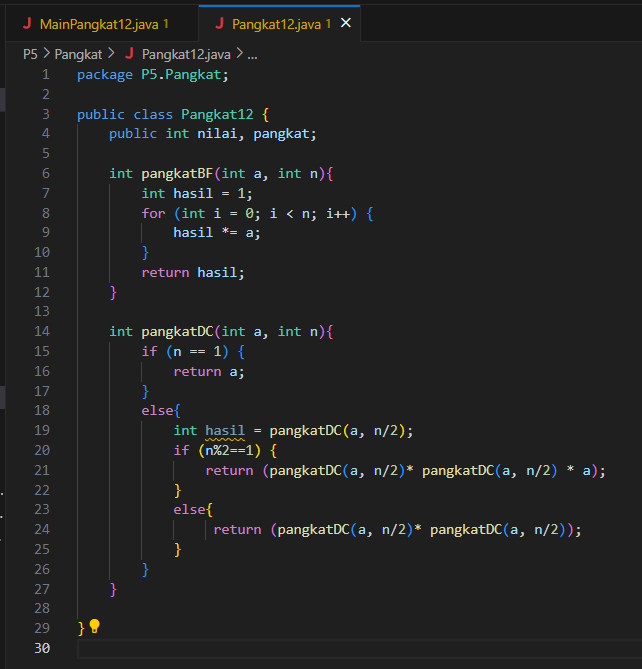
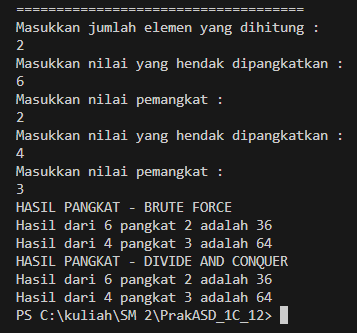
1. Apakah memungkinkan perulangan pada method faktorialBF() dirubah selain menggunakan for?Buktikan!

Jawab : memungkinkan,

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

**4.2. Menghitung Hasil Pangkat dengan Algoritma Brute Force dan Divide and Conquer**

**** * *

**Question :**

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

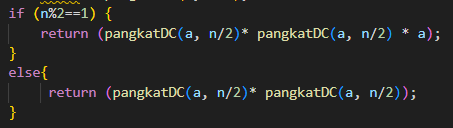
Jawab :

Pada **PangkatBF(**) menggunakan perulangan for, setiap perulangan “hasil” diperbarui dengan mengalikan “hasil” dengan “a” sebanyak n kali. a digunakan sebagai parameter untuk bilangan yang dipangkatkan, sedangkan n digunakan sebagai pangkat

Pada **PangkatDC()** menggunakan rekursif, memecah masalah menjadi submasalah yang lebih kecil. Dengan dilakukan pengecekan apakah “n==1” jika true, maka akan dikembalikan nilai “a”, kemudian jika n bukan 1 maka dibagi menjadi submasalah yang lebih kecil dengan memanggil “hasil”, jika pangkat ganjil maka “hasil” dikali dengan dirinya sendiri lalu dikali dengan “a”, jika pangkat genap maka “hasil” akan dikalikan dengan dirinya sendiri

