**LAPORAN PRAKTIKUM**

**“POST TEST 5: FUNGSI DAN KOMPOSISI DUA FUNGSI”**

Diajukan untuk memenuhi salah satu praktikum Mata Kuliah Matematika Diskrit yang di ampu oleh:

 Nur Rochmah Dyah PA, S.T., M.Kom

Disusun Oleh:

Mohammad Farid Hendianto 2200018401

Selasa 12.00-13.30

**PROGRAM STUDI INFORMATIKA**

**UNIVERSITAS AHMAD DAHLAN**

**FAKULTAS TEKNOLOGI INDUSTRI**

**TAHUN 2023**

DAFTAR SOAL

[Buatlah Program Menentukan Faktorial dari sebuah inputan, dengan ketentuan : 3](#_Toc133957957)

# Buatlah Program Menentukan Faktorial dari sebuah inputan, dengan ketentuan :

● Program bersifat dinamis

● Menggunakan Perulangan Iteratif dan Rekursif dalam proses faktorial

● User bisa memlih dalam program nantinya mau menggunakan faktorial mode Iteratif atau Rekursif

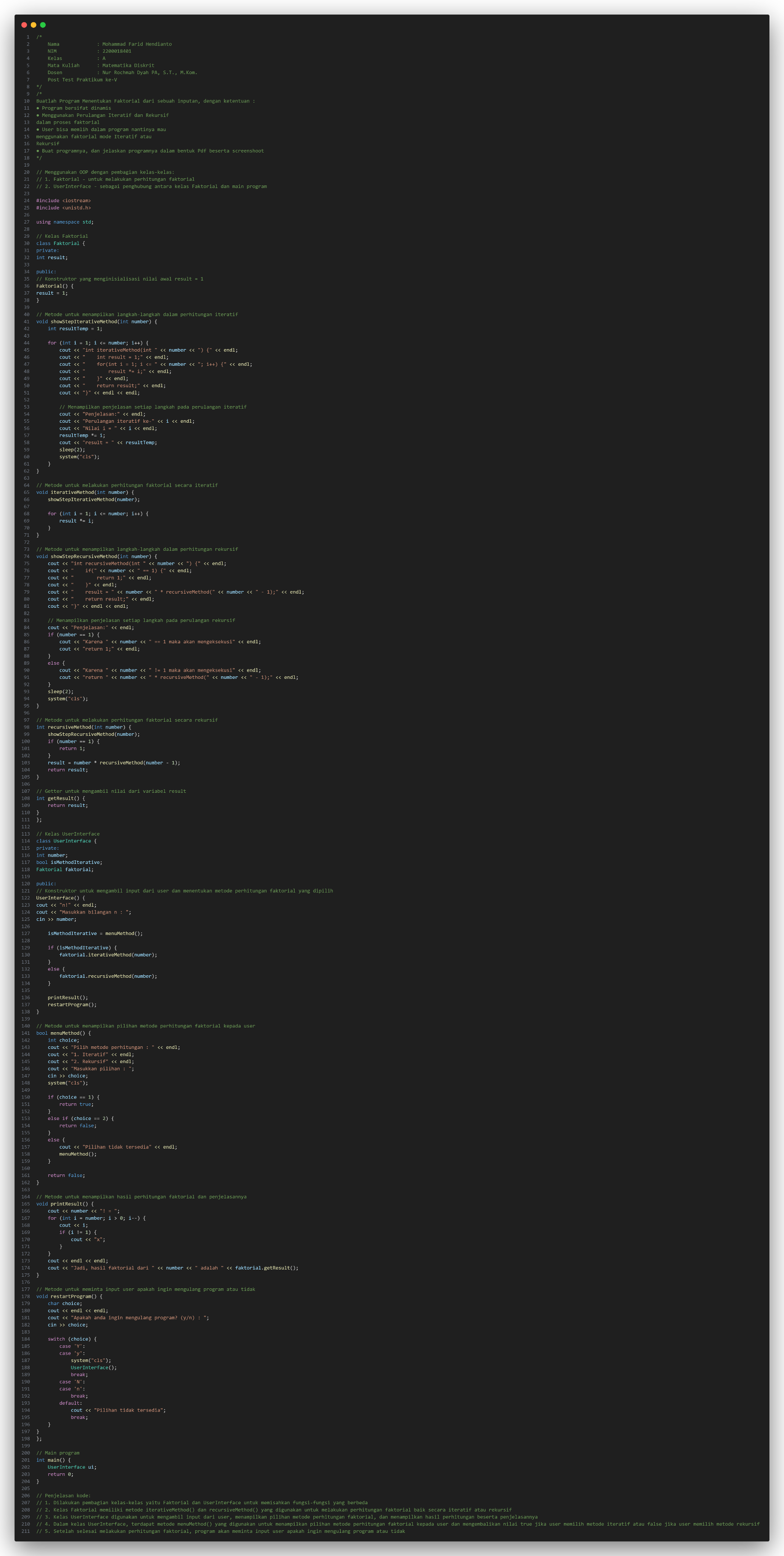
● Buat programnya, dan jelaskan programnya dalam bentuk Pdf beserta screenshoot

