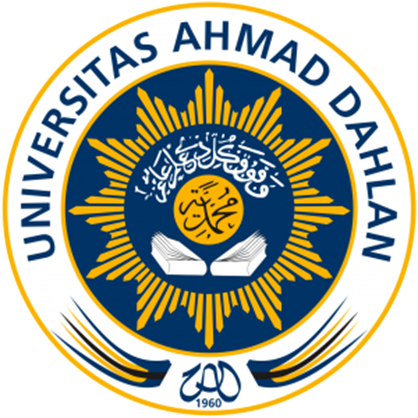
**LAPORAN PRAKTIKUM**

**“LANGKAH PRAKTIKUM 7: Aplikasi Bilangan Bulat”**

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**

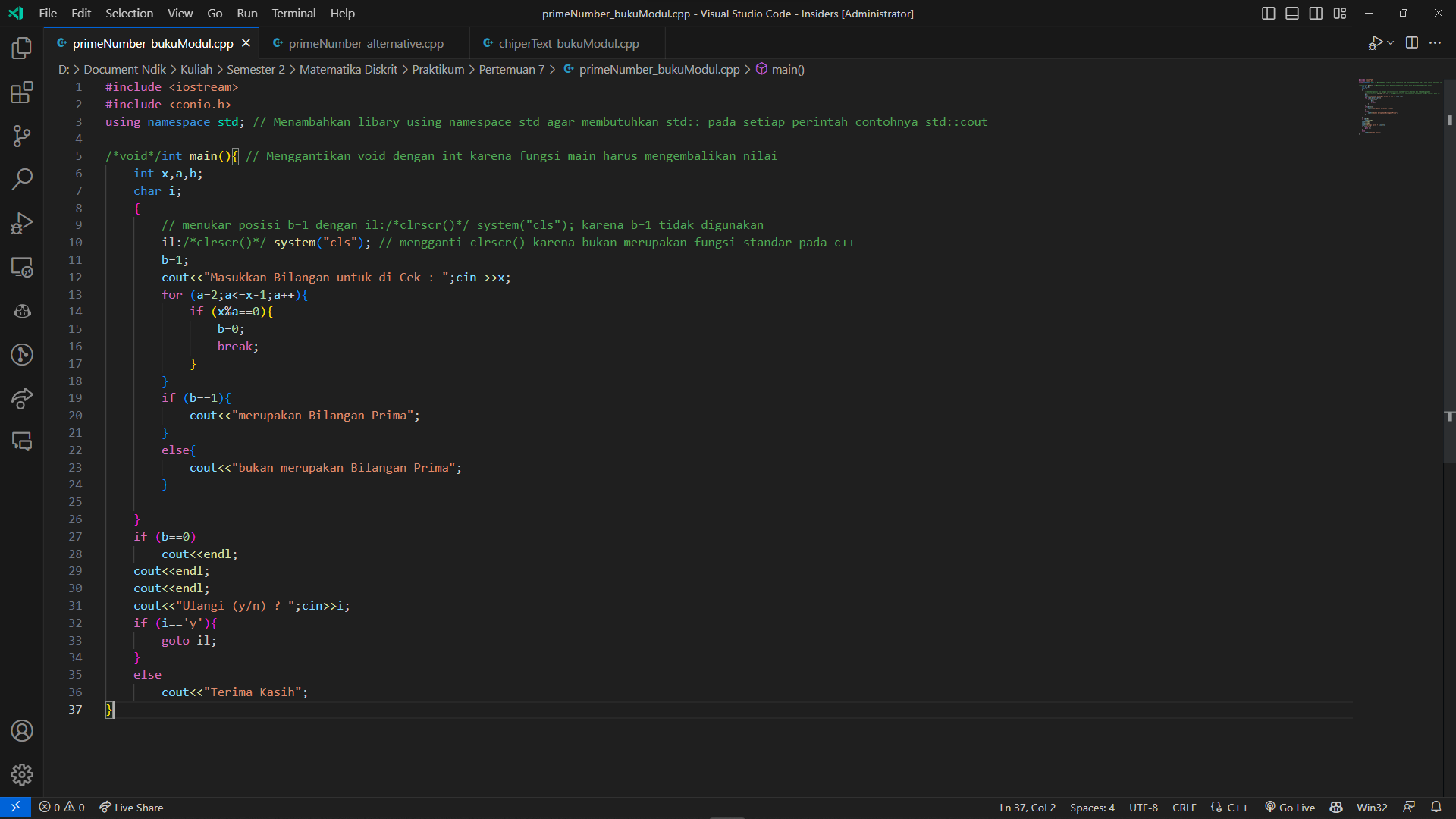
**TAHUN 2023**

DAFTAR ISI

[**Program Bilangan Prima** 3](#_Toc136340145)

[**Program ChiperText** 19](#_Toc136340146)

# **Program Bilangan Prima**



Gambar 1 Tampilan source code bilangan prima berdasarkan modul dan beberapa sudah dibenarkan. (Sumber: Penulis)

Bilangan prima adalah bilangan yang hanya bisa dibagi oleh 1 dan bilangan itu sendiri. Bilangan prima dimulai dari angka 2, yang merupakan bilangan prima terkecil dan satu-satunya bilangan prima genap.

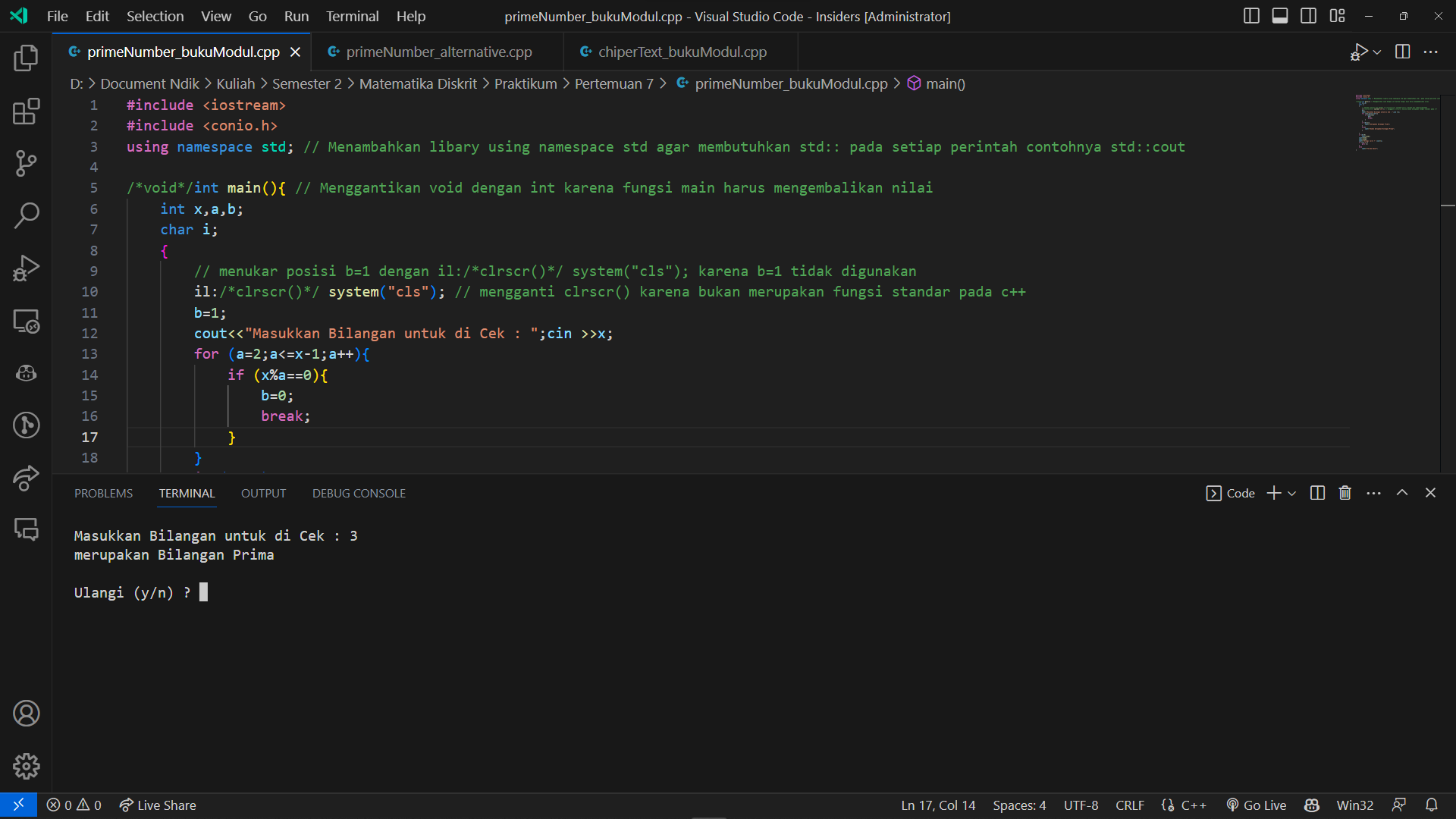
Algoritma untuk menentukan bilangan prima melibatkan pembagian bilangan yang akan dicek dengan semua bilangan di antara 2 dan bilangan tersebut dikurangi 1. Jika tidak ada pembagian yang menghasilkan sisa 0, maka bilangan tersebut adalah bilangan prima.

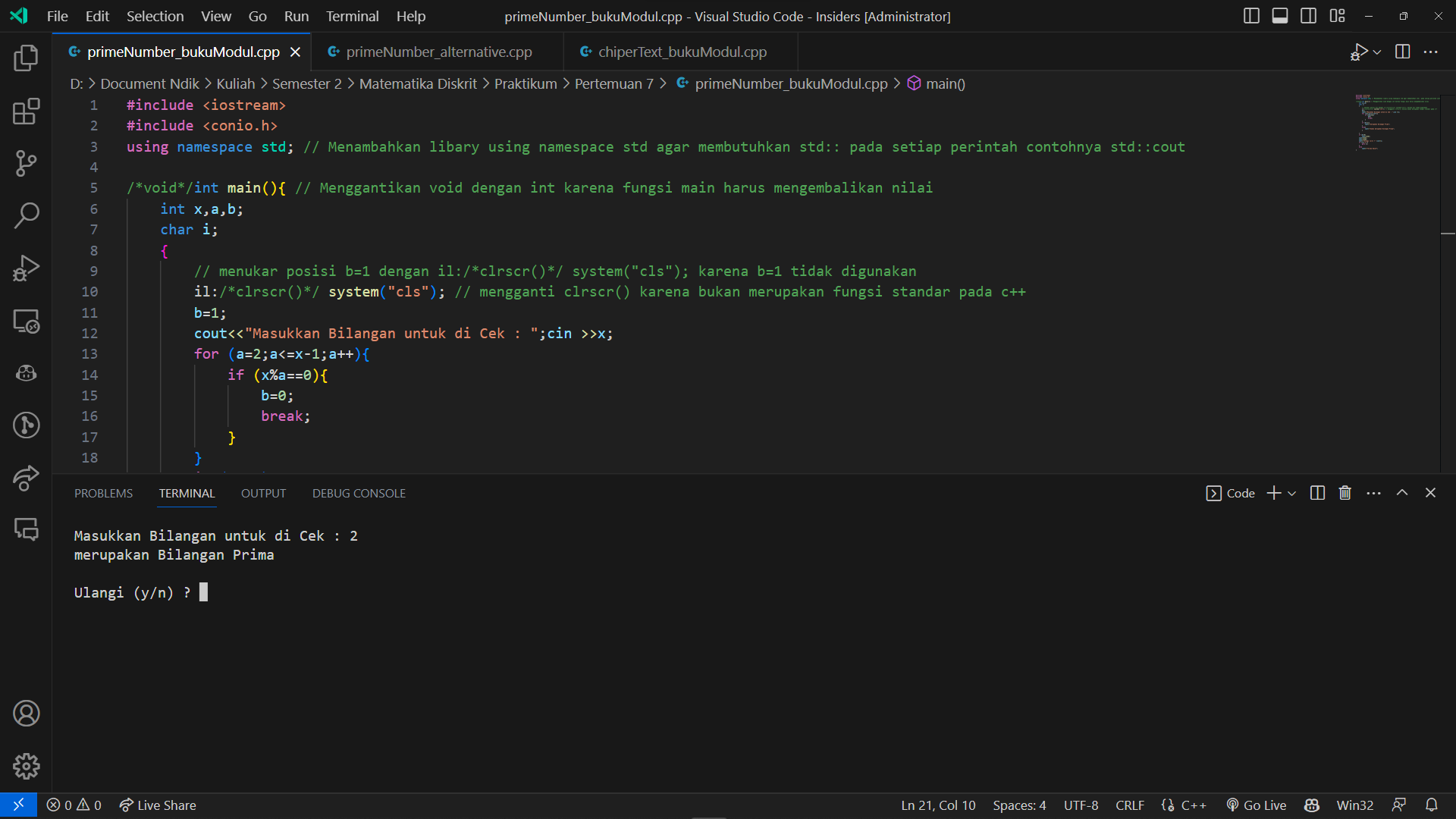
Contoh bilangan prima adalah 2,3,5,7,11,13,17 dst…