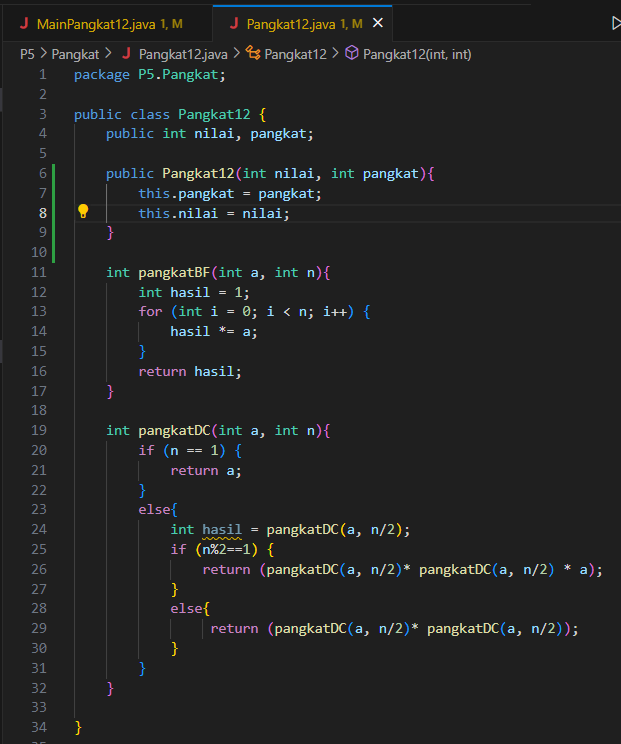
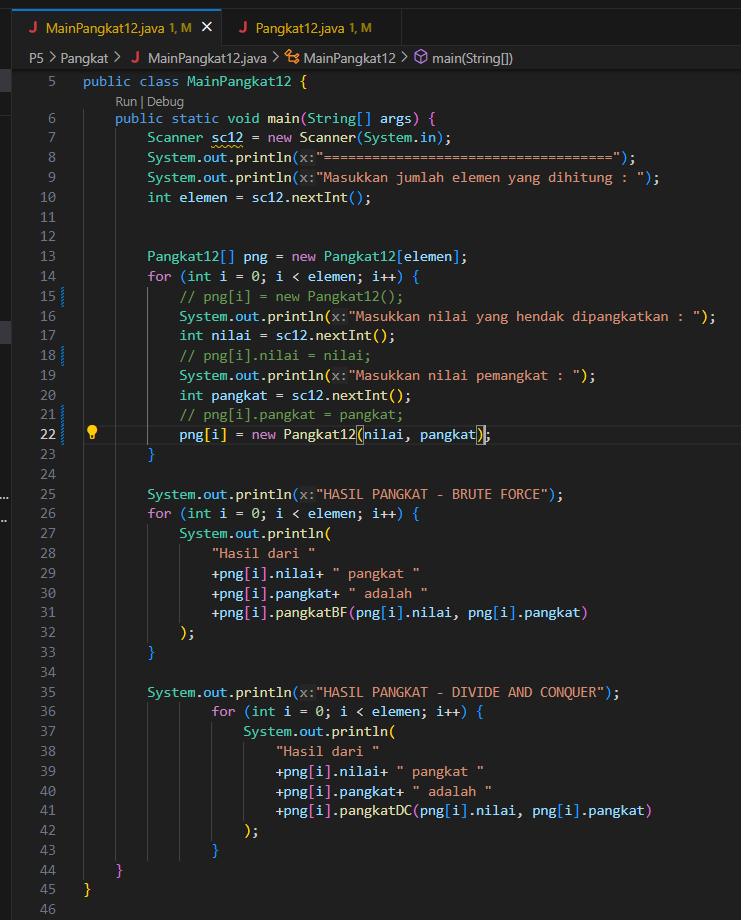
1. Apakah tahap combine sudah termasuk dalam kode tersebut?Tunjukkan!

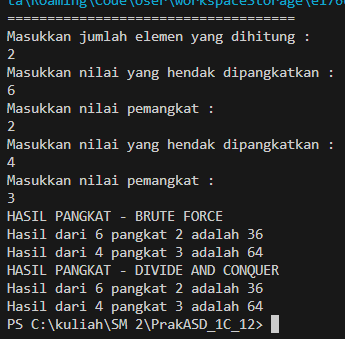
Jawab : sudah,



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

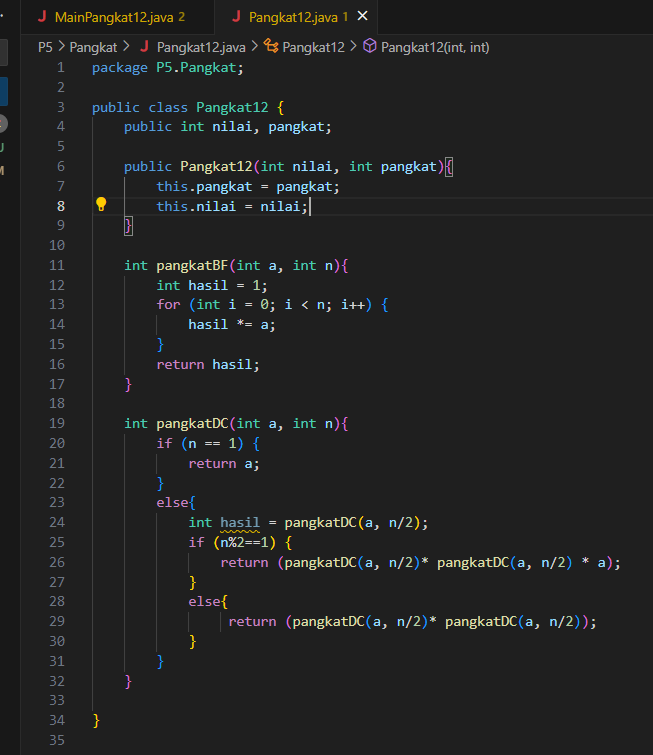
Jawab :