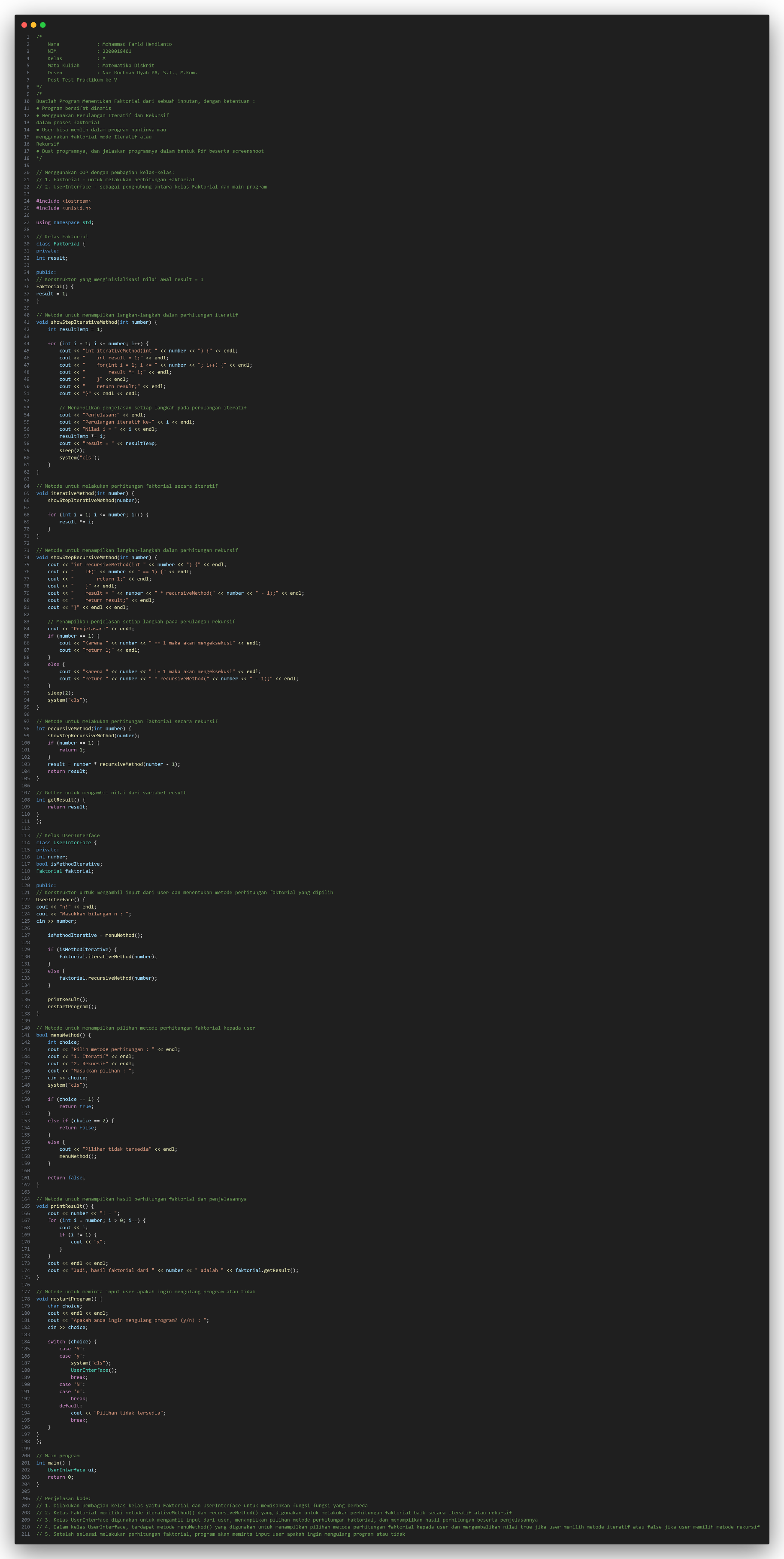
Jawab:

Untuk mengakses source code, dapat melihat link berikut ini