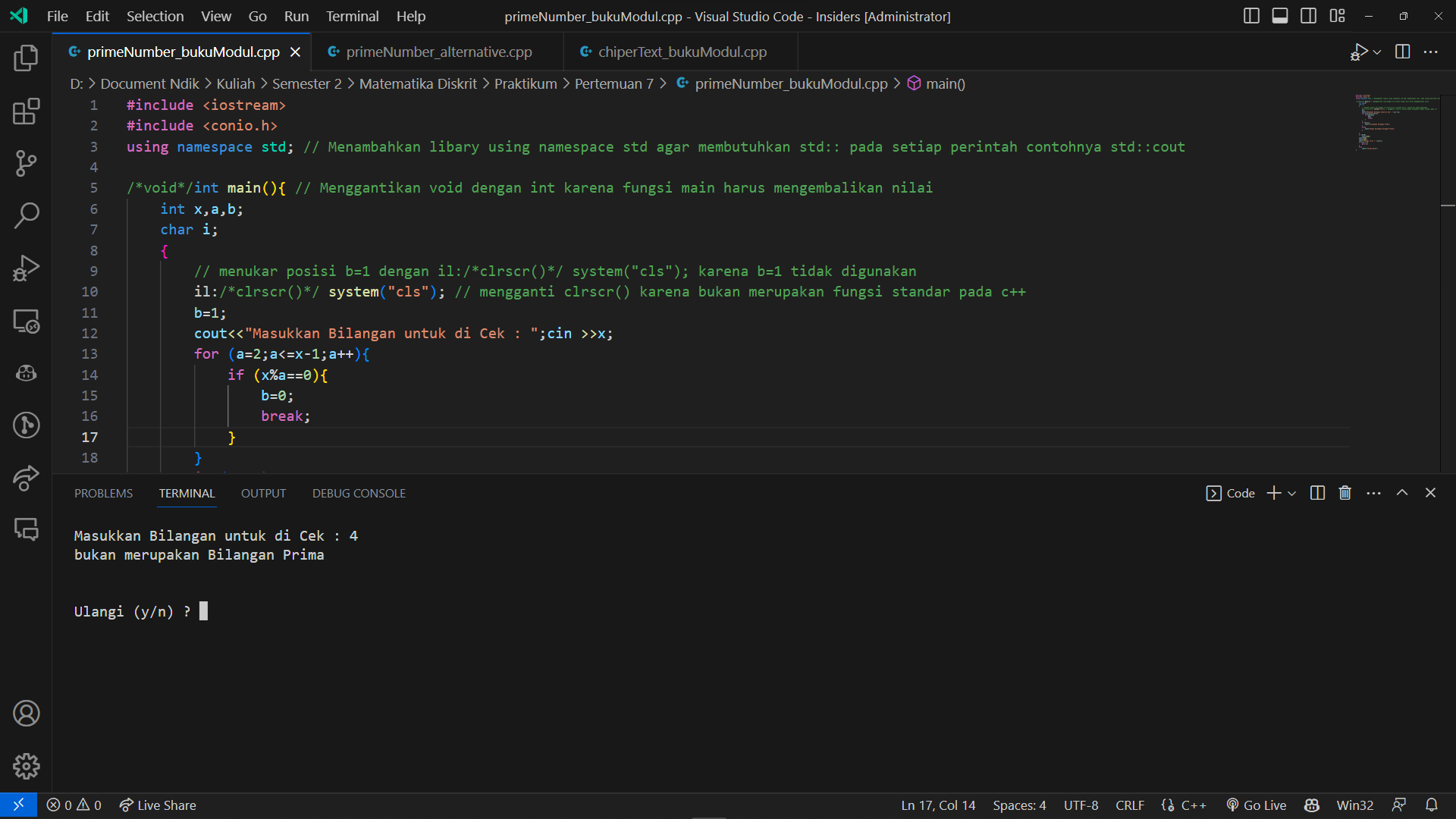
Berikut penjelasan alur program di modul:

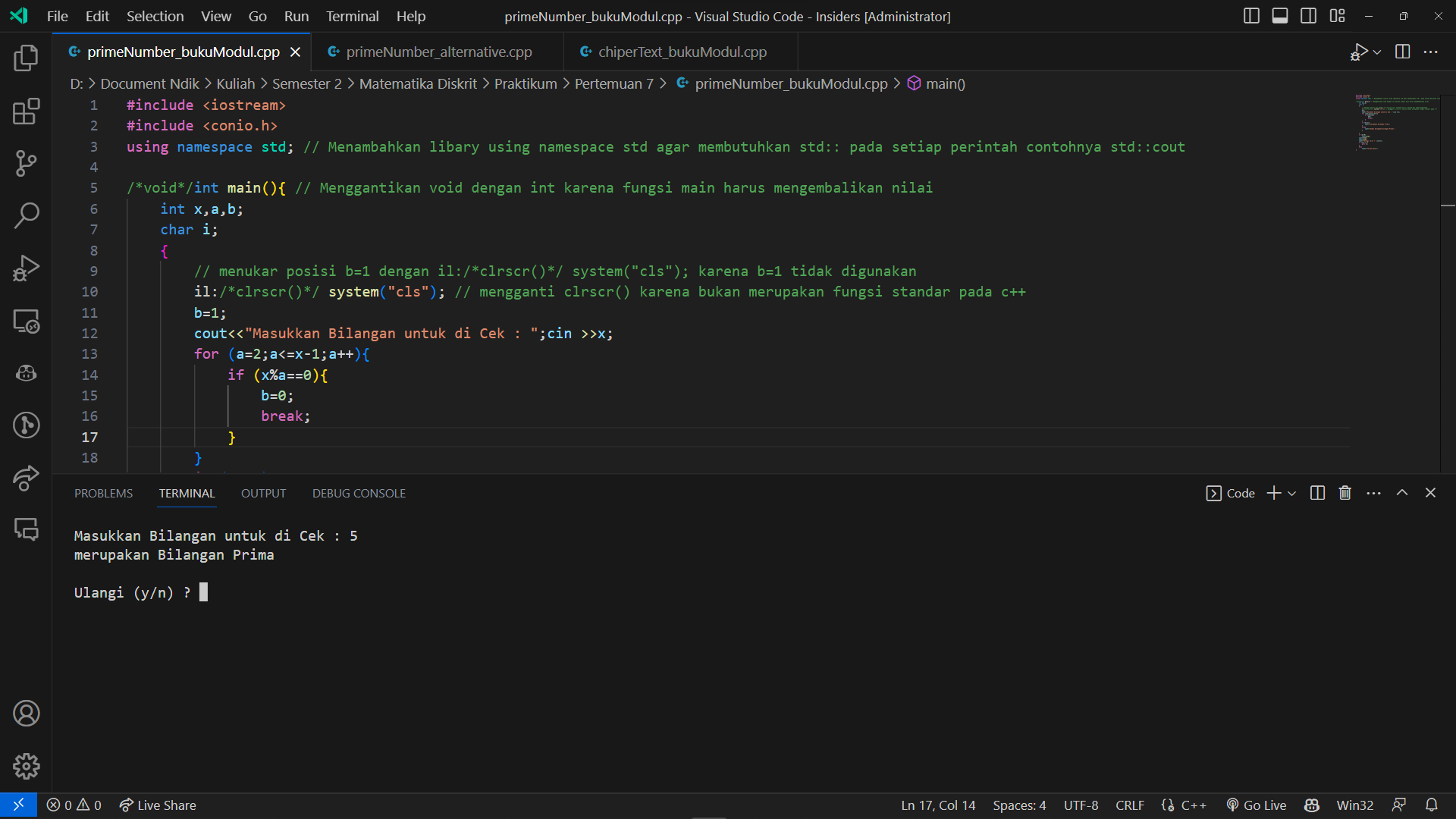
1. Program ini menggunakan library iostream dan conio.h.
2. Menggunakan namespace std untuk menghindari penulisan std:: pada setiap perintah.
3. Fungsi main() mengembalikan nilai integer.
4. Mendeklarasikan variabel x, a, b, dan i.
5. Menggunakan label 'il' untuk mengatur ulang program jika pengguna ingin mengulangi proses.
6. Membersihkan layar dengan system("cls").
7. Mengatur nilai b=1.
8. Meminta pengguna untuk memasukkan bilangan yang akan dicek (x).
9. Melakukan iterasi dari a=2 hingga x-1. a. Jika x dapat dibagi habis oleh a (x % a == 0), maka mengatur b=0 dan keluar dari loop.
10. Jika b==1, maka bilangan tersebut merupakan bilangan prima.
11. Jika b==0, maka bilangan tersebut bukan merupakan bilangan prima.
12. Menampilkan hasil dan menanyakan apakah pengguna ingin mengulangi proses.
13. Jika pengguna memasukkan 'y', program akan kembali ke label 'il'.
14. Jika pengguna memasukkan selain 'y', program akan menampilkan "Terima Kasih" dan berakhir.

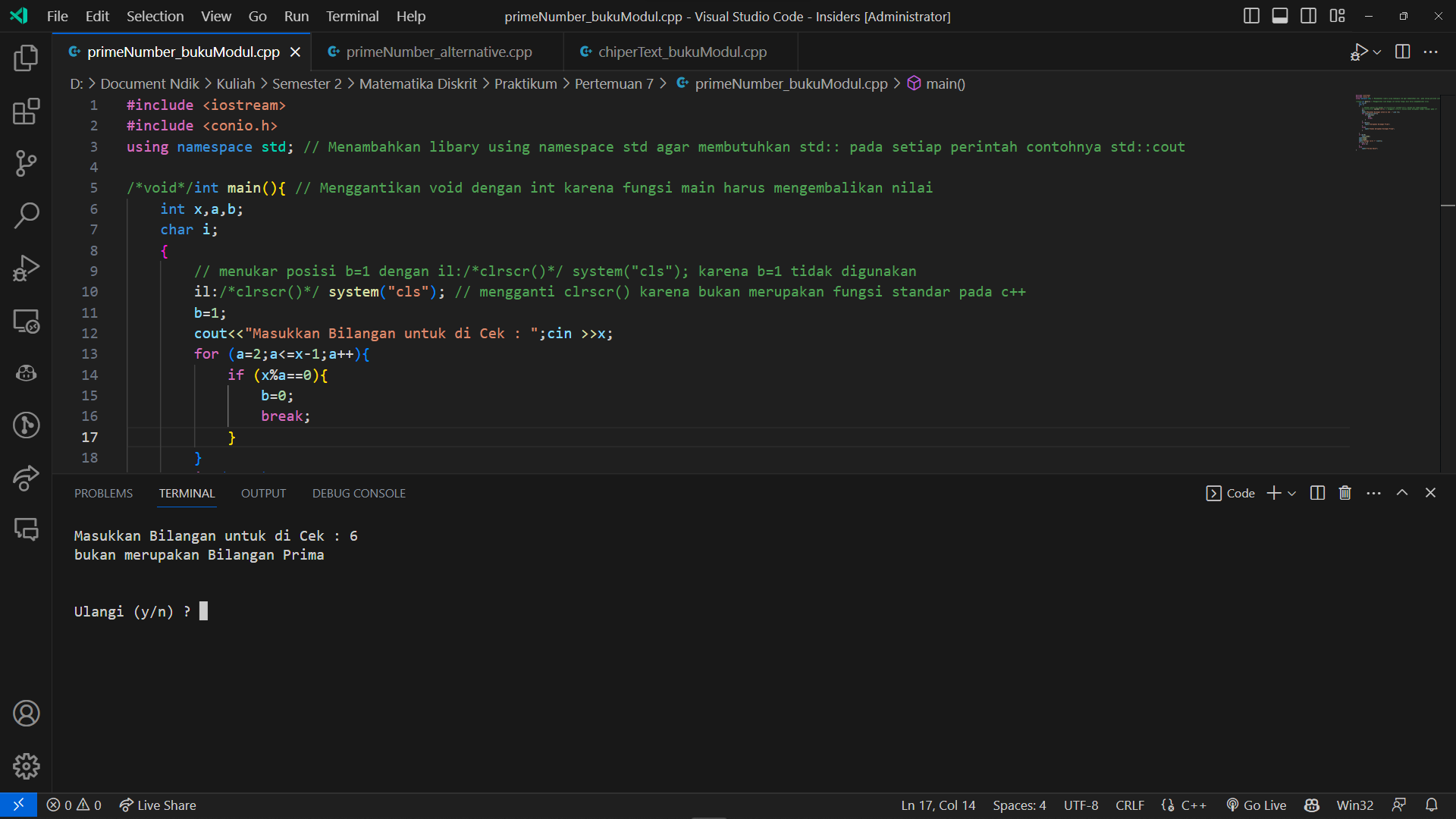
Berikut adalah hasil outputnya:

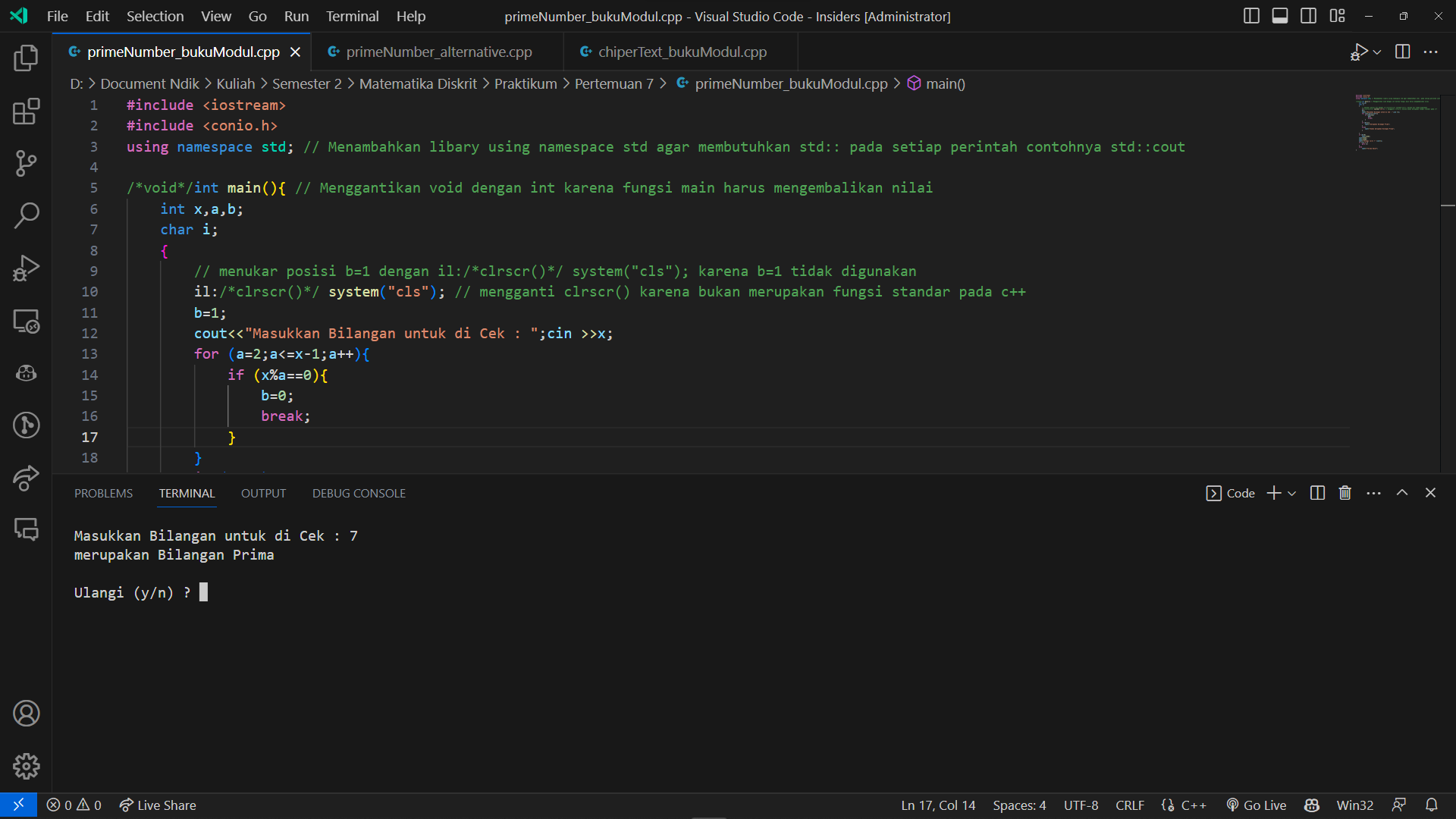


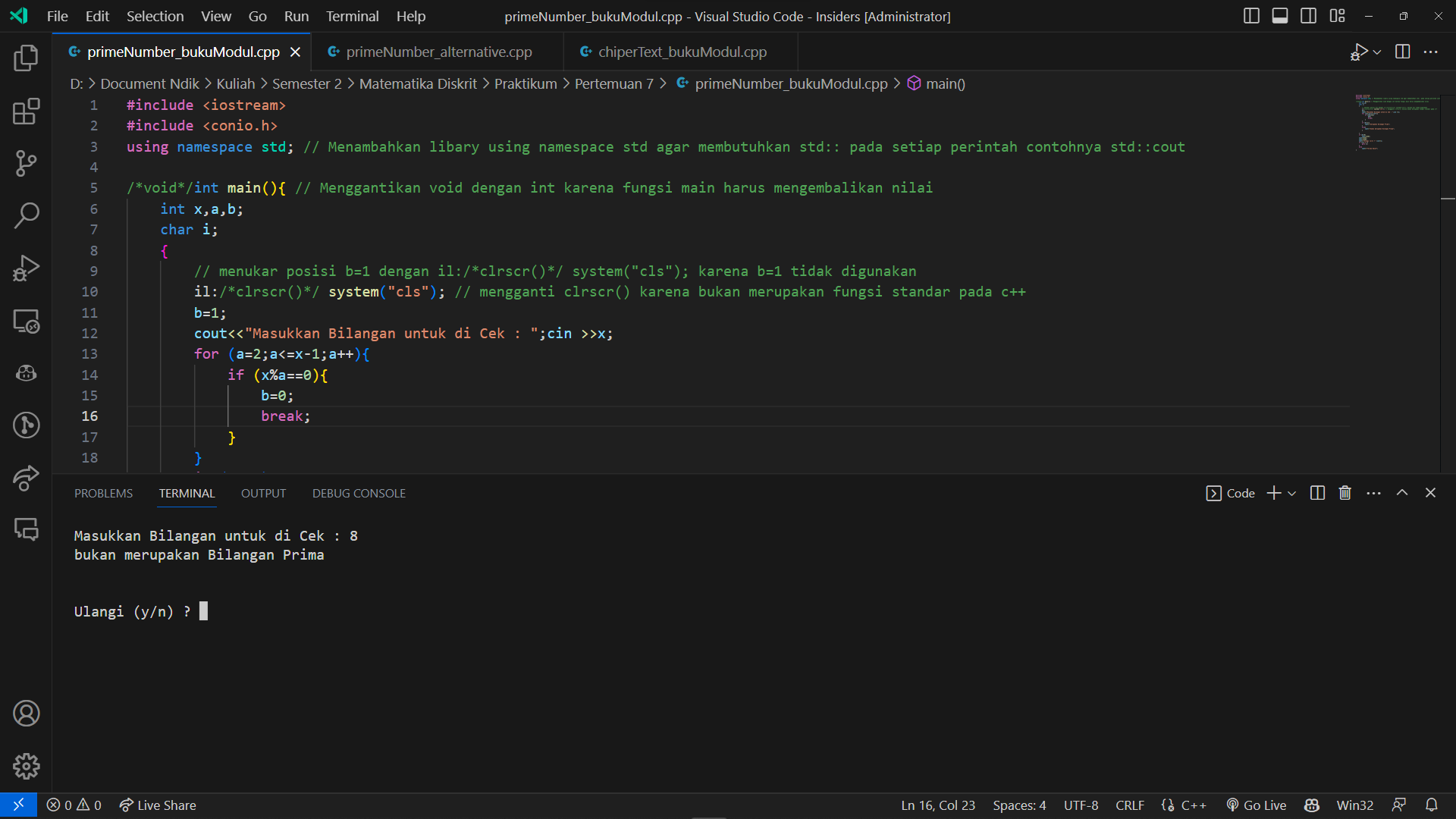


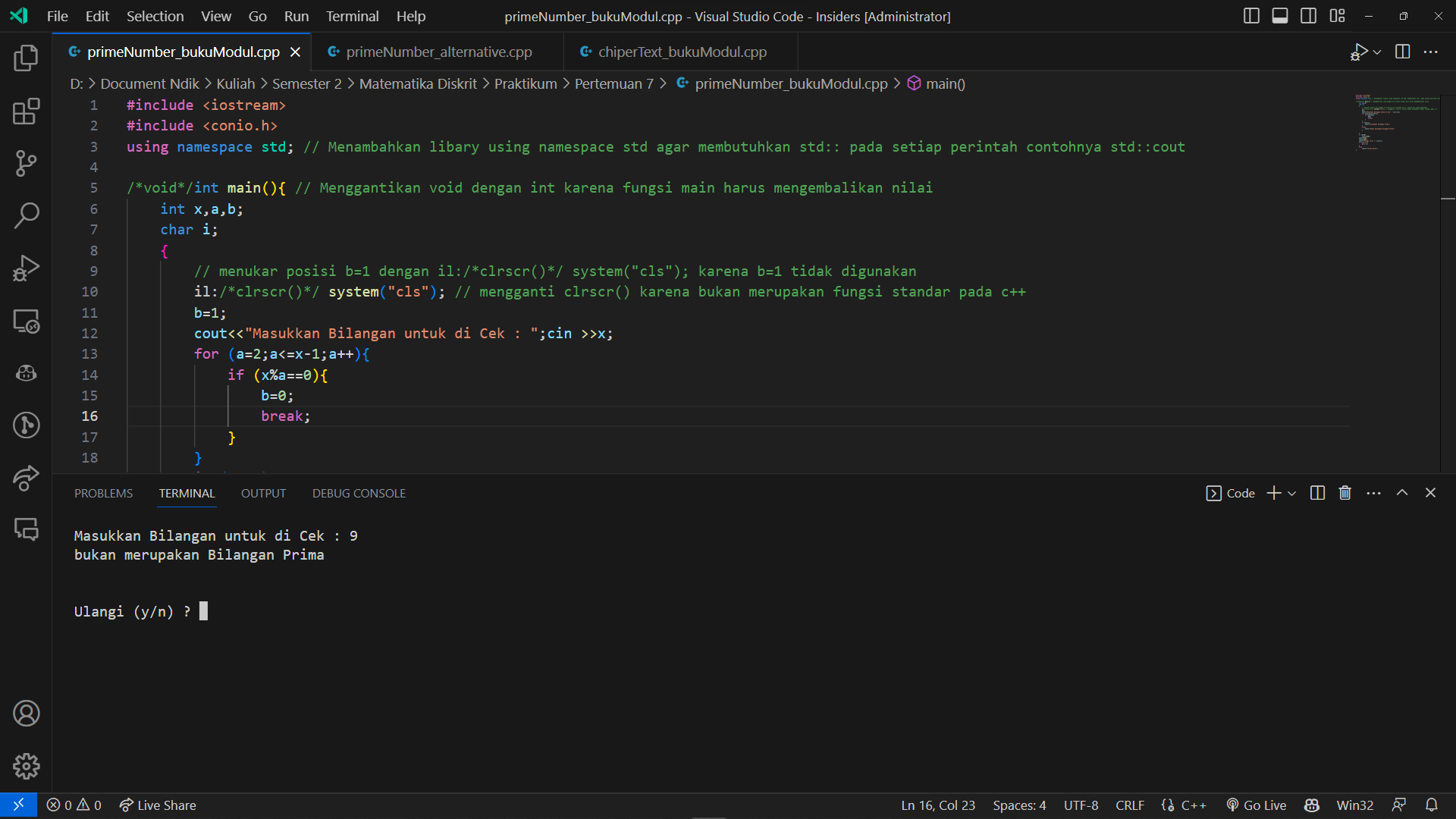


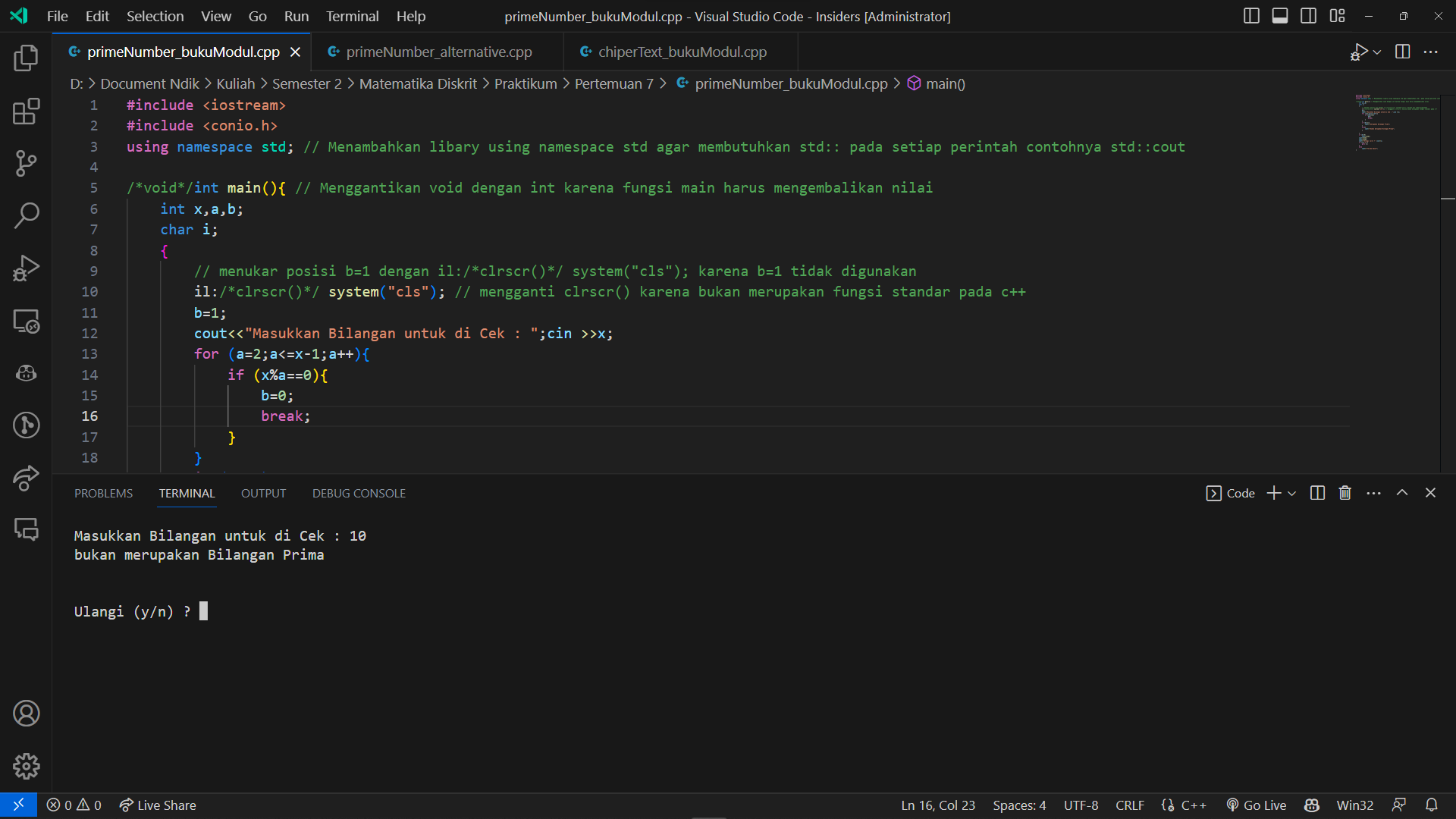








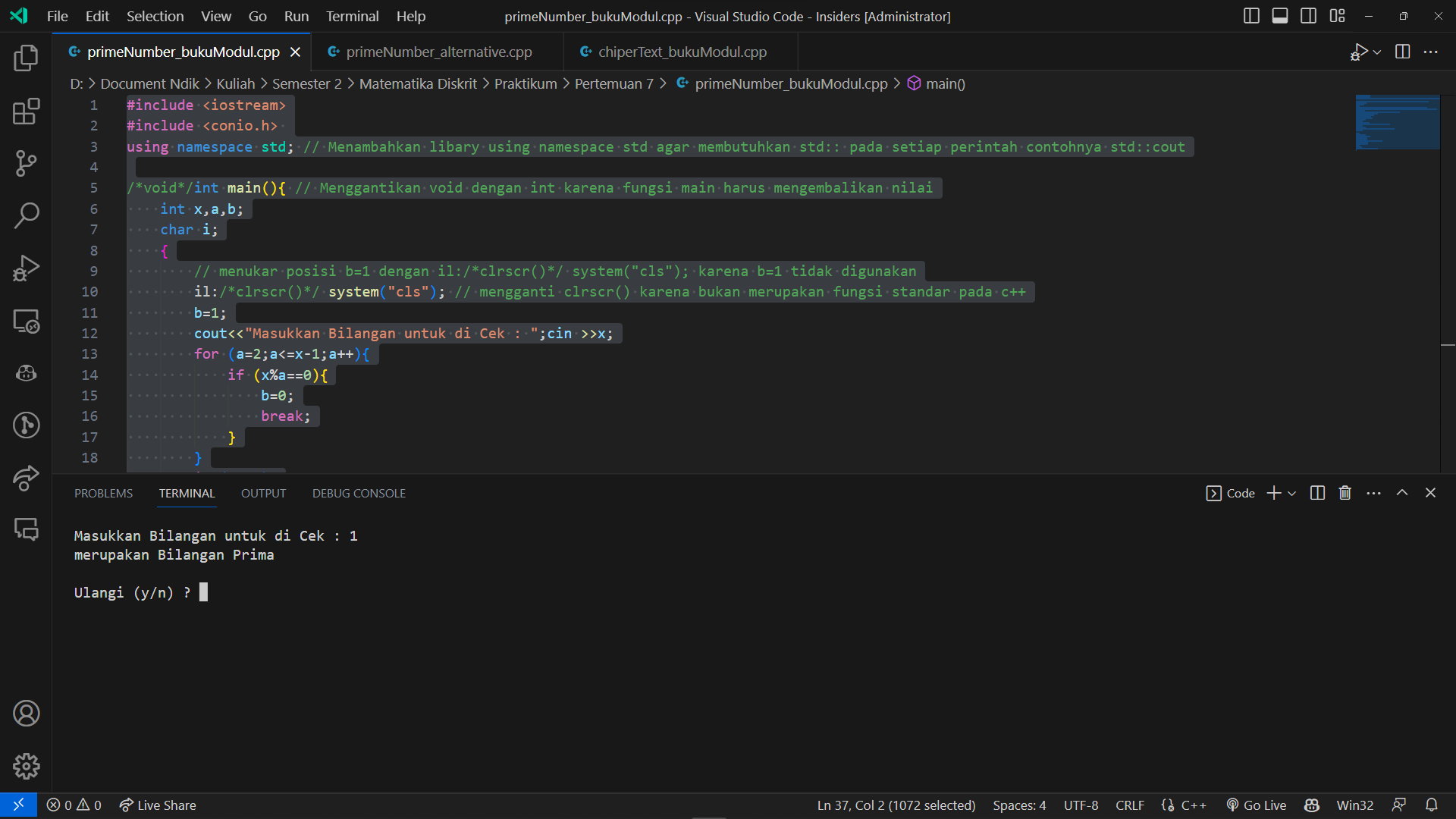




Gambar 2 Mengecek bilangan prima.(Sumber: Penulis)

Berikut adalah analisis kesalahan kodingna pada modul sebelum diperbaiki.

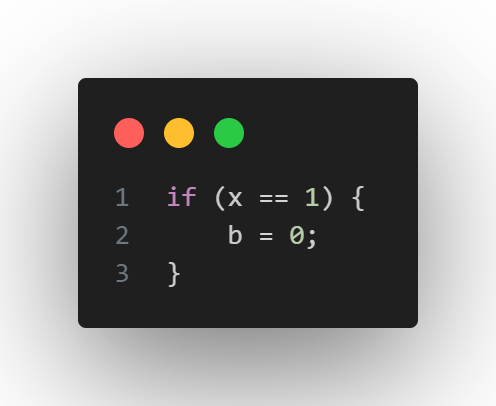
1. Kesalahan dalam menangani bilangan 1 sebagai bilangan prima:



Gambar 3 Kesalahan kodingan pada modul, angka 1 seharusnya bukan merupakan Bilangan Prima. (Sumber : Penulis)

Program ini menganggap bilangan 1 sebagai bilangan prima karena tidak ada pengecekan khusus untuk bilangan 1. Untuk memperbaiki kesalahan ini, Anda perlu menambahkan pengecekan khusus sebelum memulai iterasi pembagian. Tambahkan kode berikut sebelum for loop:

Tambahkan kode berikut sebelum for loop:

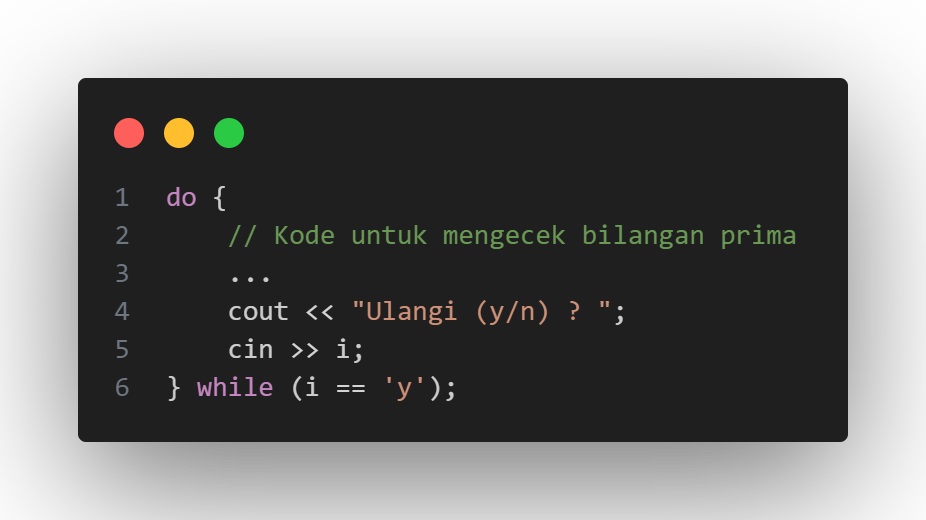


Gambar 4 menabahkan kondisional untuk mereset isi variabel b kembali menjadi 0. (Sunber: Penulis)

1. Penggunaan goto:

Penggunaan goto tidak disarankan dalam pemrograman modern karena dapat menyebabkan struktur program yang sulit dipahami dan dikelola. Sebagai gantinya, bisa menggunakan loop, seperti do-while, untuk mengulangi proses jika pengguna ingin melakukannya.

Gantilah bagian kode yang menggunakan goto dengan do-while loop:



Gambar 5 mengganti goto menjadi do while. (Sumber: Penulis)

Dengan perbaikan-perbaikan ini, program Anda akan lebih akurat dalam menentukan bilangan prima dan lebih mudah untuk dipahami dan dikelola.

1. Menambahkan using namespace std;

Dengan menambahkan using namespace std, dapat memudahkan programmer karena ada beberapa syntax yang memerlukan seperti std::cout dan std::cin. Apabila tidak ingin menggunakan using namespace std; dapat menggantikan dengan menambahkan std:: sebelum cin dan cout.

1. Mengganti void main() dengan int main()

void main() dan int main() adalah dua cara yang berbeda untuk menulis fungsi utama dalam bahasa C++. Namun, int main() adalah cara yang disarankan untuk menulis fungsi utama dalam bahasa C++.

int main() mengembalikan nilai integer sebagai status keluaran dari program. Status keluaran ini dapat digunakan oleh program yang memanggil program C++ sebagai indikator apakah program C++ berhasil atau gagal. Nilai kembali 0 menunjukkan bahwa program berjalan dengan sukses, sedangkan nilai kembali selain 0 menunjukkan bahwa program gagal.

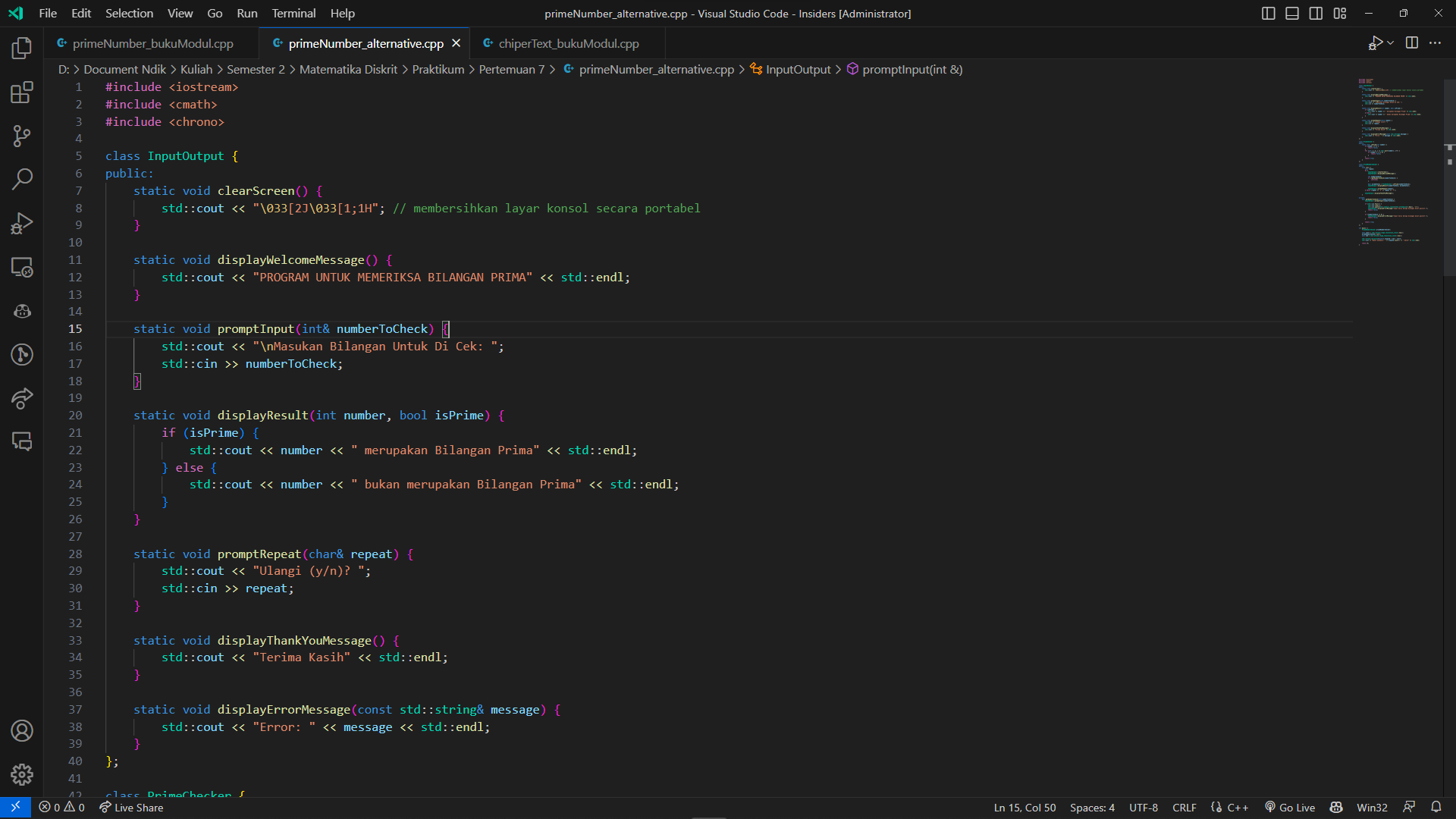
Di sisi lain, void main() tidak mengembalikan nilai apa pun. Oleh karena itu, tidak ada cara untuk mengetahui apakah program berhasil atau gagal ketika menggunakan void main(). Selain itu, void main() tidak didefinisikan dalam standar bahasa C++, sehingga penggunaannya tidak disarankan.