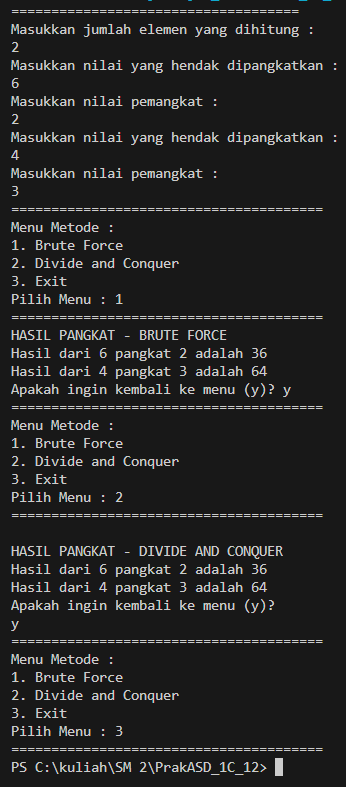
 



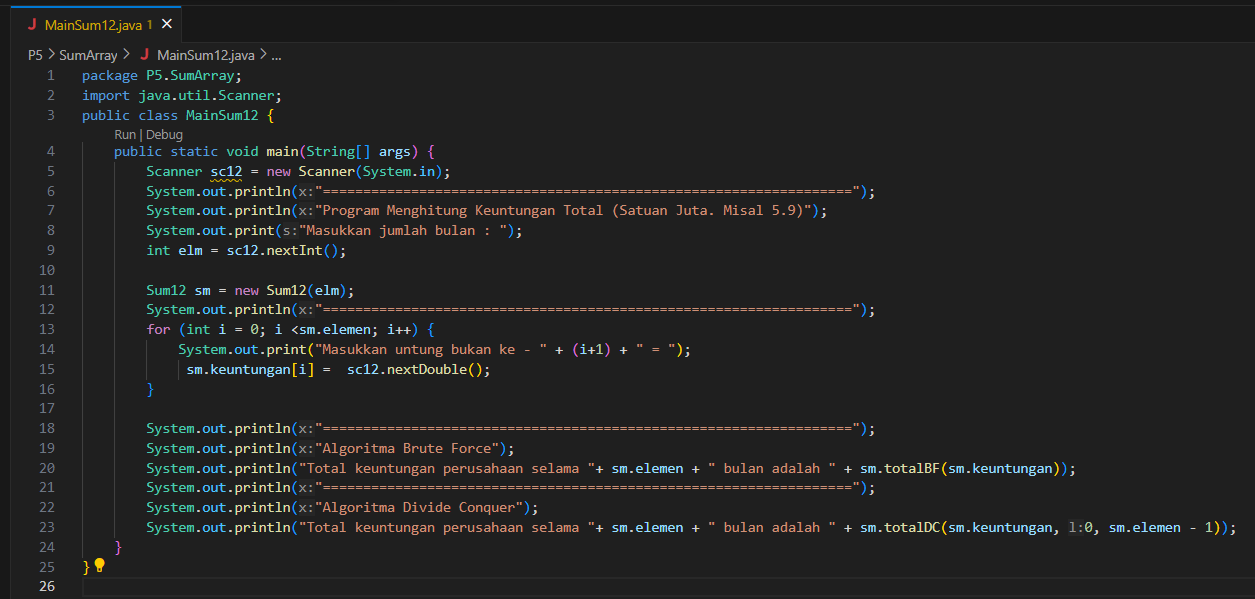
1. Tambahkan menu agar salah satu method yang terpilih saja yang akan dijalankan menggunakan switch-case!