<https://github.com/IRedDragonICY/Matematika-Diskrit>

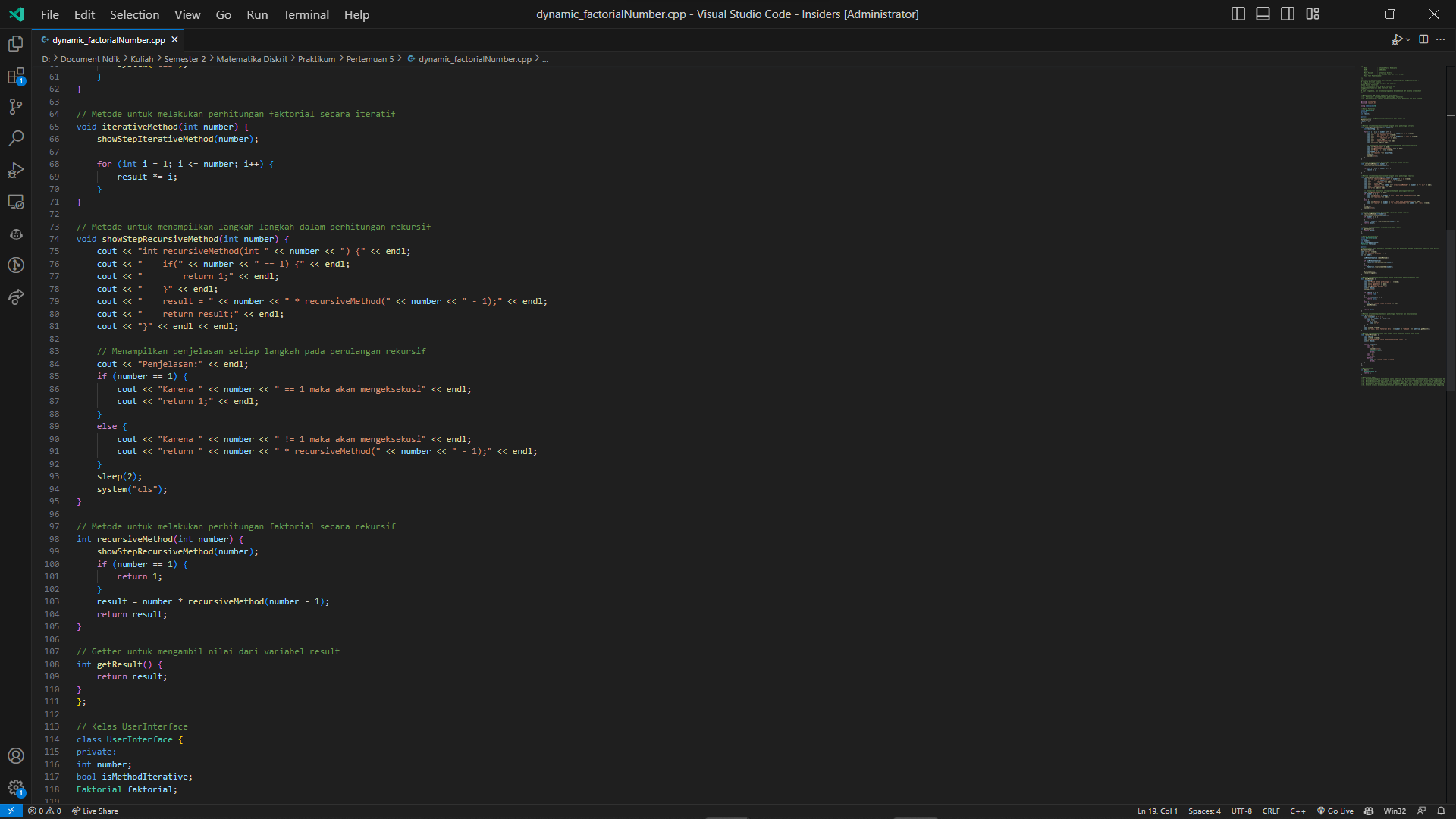
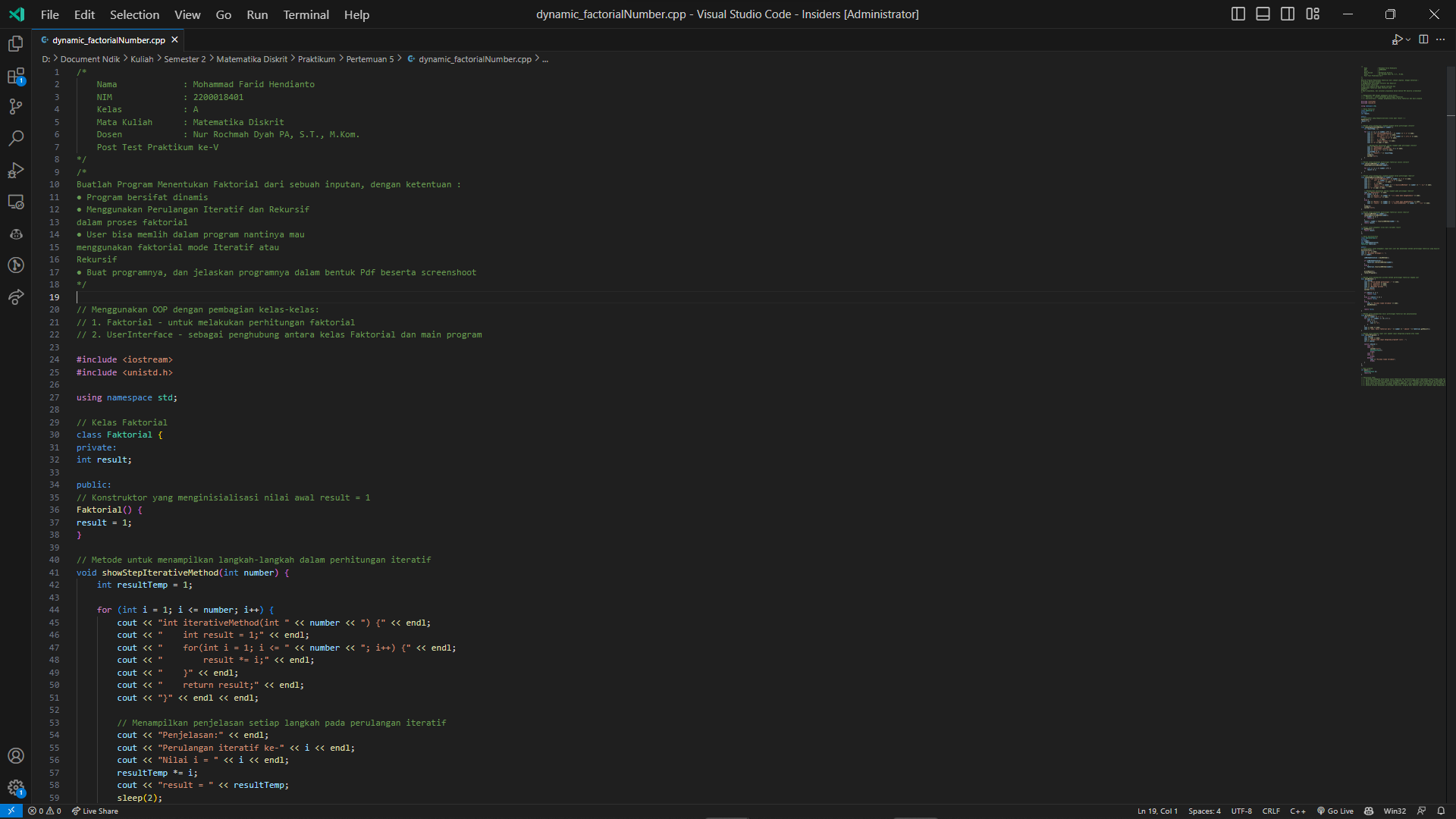