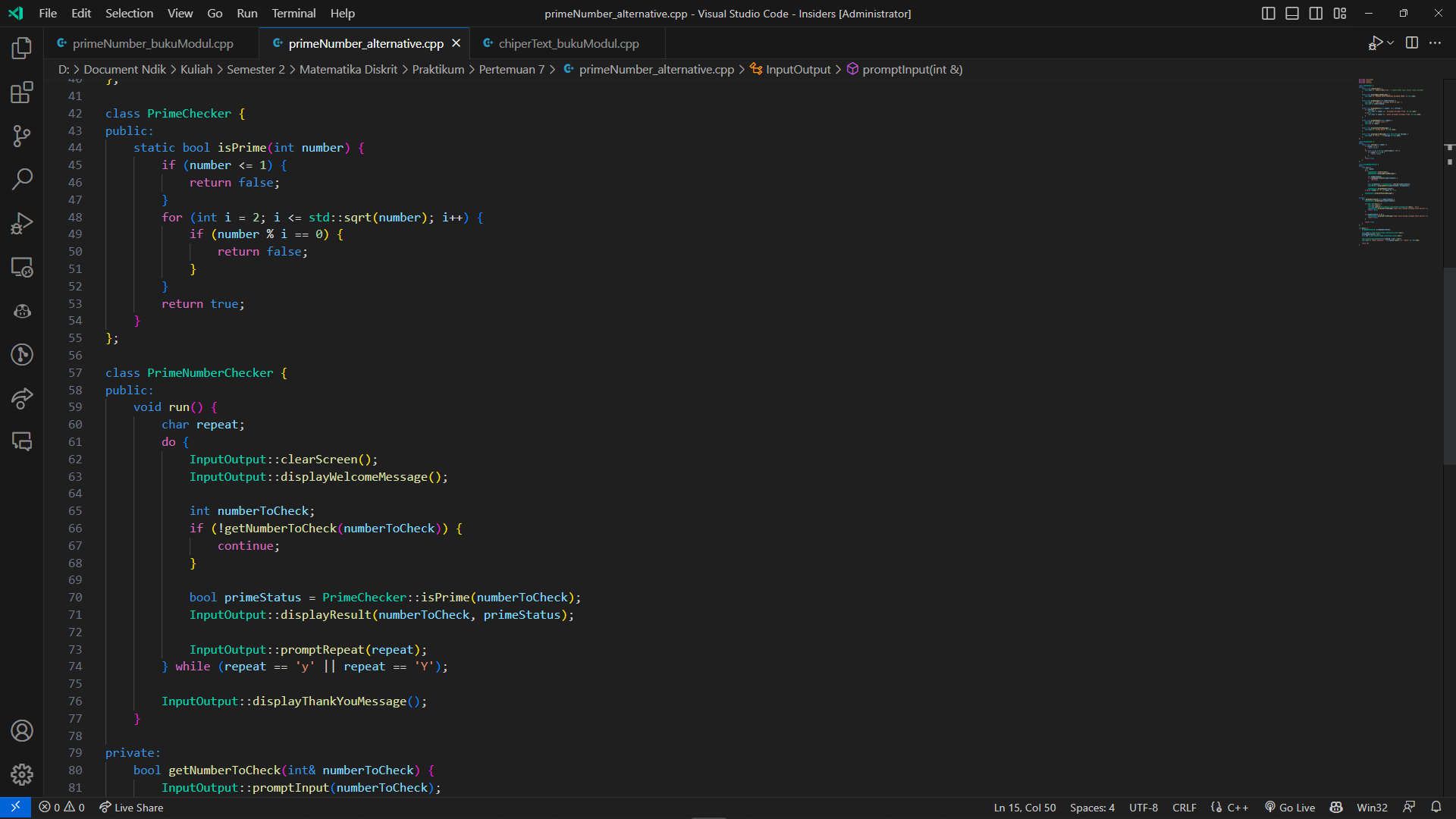
1. Mengganti clrscr() dengan system(“cls”)

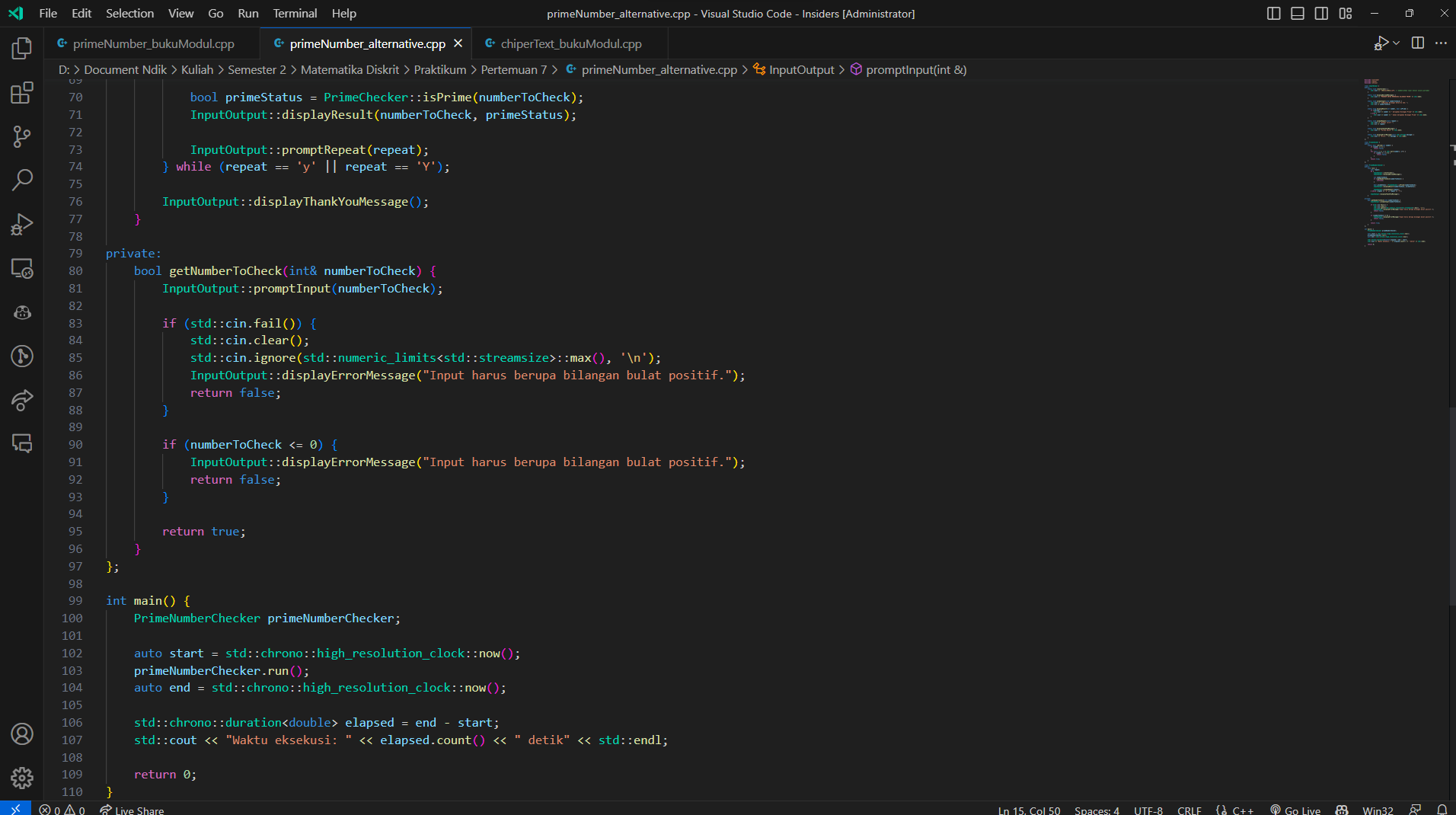
Kode il:/\*clrscr()\*/ system("cls"); digunakan untuk membersihkan layar konsol pada program C++. Namun, clrscr() bukanlah fungsi standar dalam bahasa C++, sehingga kode tersebut diganti dengan system("cls").

Fungsi system("cls") memanggil perintah sistem untuk membersihkan layar konsol. Perintah ini bekerja pada sistem operasi Windows dan akan menghapus semua teks yang ada di layar konsol.

**Berikut adalah kodingan alternatifnya**

****





Gambar 6 Kodingan alternatif untuk mengecek apakah bilangan prima atau bukan dengan. (Sumber: Penulis)

Program ini bertujuan untuk memeriksa apakah sebuah bilangan merupakan bilangan prima atau bukan. Program ini akan meminta input bilangan dari pengguna, kemudian memeriksa apakah bilangan tersebut prima atau bukan. Jika bilangan tersebut prima, maka program akan menampilkan pesan "bilangan tersebut merupakan bilangan prima", sedangkan jika bukan, program akan menampilkan pesan "bilangan tersebut bukan merupakan bilangan prima". Setelah itu, program akan meminta pengguna untuk mengulang atau tidak. Program akan terus berjalan dan meminta input bilangan dari pengguna sampai pengguna memilih untuk tidak mengulang lagi.

Dalam program tersebut, konsep Pemrograman Berorientasi Objek (OOP) diimplementasikan melalui penggunaan kelas dan objek. Berikut adalah penjelasan tentang bagaimana OOP diterapkan dalam program ini:

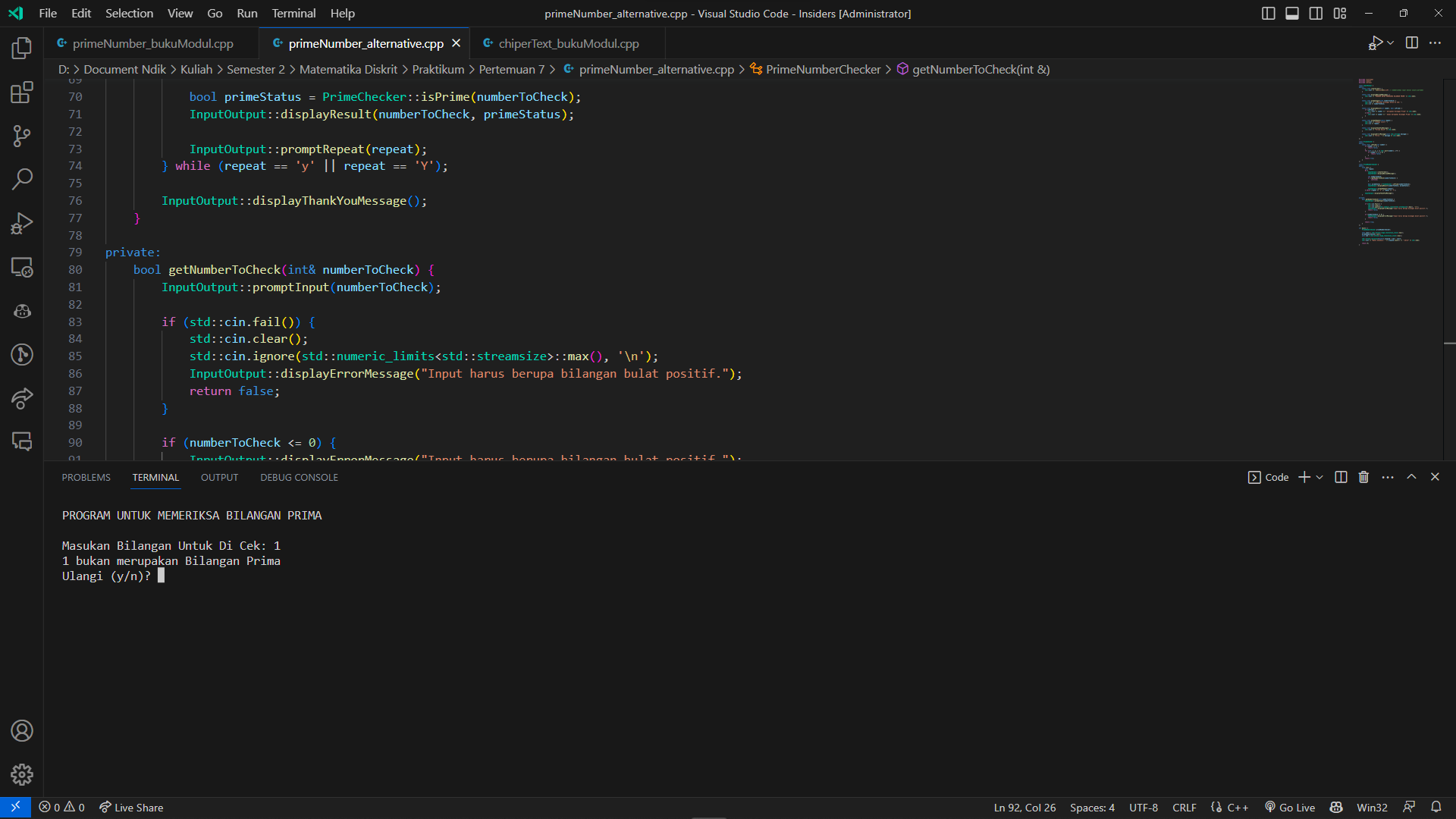
* Kelas: Program ini mendefinisikan satu kelas bernama Enkripsi. Kelas ini berfungsi sebagai cetak biru untuk membuat objek yang dapat melakukan enkripsi dan dekripsi teks menggunakan metode Caesar cipher. Kelas ini memiliki atribut dan metode yang saling terkait untuk mencapai tujuan ini.
* Atribut: Atribut dalam kelas Enkripsi adalah chiper, key, dan plain. Atribut-atribut ini digunakan untuk menyimpan informasi yang relevan dengan proses enkripsi dan dekripsi, seperti teks yang akan dienkripsi, kunci enkripsi, dan teks yang telah didekripsi.
* Metode: Metode dalam kelas Enkripsi adalah Enkripsi(), enkripsi(), deskripsi(), dan output(). Metode-metode ini mendefinisikan perilaku objek yang dibuat dari kelas Enkripsi. Metode Enkripsi() adalah konstruktor yang dipanggil saat objek dibuat, dan metode enkripsi(), deskripsi(), dan output() adalah metode yang digunakan untuk mengenkripsi teks, mendekripsi teks, dan mencetak teks yang telah dienkripsi atau didekripsi.
* Objek: Dalam fungsi main, objek Deskripsi dibuat dari kelas Enkripsi. Objek ini digunakan untuk memanggil metode yang ada dalam kelas Enkripsi untuk mengenkripsi dan mendekripsi teks yang dimasukkan oleh pengguna. Objek ini juga menyimpan atribut yang relevan dengan proses enkripsi dan dekripsi.