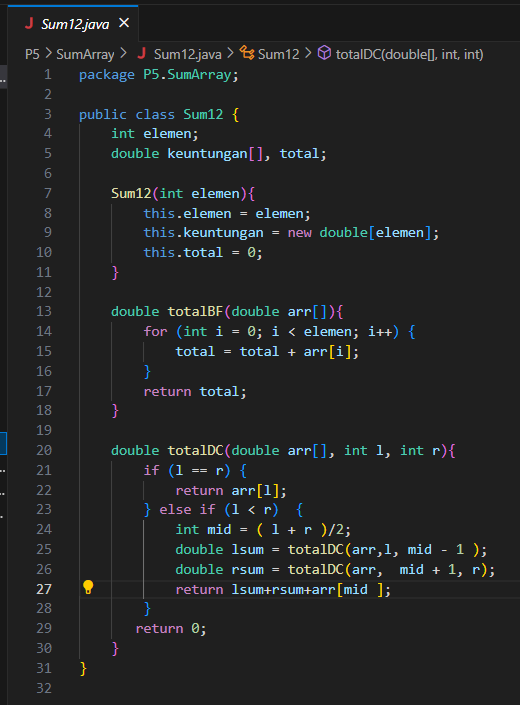
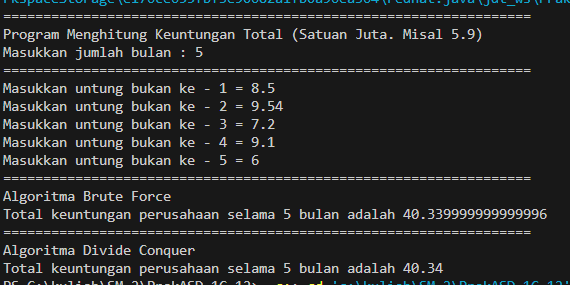
Jawab :



**4.3. Menghitung Sum Array dengan Algoritma Brute Force dan Divide and Conquer**

****

** **

**Question :**

1. Mengapa terdapat formulasi return value berikut?Jelaskan!



Jawab :

Karena return value digunakan untuk menggabungkan hasil dari penjumlahan lsum (elemen bagian kiri) , rsum (elemen bagian kanan), arr [mid] (elemen Tengah) agar menghasilkan hasil akhir

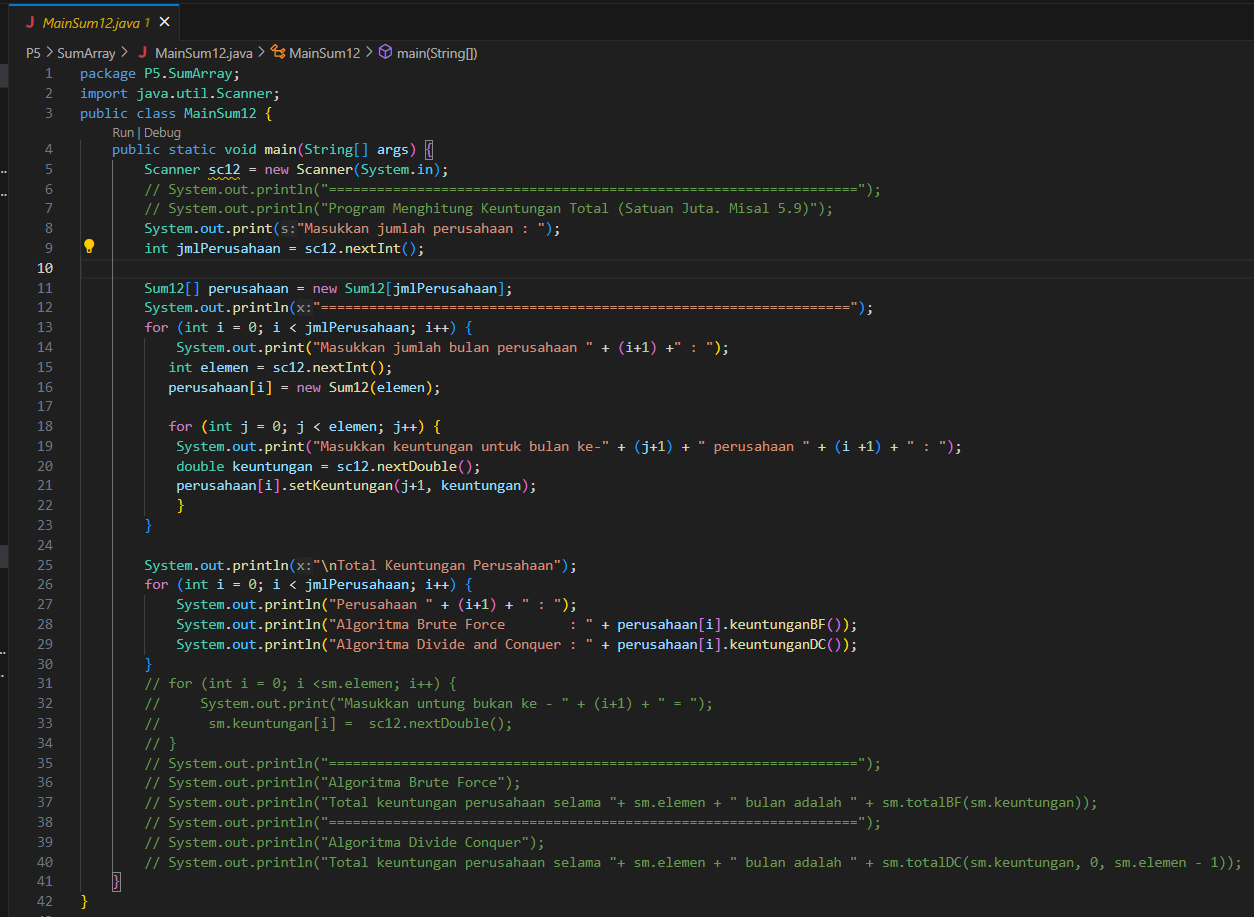
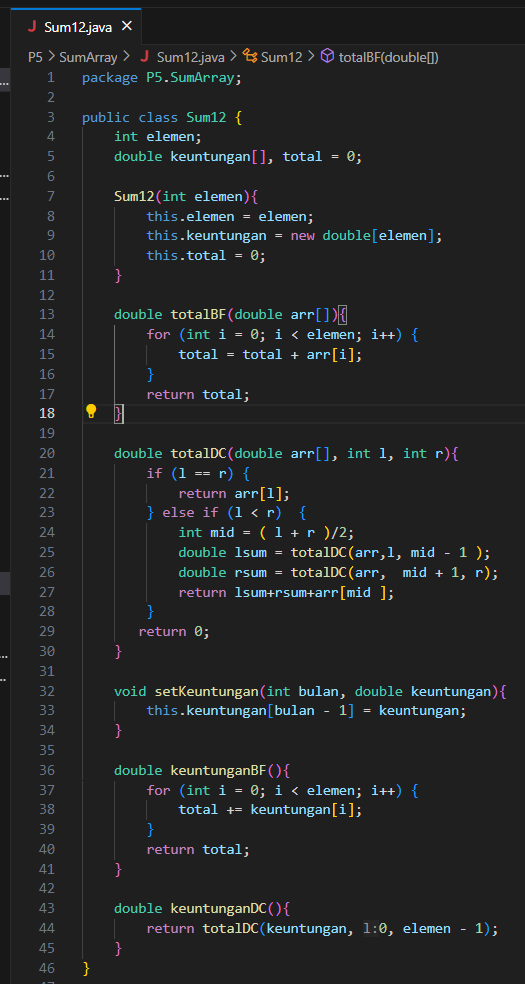
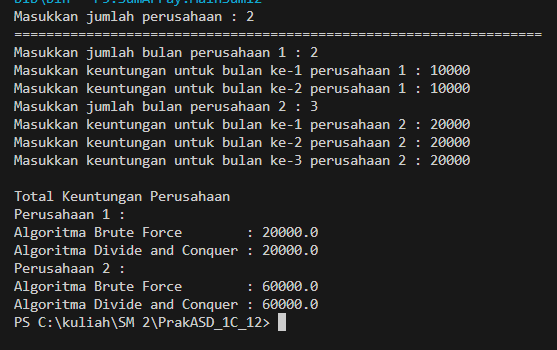
1. Kenapa dibutuhkan variable mid pada method TotalDC()?