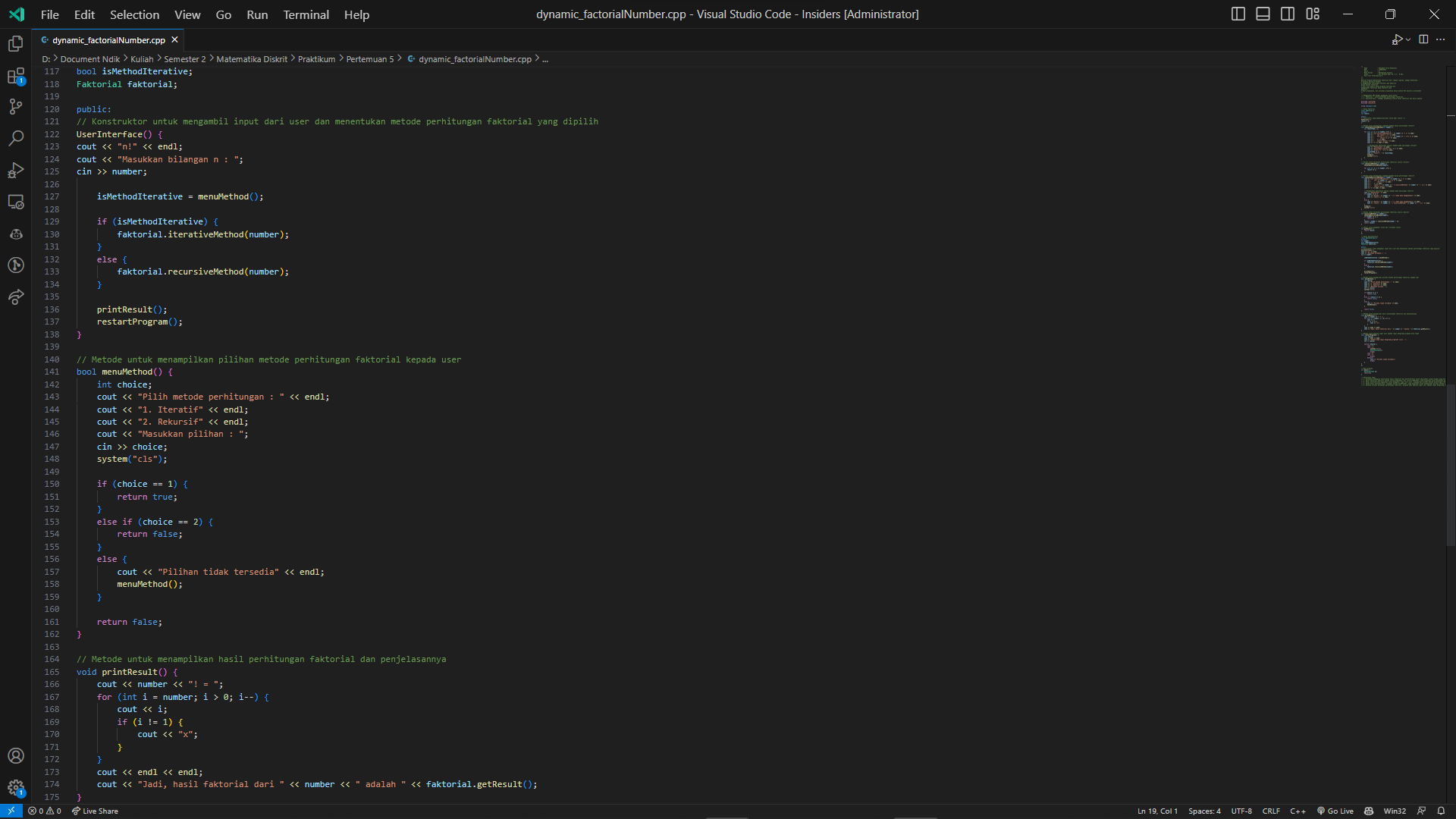


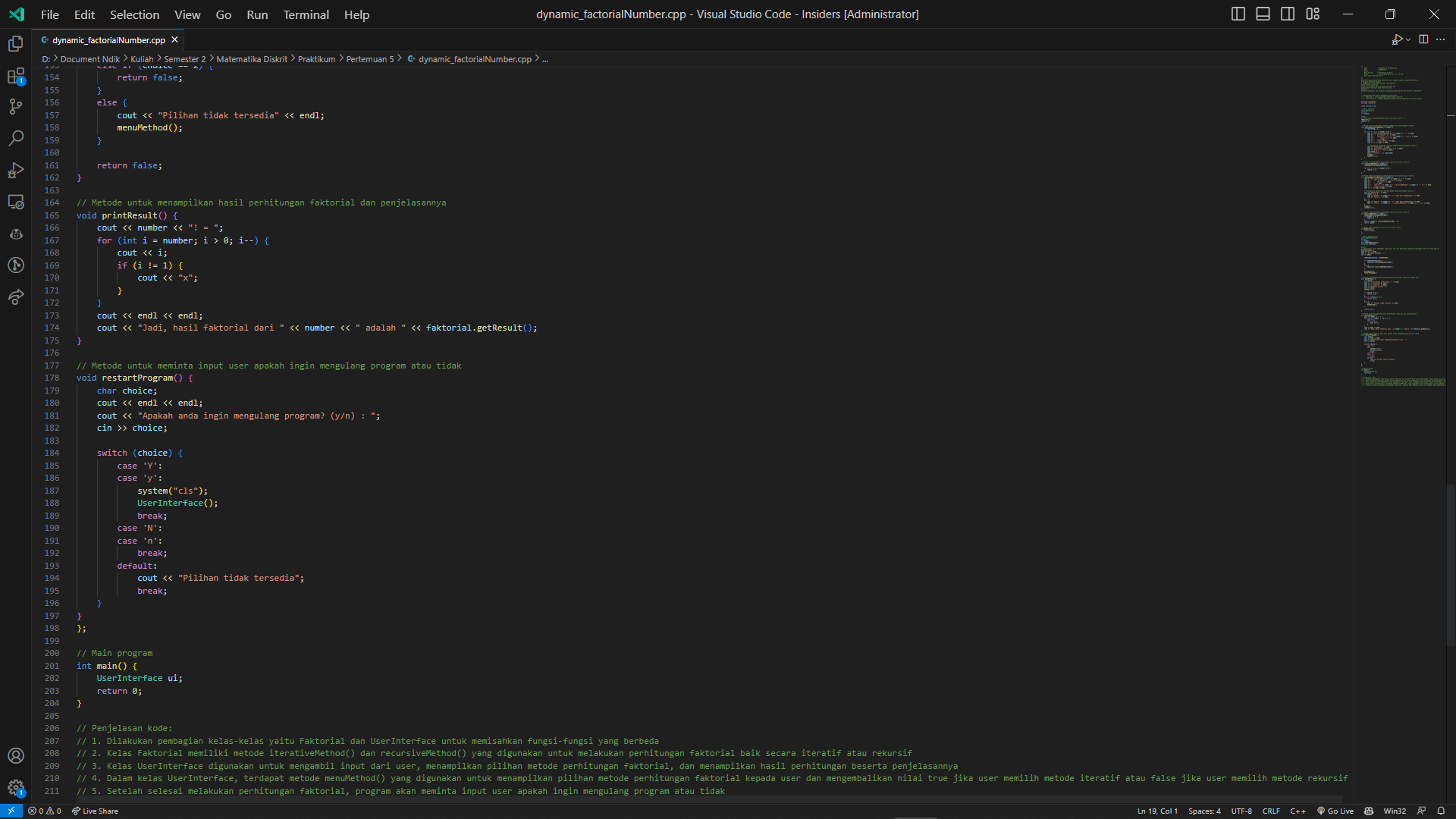


Gambar 1 Tampilan full source code. (Sumber: Penulis)

Berikut adalah tampilan kodingan di visual studio code:







Gambar 2 Tampilan kodingan yang berada di Visual Studio Code. (Sumber: penulis)

Program di atas adalah program sederhana untuk menghitung faktorial suatu bilangan dengan menggunakan perulangan iteratif atau rekursif. Program ini dibagi menjadi dua kelas yaitu Faktorial dan UserInterface.

Kelas Faktorial memiliki tiga metode yaitu iterativeMethod(), recursiveMethod(), dan getResult(). Metode iterativeMethod() digunakan untuk melakukan perhitungan faktorial secara iteratif, sedangkan metode recursiveMethod() digunakan untuk melakukan perhitungan faktorial secara rekursif. Metode getResult() digunakan untuk mengambil nilai dari variabel result yang merupakan hasil dari perhitungan faktorial.

Kelas UserInterface digunakan untuk menghubungkan antara kelas Faktorial dengan main program. Kelas ini memiliki empat metode yaitu menuMethod(), printResult(), restartProgram(), dan konstruktor. Metode menuMethod() digunakan untuk menampilkan pilihan metode perhitungan faktorial kepada user dan mengembalikan nilai true jika user memilih metode iteratif atau false jika user memilih metode rekursif. Metode printResult() digunakan untuk menampilkan hasil perhitungan faktorial beserta penjelasannya ke layar. Metode restartProgram() digunakan untuk meminta input user apakah ingin mengulang program atau tidak. Konstruktor digunakan untuk mengambil input dari user, menentukan metode perhitungan faktorial yang dipilih, dan memanggil metode iterativeMethod() atau recursiveMethod() dari kelas Faktorial tergantung pada pilihan user.

Kelebihan dari program ini adalah dapat menghitung faktorial baik secara iteratif maupun rekursif sehingga user dapat memilih metode yang paling sesuai untuk digunakan. Selain itu, program ini bersifat dinamis karena user dapat mengulang program jika diinginkan. Program ini juga memiliki tampilan yang cukup intuitif dan mudah dipahami oleh user.