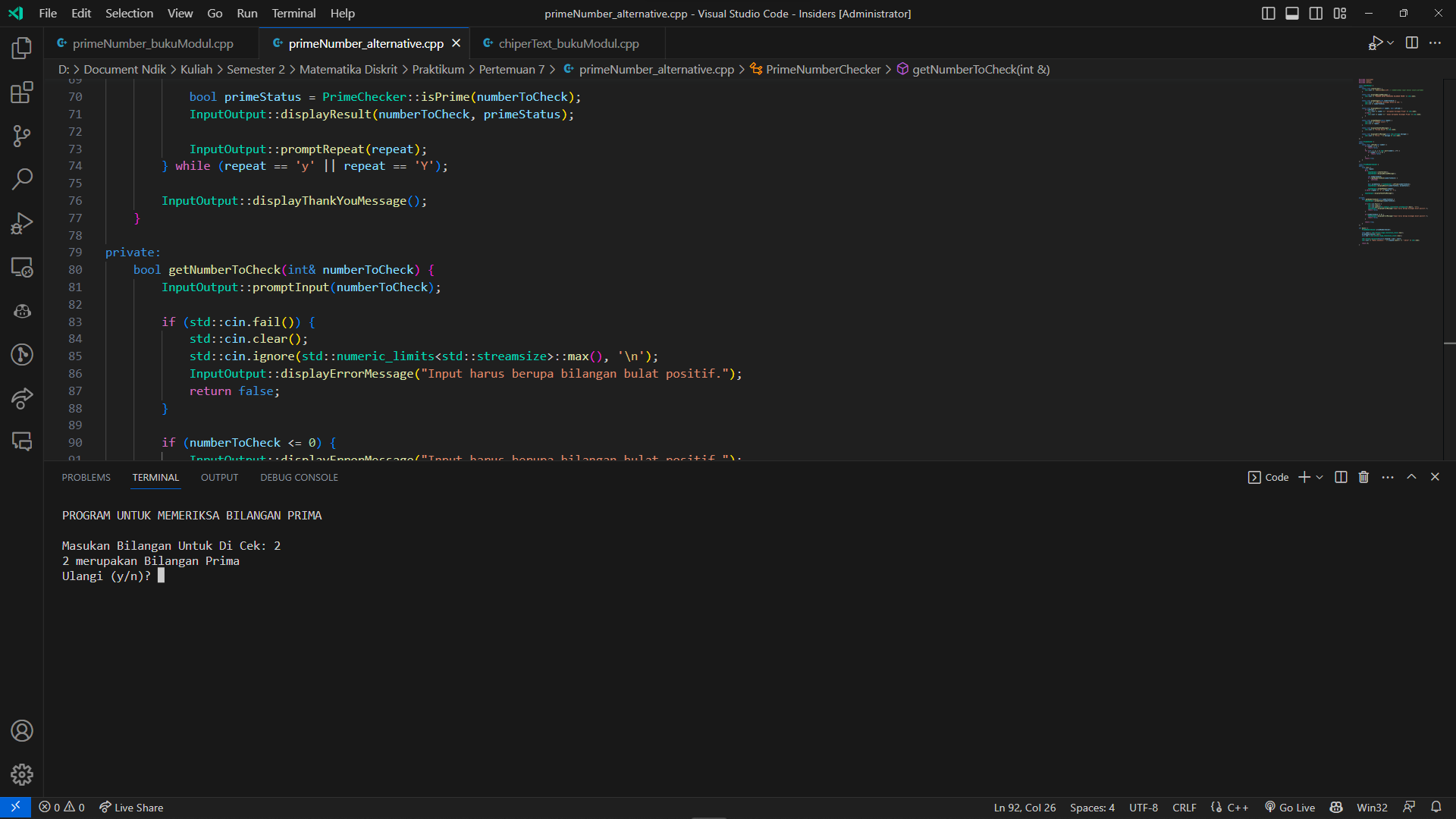
Dengan menggunakan konsep OOP, program ini dapat dengan mudah diperluas atau dimodifikasi. Misalnya, Anda dapat menambahkan metode baru atau mengubah metode yang ada dalam kelas Enkripsi untuk mengimplementasikan algoritma enkripsi yang berbeda.

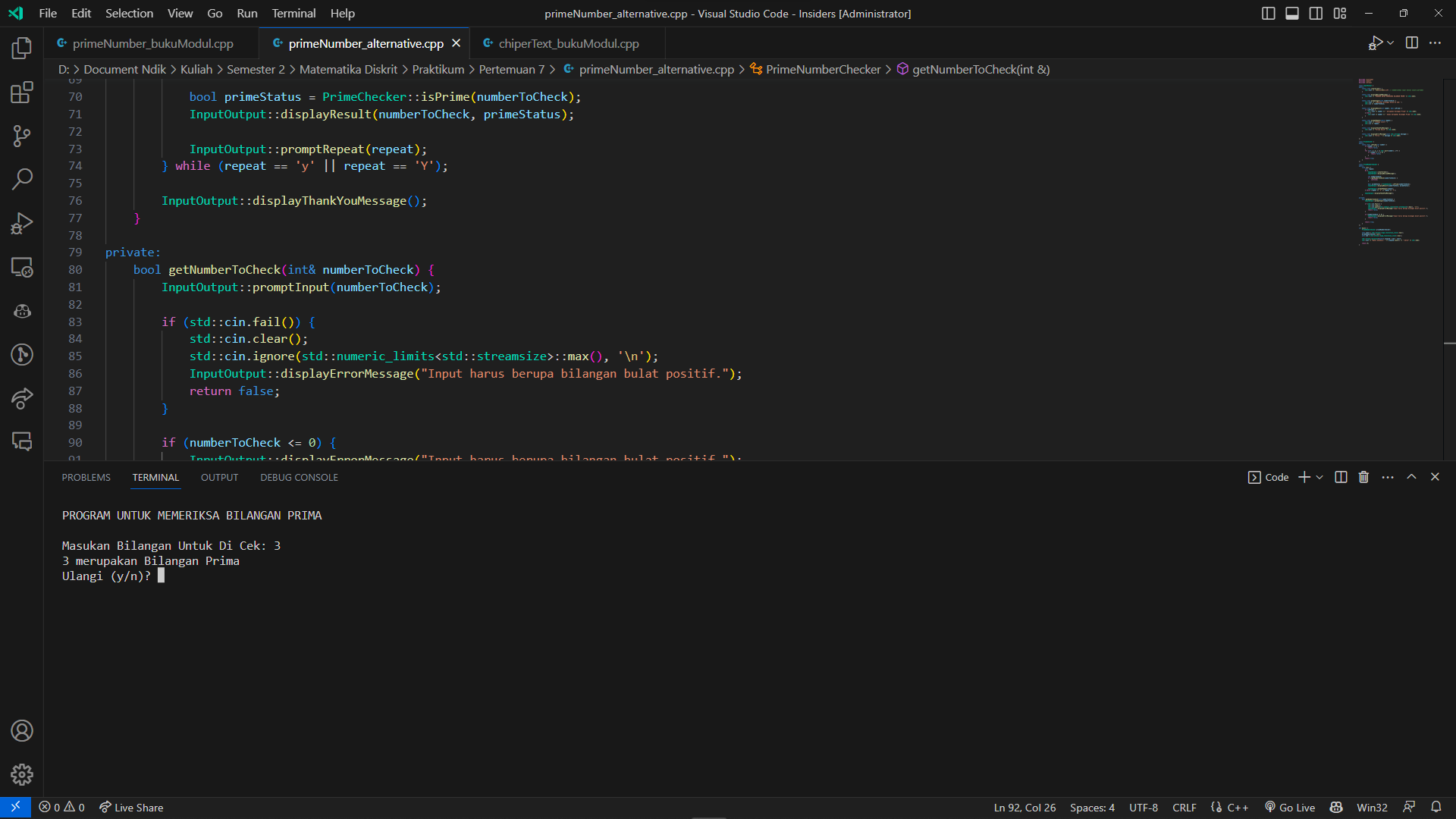
Alur programnya adalah sebagai berikut:

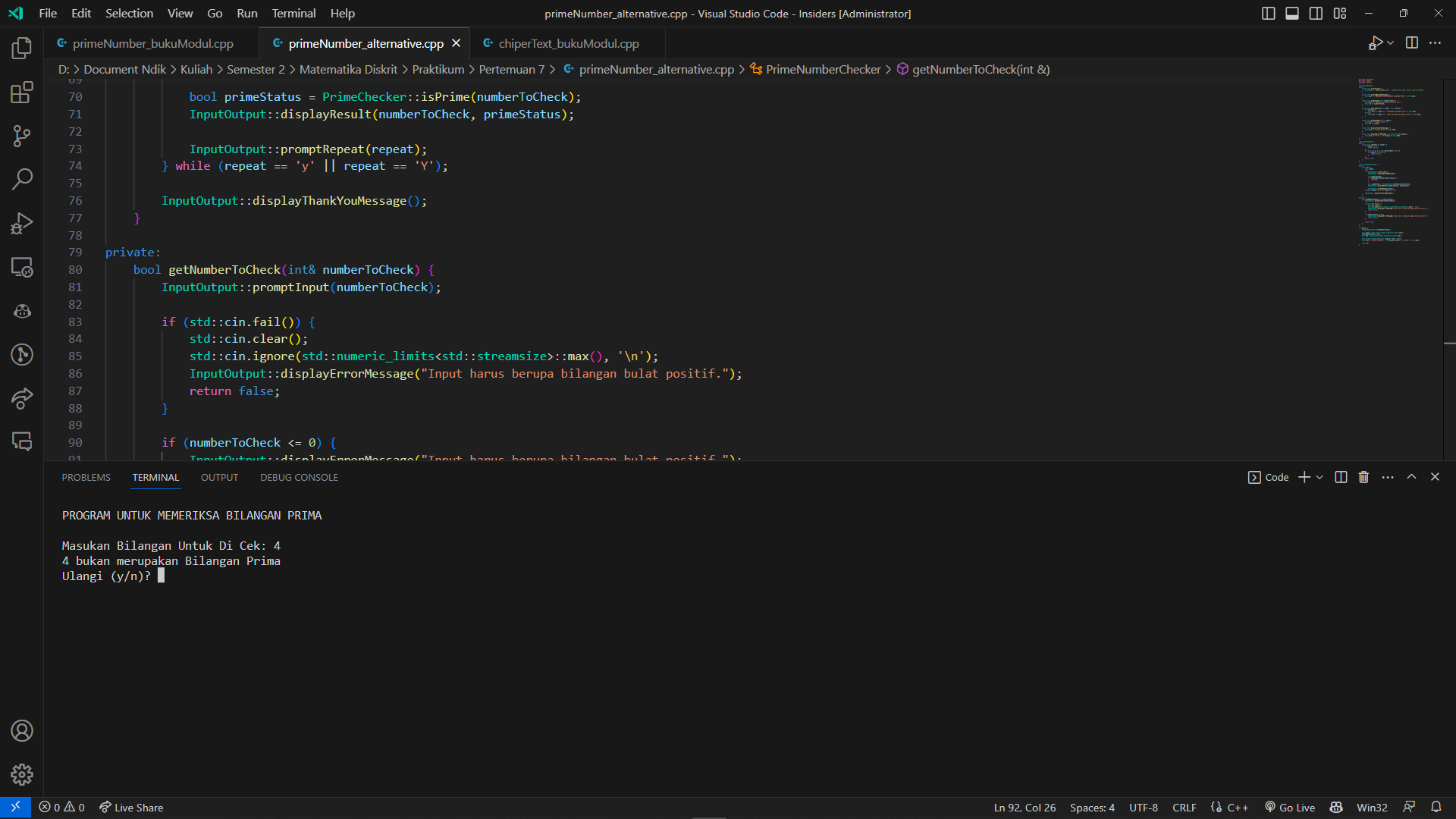
1. Program akan menampilkan pesan "PROGRAM UNTUK MEMERIKSA BILANGAN PRIMA".
2. Program akan meminta input bilangan dari pengguna.
3. Program akan memeriksa apakah input bilangan tersebut valid (bilangan bulat positif).
4. Jika input bilangan tidak valid, program akan menampilkan pesan error dan meminta input bilangan lagi.
5. Jika input bilangan valid, program akan memeriksa apakah bilangan tersebut prima atau bukan.
6. Jika bilangan tersebut prima, program akan menampilkan pesan "bilangan tersebut merupakan bilangan prima".
7. Jika bilangan tersebut bukan prima, program akan menampilkan pesan "bilangan tersebut bukan merupakan bilangan prima".
8. Program akan meminta pengguna untuk mengulang atau tidak.
9. Jika pengguna memilih untuk mengulang, program akan kembali ke langkah 2.
10. Jika pengguna memilih untuk tidak mengulang, program akan menampilkan pesan "Terima Kasih" dan berhenti.
11. Program akan menampilkan waktu eksekusi dari program.