Jawab :

Karena digunakan untuk menghitung total dari elemen array sebagai index tengah daan menggabungkan total dari bagian kanan dan kiri array

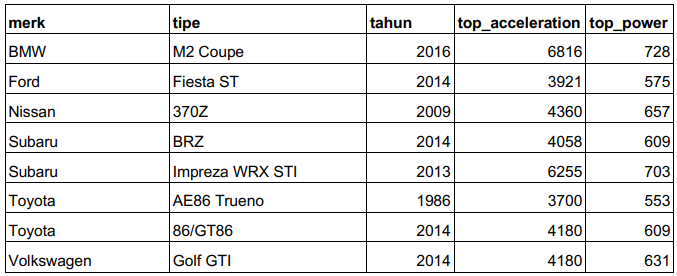
1. Program perhitungan keuntungan suatu perusahaan ini hanya untuk satu perusahaan saja. Bagaimana cara menghitung sekaligus keuntungan beberapa bulan untuk beberapa perusahaan.(Setiap perusahaan bisa saja memiliki jumlah bulan berbeda-beda)? Buktikan dengan program!

Jawab :

**4.5. Latihan Praktikum**

1. Sebuah showroom memiliki daftar mobil dengan data sesuai tabel di bawah ini



Tentukan:

1. top\_acceleration tertinggi menggunakan Divide and Conquer!
2. top\_acceleration terendah menggunakan Divide and Conquer!
3. Rata-rata top\_power dari seluruh mobil menggunakan Brute Force!