Namun, program ini memiliki kekurangan yaitu tidak dilakukan validasi input dari user sehingga program dapat mengalami error jika user memasukkan input yang tidak sesuai. Selain itu, penjelasan langkah-langkah perhitungan hanya ditampilkan dalam bentuk teks dan tidak ada visualisasi grafis atau animasi sehingga mungkin sulit dipahami oleh beberapa user.

Program di atas bersifat dinamis karena program ini memberikan fleksibilitas kepada user untuk memilih metode perhitungan faktorial yang diinginkan, yaitu menggunakan perulangan iteratif atau rekursif. Pengguna dapat memilih salah satu metode sesuai dengan kebutuhan dan preferensi mereka. Selain itu, program ini juga dapat diulang jika pengguna menginginkannya.

Contoh dari kelebihan dinamis pada program ini terlihat pada metode menuMethod() yang dapat menampilkan pilihan metode perhitungan faktorial kepada user dan mengembalikan nilai true jika user memilih metode iteratif atau false jika user memilih metode rekursif. Hal ini memberikan fleksibilitas kepada pengguna untuk memilih metode yang paling sesuai dengan kebutuhan mereka. Selain itu, program ini juga memiliki kemampuan untuk diulang jika pengguna menginginkannya, sehingga pengguna dapat melakukan perhitungan faktorial berulang-ulang tanpa harus memulai ulang program.

Salah satu kutipan kodenya adalah sebagai berikut:



Gambar 3 Dua metode untuk mencari faktorial, iteratif atau rekursif. (Sumber: Penulis)

Program yang Anda berikan memungkinkan pengguna untuk memilih metode perhitungan faktorial yang ingin digunakan, baik itu iteratif atau rekursif. Setelah pengguna memilih metode, program akan melakukan perhitungan sesuai dengan pilihan pengguna.

Ada fungsi menuMethod, program akan menampilkan pilihan metode perhitungan faktorial pada layar dan meminta pengguna untuk memilih salah satu metode. Jika pengguna memilih iteratif (pilihan nomor 1), maka program akan mengembalikan nilai true. Jika pengguna memilih rekursif (pilihan nomor 2), maka program akan mengembalikan nilai false. Jika pengguna memasukkan pilihan yang tidak tersedia, maka program akan menampilkan pesan kesalahan dan memanggil kembali fungsi menuMethod.

Setelah pengguna memilih metode perhitungan faktorial, program akan memanggil metode calculateFactorial dengan parameter bool isIterative yang menentukan apakah metode perhitungan yang digunakan adalah iteratif atau rekursif:

Kelebihan dari program ini adalah dapat menghitung faktorial baik secara iteratif maupun rekursif sehingga user dapat memilih metode yang paling sesuai untuk digunakan. Selain itu, program ini bersifat dinamis karena user dapat mengulang program jika diinginkan dan program memberikan opsi kepada pengguna untuk memilih jenis perhitungan faktorial yang ingin digunakan. Namun, program ini memiliki kekurangan seperti tidak melakukan validasi input dari pengguna.

Program ini untuk menentukan faktorial dari sebuah inputan dengan memilih antara metode perhitungan iteratif atau rekursif. Program ini menggunakan konsep Object Oriented Programming (OOP) dengan membagi program menjadi 2 kelas yaitu Faktorial dan UserInterface. Kelas Faktorial berisi metode iterativeMethod() dan recursiveMethod() yang digunakan untuk melakukan perhitungan faktorial secara iteratif dan rekursif, sedangkan kelas UserInterface digunakan untuk mengambil input dari user, menampilkan pilihan metode perhitungan faktorial, dan menampilkan hasil perhitungan beserta penjelasannya.

Perulangan iteratif digunakan pada metode iterativeMethod() untuk melakukan perhitungan faktorial. Dalam perulangan ini, program akan mengalikan angka dengan result dimulai dari 1 hingga mencapai inputan yang diinginkan. Setiap langkah pada perulangan juga ditampilkan melalui showStepIterativeMethod() untuk memberikan pemahaman yang lebih baik kepada pengguna tentang bagaimana program melakukan perhitungan faktorial secara iteratif.

Rekursi digunakan pada metode recursiveMethod() untuk melakukan perhitungan faktorial. Dalam rekursi, program akan mengalikan angka dengan hasil rekursif dari angka -1 hingga mencapai inputan yang diinginkan. Setiap langkah pada rekursi juga ditampilkan melalui showStepRecursiveMethod() untuk memberikan pemahaman yang lebih baik kepada pengguna tentang bagaimana program melakukan perhitungan faktorial secara rekursif.

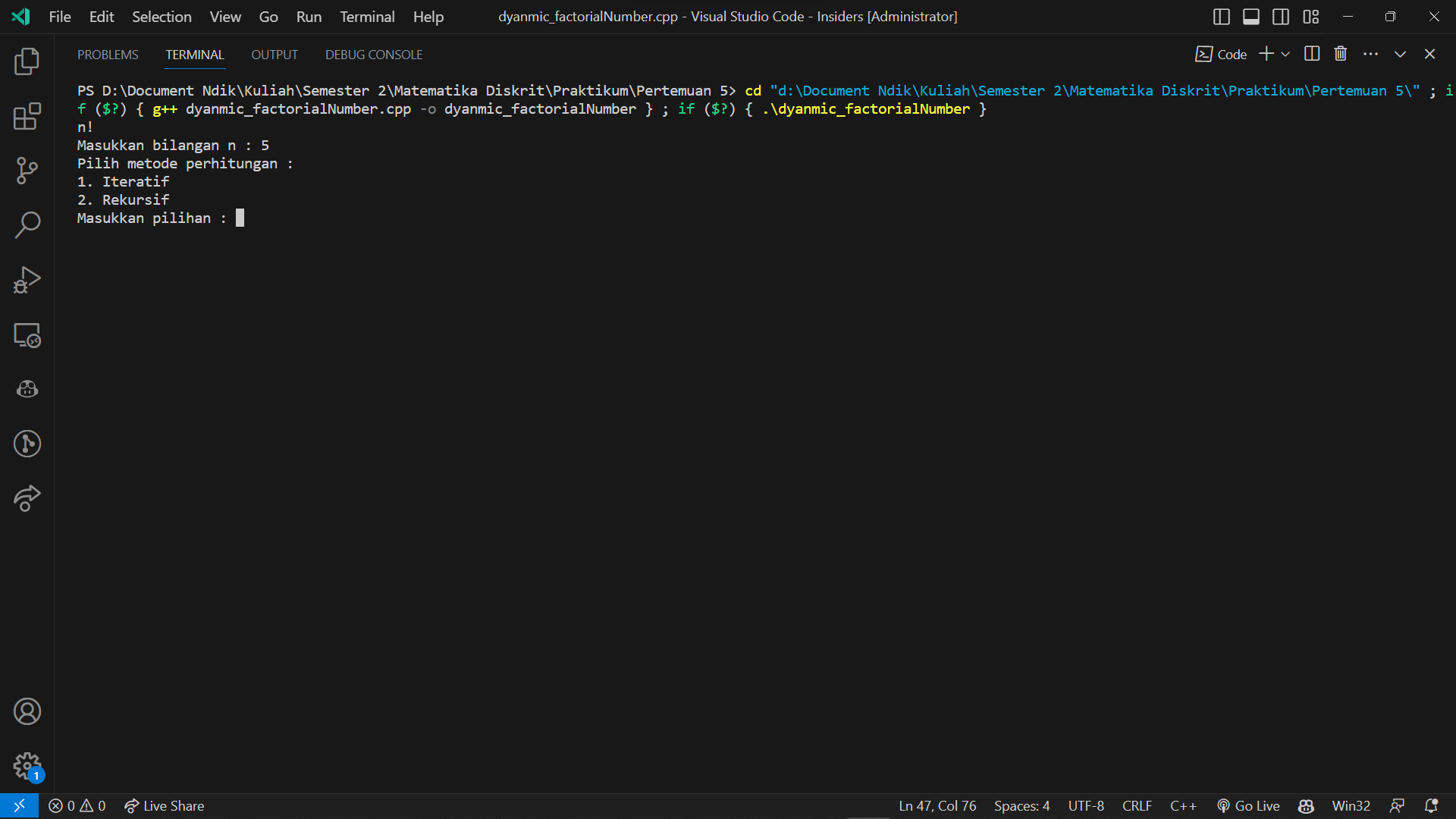
Berikut adalah alur programnya

1. Program dimulai dengan mengimport library yang dibutuhkan, yaitu iostream dan unistd.h.
2. Kemudian, dibuat dua kelas yaitu Faktorial dan UserInterface untuk memisahkan fungsi-fungsi yang berbeda dari program.
3. Kelas Faktorial memiliki tiga metode yaitu showStepIterativeMethod(), iterativeMethod(), dan recursiveMethod().
4. Metode showStepIterativeMethod() digunakan untuk menampilkan langkah-langkah pada perhitungan faktorial secara iteratif.
5. Metode iterativeMethod() digunakan untuk melakukan perhitungan faktorial secara iteratif dengan menggunakan perulangan for dan memanggil metode showStepIterativeMethod() untuk menampilkan langkah-langkah dalam perhitungan iteratif.
6. Metode showStepRecursiveMethod() digunakan untuk menampilkan langkah-langkah pada perhitungan faktorial secara rekursif.
7. Metode recursiveMethod() digunakan untuk melakukan perhitungan faktorial secara rekursif dengan menggunakan fungsi rekursif dan memanggil metode showStepRecursiveMethod() untuk menampilkan langkah-langkah dalam perhitungan rekursif.
8. Kelas UserInterface memiliki empat metode yaitu menuMethod(), printResult(), restartProgram(), dan konstruktor.
9. Metode menuMethod() digunakan untuk menampilkan pilihan metode perhitungan faktorial kepada user dan mengembalikan nilai true jika user memilih metode iteratif atau false jika user memilih metode rekursif.
10. Metode printResult() digunakan untuk menampilkan hasil perhitungan faktorial beserta penjelasannya.
11. Metode restartProgram() digunakan untuk meminta input user apakah ingin mengulang program atau tidak.
12. Konstruktor UserInterface() digunakan untuk memanggil metode-menu method(), dan melakukan perhitungan faktorial sesuai dengan metode yang dipilih oleh user.
13. Setelah selesai melakukan perhitungan faktorial, program akan meminta input user apakah ingin mengulang program atau tidak dengan memanggil metode restartProgram().
14. Program dimulai dengan membuat objek dari kelas UserInterface dan menjalankan konstruktor.
15. Pada konstruktor UserInterface(), program akan menampilkan inputan "n!" untuk meminta user untuk memasukkan bilangan n yang akan dihitung faktorialnya.
16. Kemudian, program akan menampilkan pilihan metode perhitungan faktorial yaitu iteratif atau rekursif melalui metode menuMethod() yang dipanggil pada konstruktor.
17. User memilih metode perhitungan faktorial dan program akan memanggil metode iterativeMethod() atau recursiveMethod() dari objek kelas Faktorial sesuai dengan pilihan yang dilakukan user.
18. Setelah perhitungan faktorial selesai, program akan menampilkan hasil perhitungan beserta penjelasannya dengan memanggil metode printResult().
19. Selanjutnya, program akan meminta input user apakah ingin mengulang program atau tidak dengan memanggil metode restartProgram().
20. Jika user memilih untuk mengulang program, program akan restart dan meminta user memasukkan input lagi. Jika user memilih untuk tidak mengulang program, program akan berakhir.

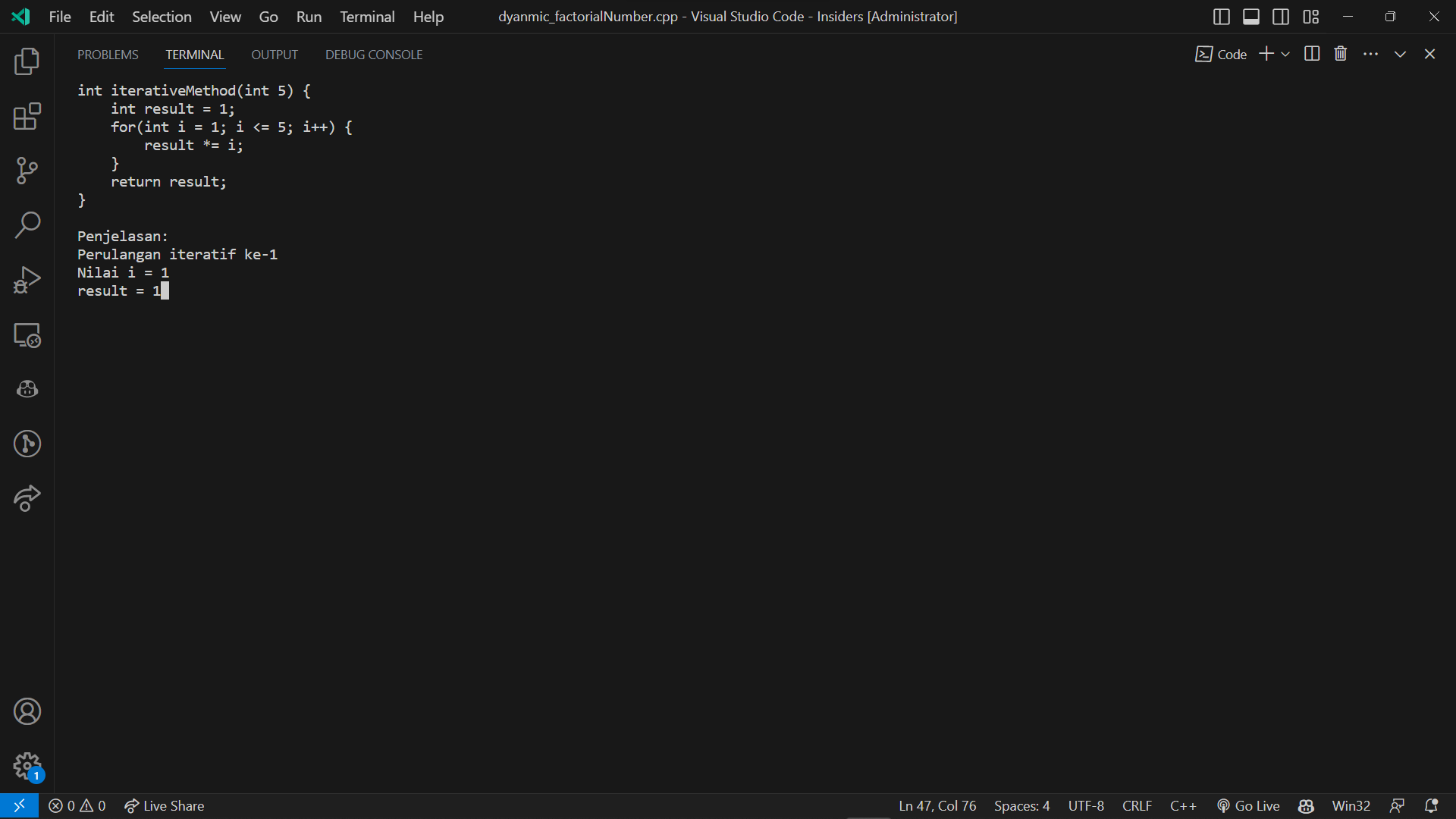
Berikut adalah penjelasan syntax pada program yang diberikan:

* #include <iostream>: merupakan preprocessor directive yang digunakan untuk mengimport library iostream yang berisi fungsi input-output standar.
* #include <unistd.h>: merupakan preprocessor directive yang digunakan untuk mengimport library unistd.h yang berisi fungsi sleep() dan system().
* using namespace std;: merupakan statement yang digunakan untuk mempermudah penggunaan fungsi-fungsi dalam namespace std, sehingga tidak perlu menuliskan std:: sebelum nama fungsi.
* class Faktorial { ... };: merupakan definisi kelas Faktorial yang berisi method-method untuk melakukan perhitungan faktorial.
* class UserInterface { ... };: merupakan definisi kelas UserInterface yang berisi method-method untuk menerima input dari user, menampilkan pilihan metode perhitungan faktorial, dan menampilkan hasil perhitungan.
* Faktorial() {...}: merupakan konstruktor kelas Faktorial yang akan dipanggil ketika objek Faktorial dibuat. Dalam konstruktor ini, nilai awal dari variabel result diinisialisasi menjadi 1.
* void showStepIterativeMethod(int number) { ... }: merupakan method kelas Faktorial yang digunakan untuk menampilkan langkah-langkah dalam perhitungan faktorial secara iteratif.
* void iterativeMethod(int number) { ... }: merupakan method kelas Faktorial yang digunakan untuk melakukan perhitungan faktorial secara iteratif.
* void showStepRecursiveMethod(int number) { ... }: merupakan method kelas Faktorial yang digunakan untuk menampilkan langkah-langkah dalam perhitungan faktorial secara rekursif.
* int recursiveMethod(int number) { ... }: merupakan method kelas Faktorial yang digunakan untuk melakukan perhitungan faktorial secara rekursif dan mengembalikan nilai hasil faktorial.
* int getResult() {...}: merupakan method kelas Faktorial yang digunakan untuk mengambil hasil faktorial dari variabel result.
* UserInterface() {...}: merupakan konstruktor kelas UserInterface yang akan dipanggil ketika objek UserInterface dibuat. Dalam konstruktor ini, input dari user diterima, pilihan metode perhitungan ditampilkan, dan hasil perhitungan faktorial beserta penjelasannya ditampilkan.
* bool menuMethod() {...}: merupakan method kelas UserInterface yang digunakan untuk menampilkan pilihan metode perhitungan faktorial kepada user dan mengembalikan nilai true jika user memilih metode iteratif atau false jika user memilih metode rekursif.
* void printResult() {...}: merupakan method kelas UserInterface yang digunakan untuk menampilkan hasil perhitungan faktorial dan penjelasannya ke layar.
* void restartProgram() {...}: merupakan method kelas UserInterface yang digunakan untuk meminta input user apakah ingin mengulang program atau tidak.
* int main() { ... }: merupakan fungsi utama pada program yang akan dijalankan pertama kali ketika program di-run. Di dalamnya, objek dari kelas UserInterface dibuat untuk memulai program.

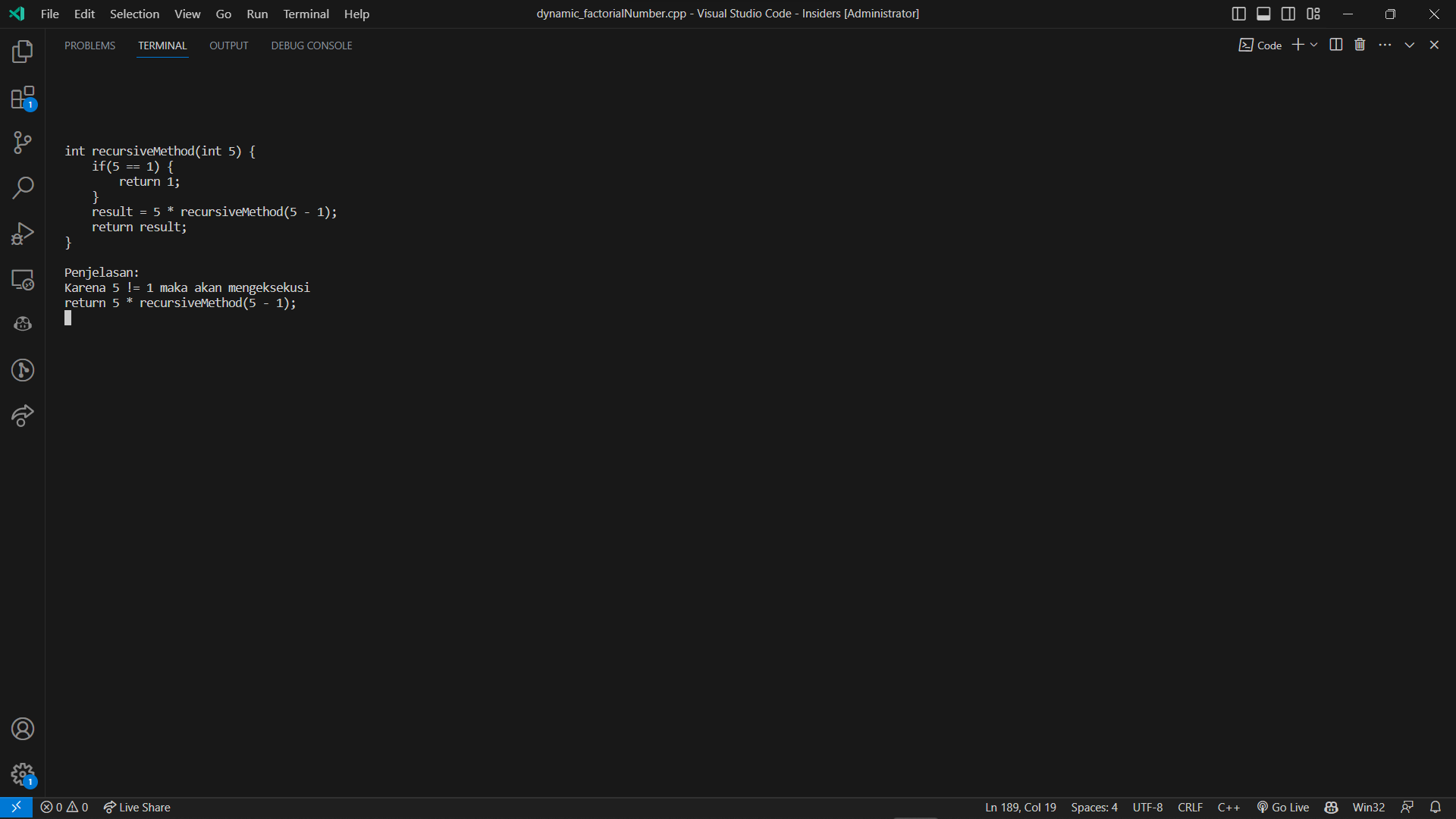
Berikut adalah contoh outputnya:



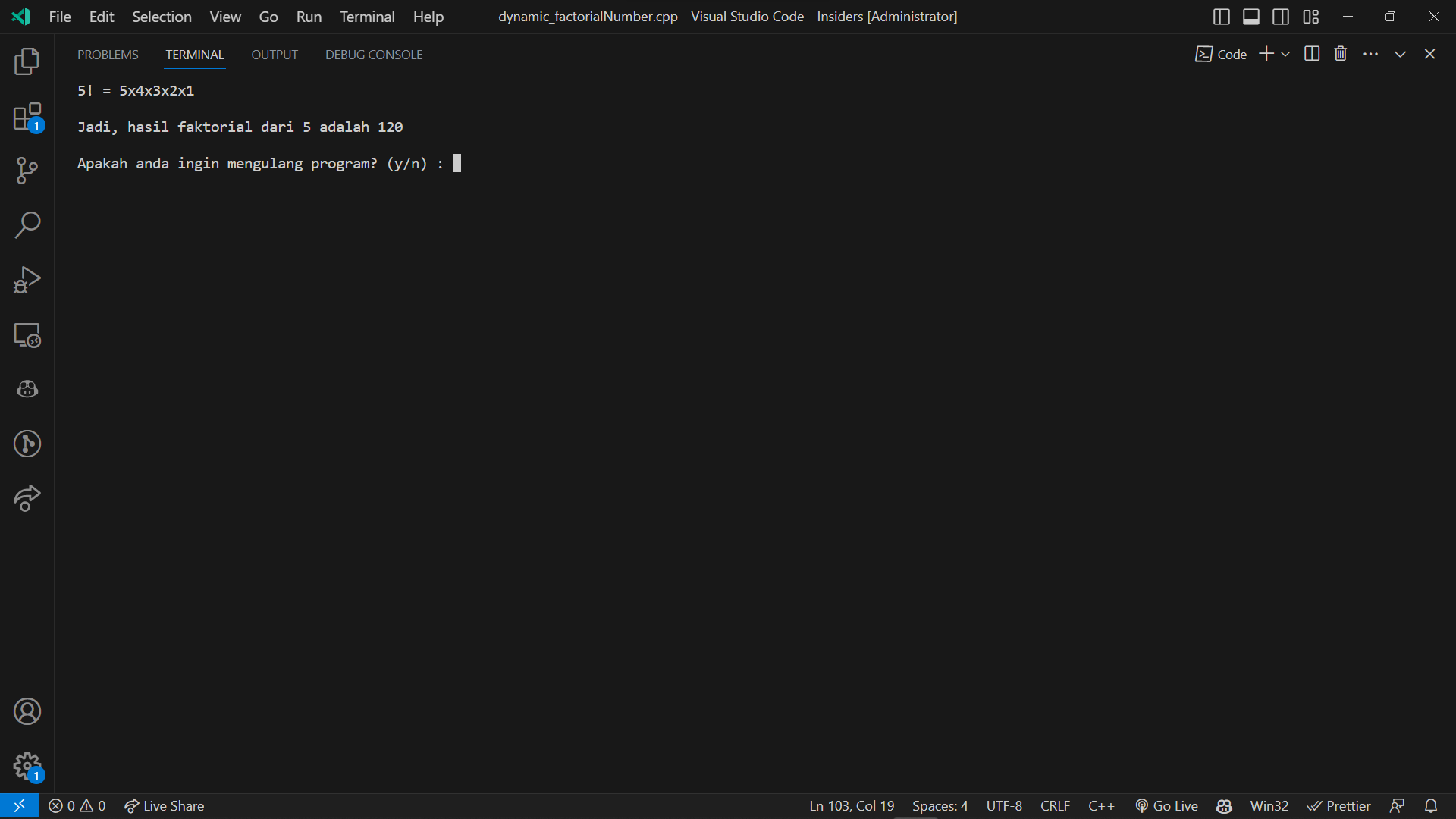
Gambar 4 Pertama, user menginputkan n!, kemudian user memilih metode apa yang digunakan untuk mencari bilangan faktorial. (Sumber :Penulis)



Gambar 5 Jika memilih iteratif, akan ditampilkan menghitung faktorial dengan iteratif. (Sumber: Penulis)



Gambar 6 Jika memilih rekursif, akan ditampilkan menghitung faktorial dengan rekursif. (Sumber: Penulis)



Gambar 7 Output hasil, kemudian diminta apakah program ingin dijalankan ulang (Sumber:Penulis)