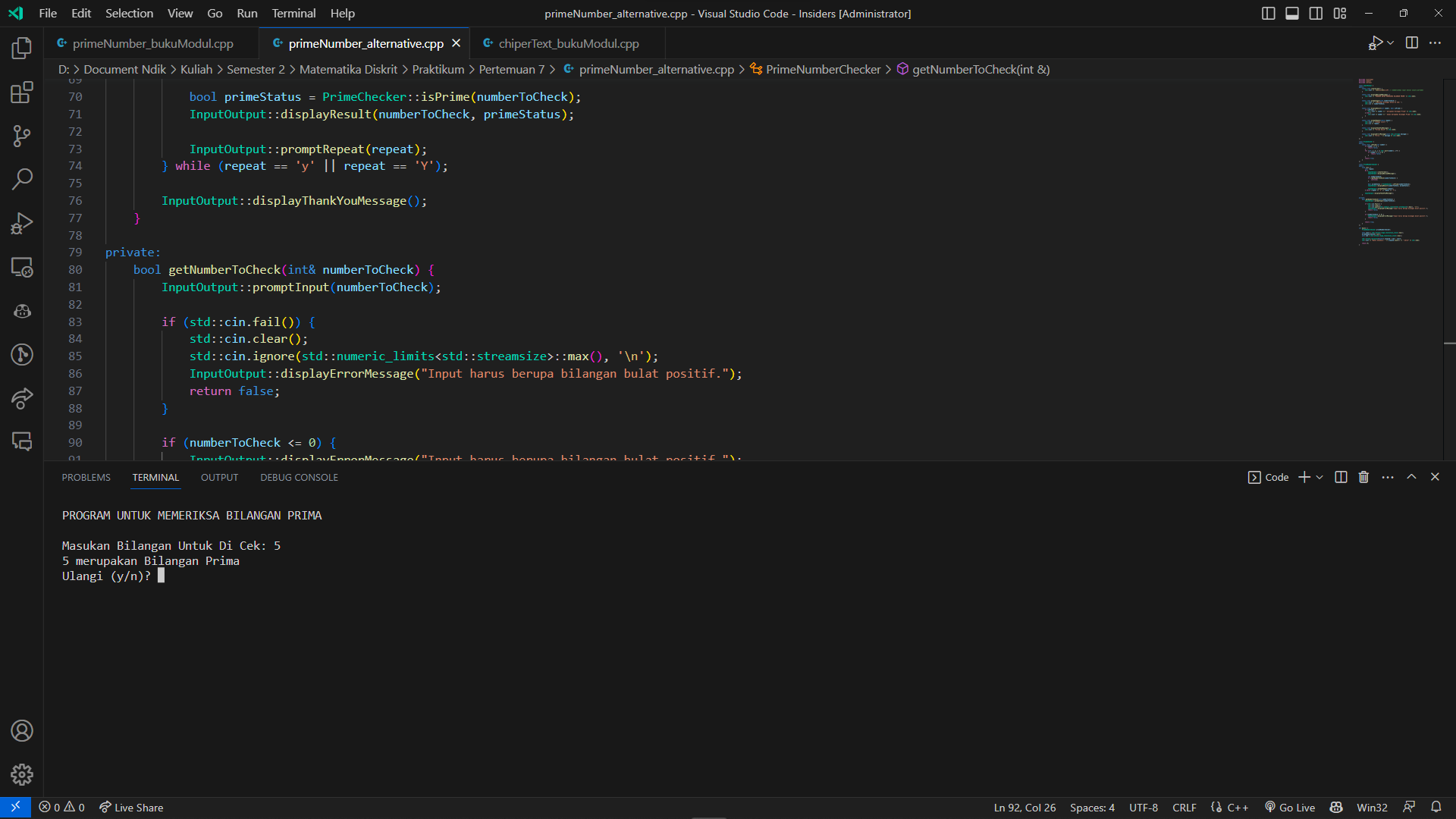
Berikut adalah outputnya:

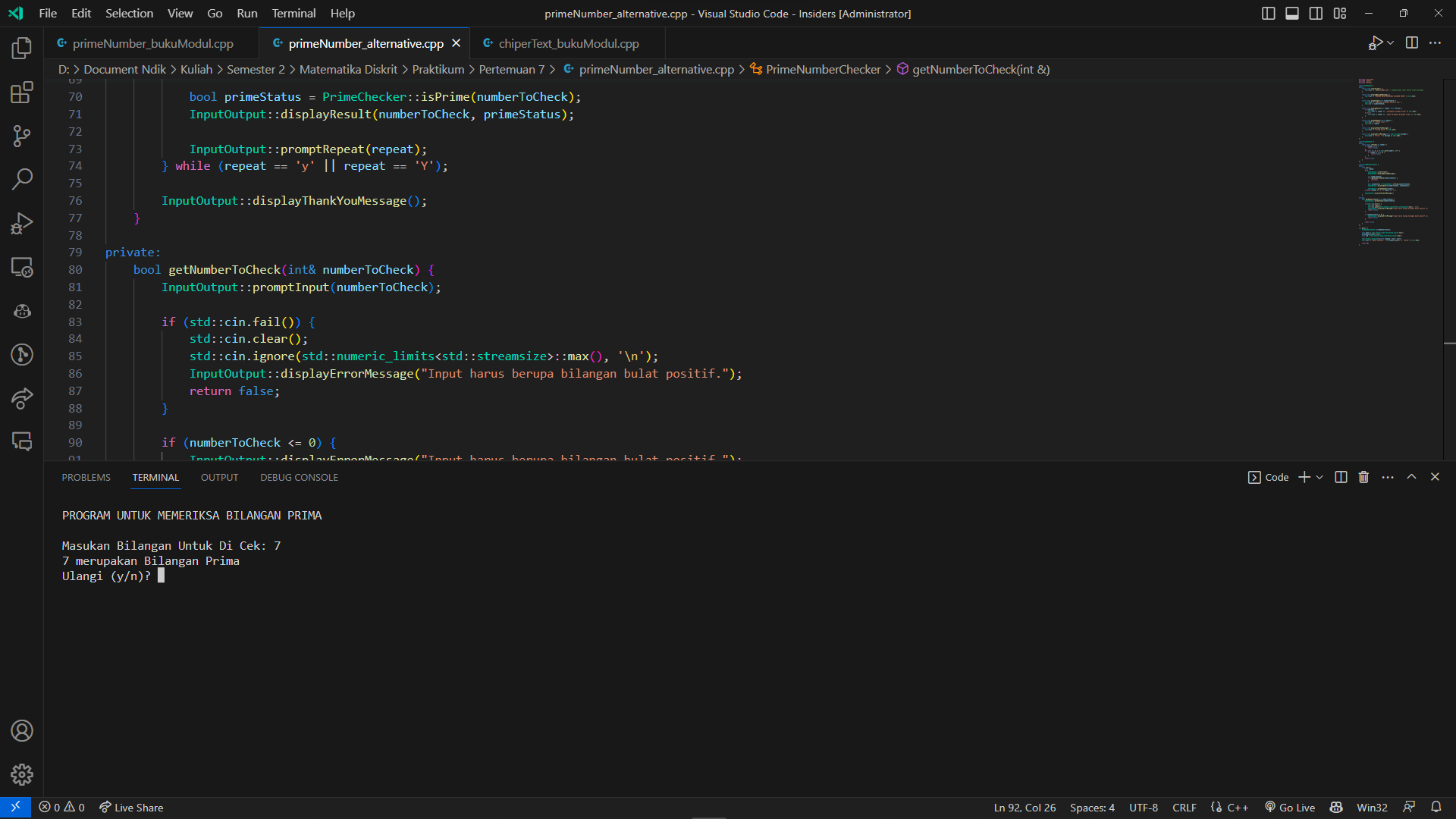


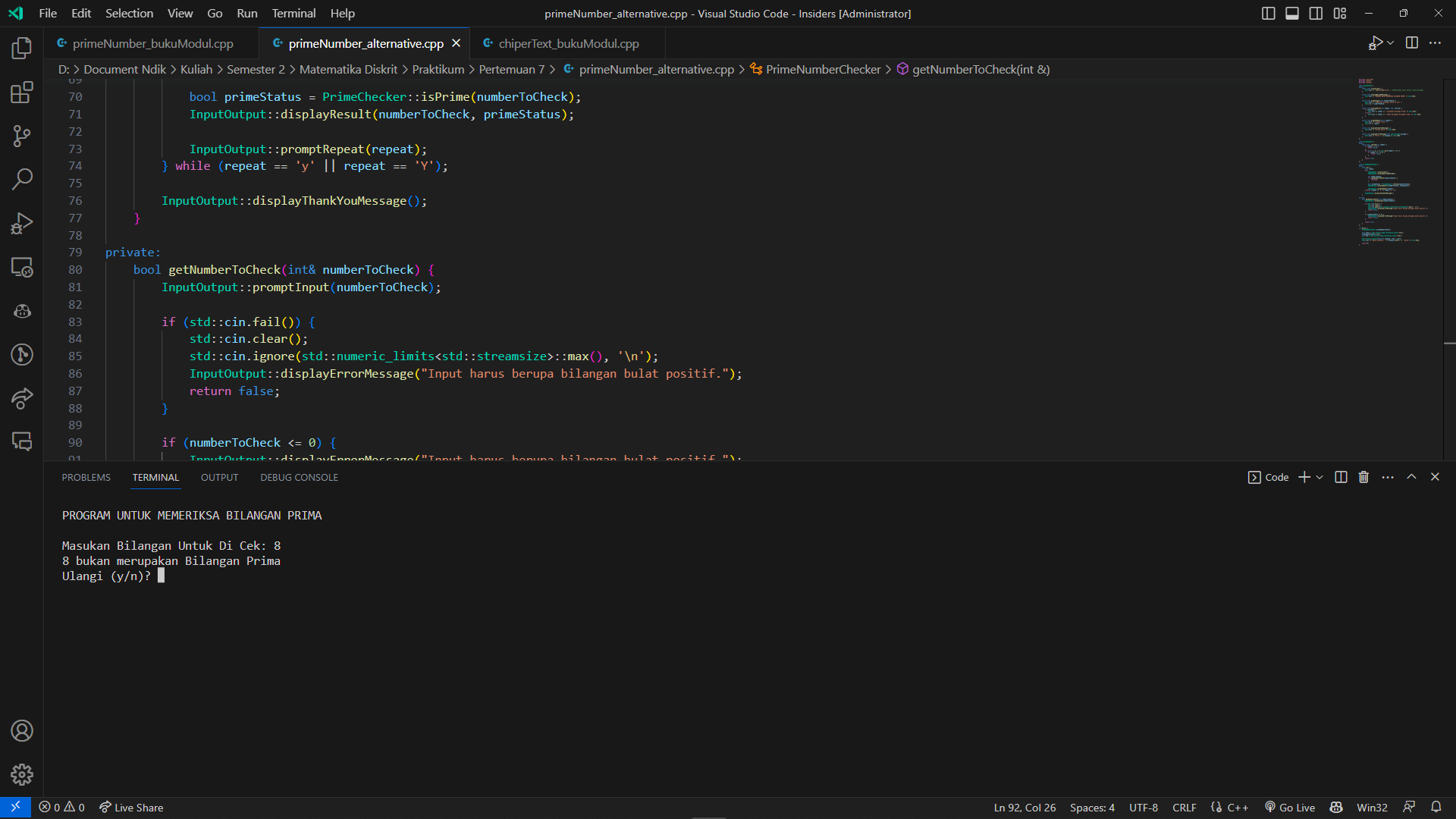


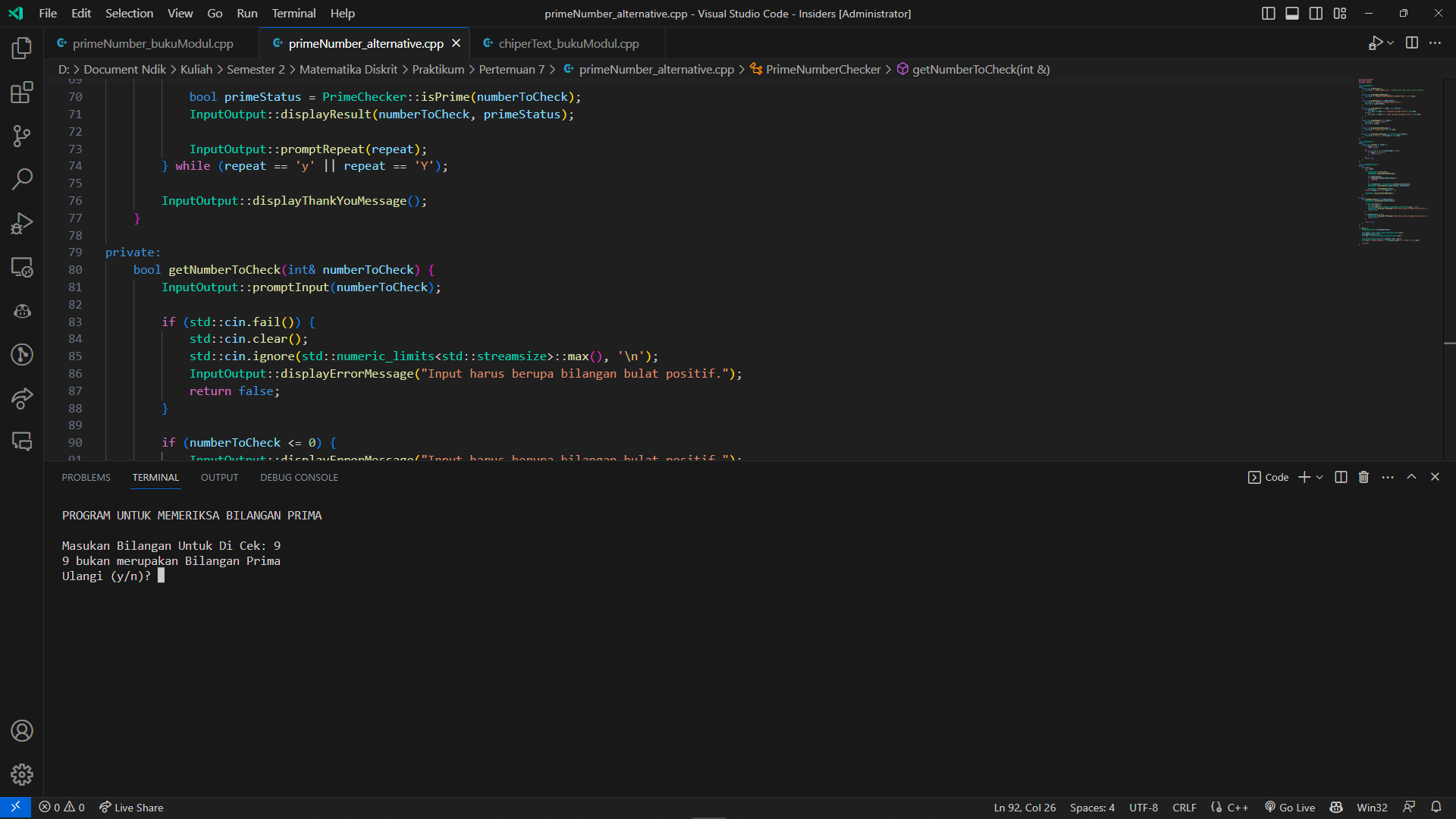


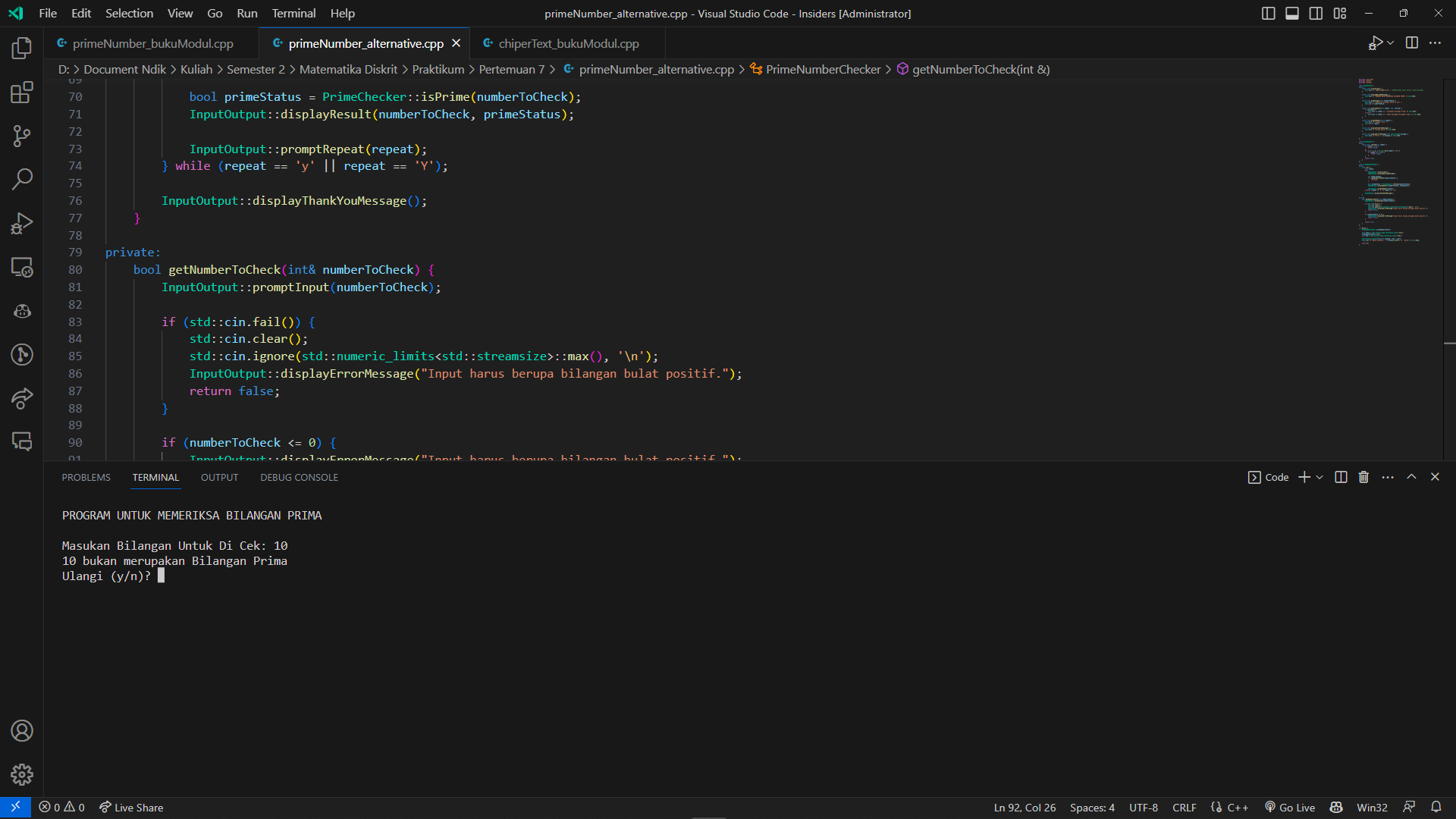




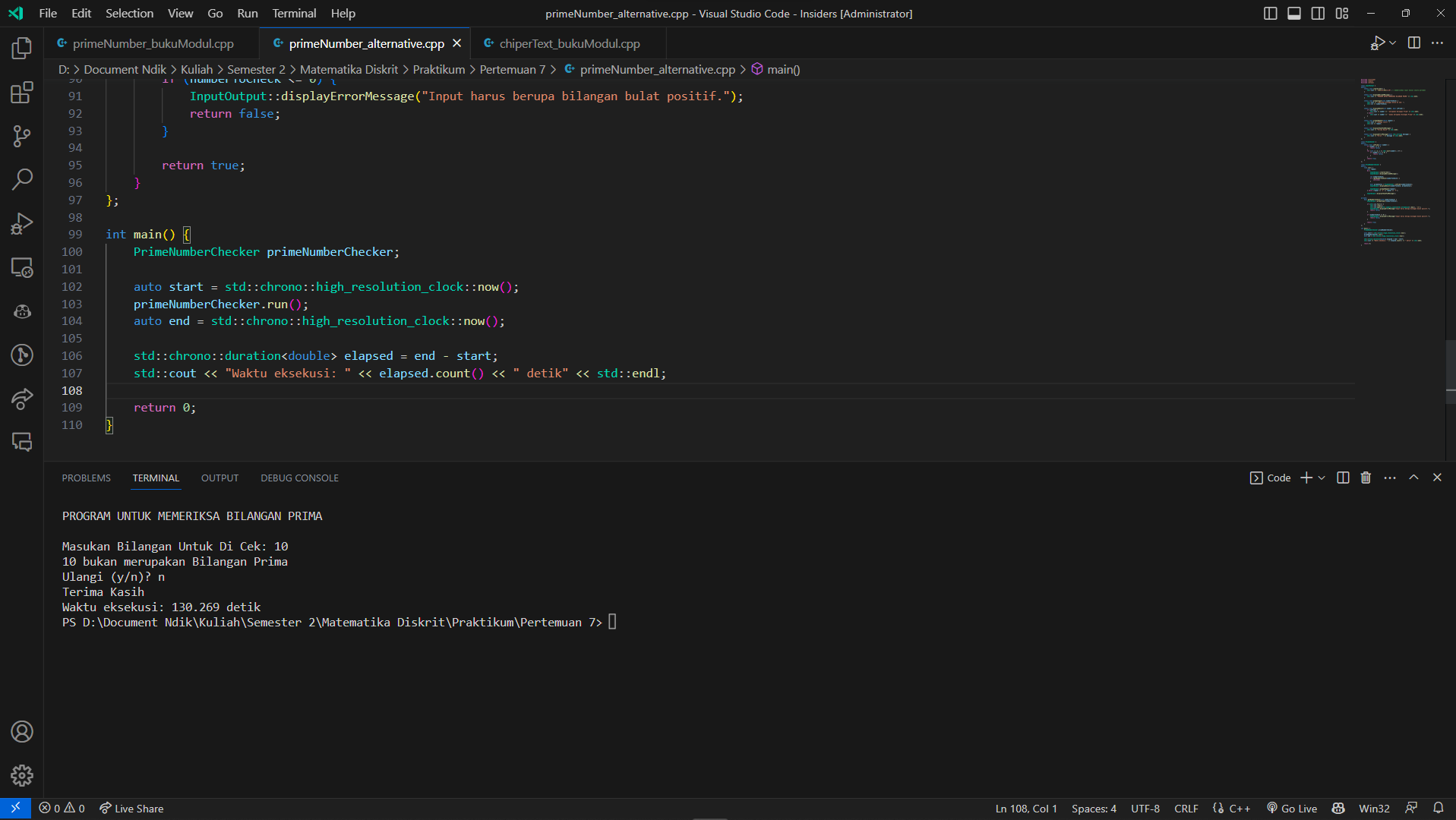






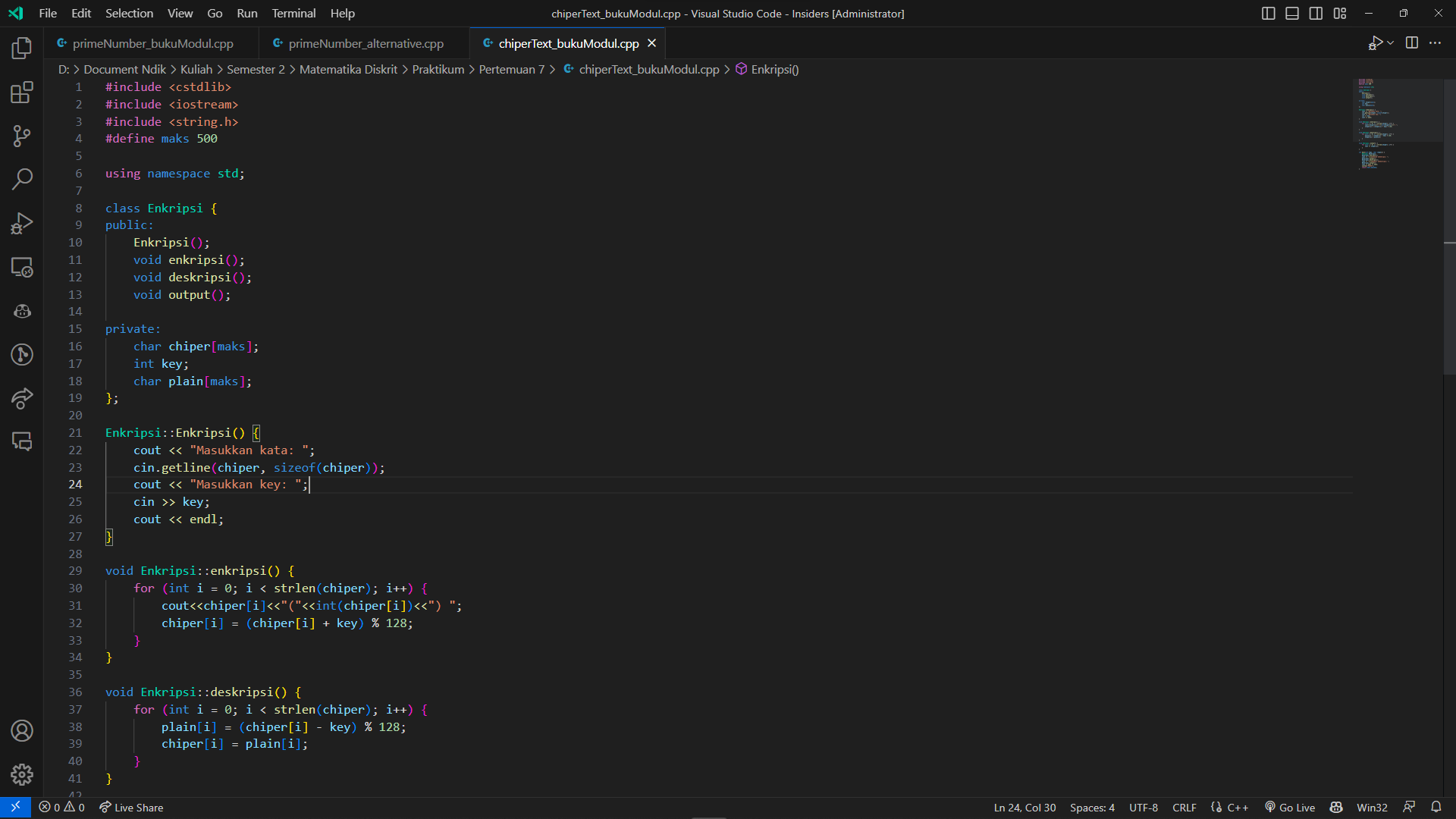


Gambar 7 Output hasil cek apakah bilangan prima atau bukan. (Sumber: Penulis)



Gambar 8 Tambahan hasil kecepatan eksekusi untuk mengecek perfoma kodingan. (Sumber: Penulis)

# **Program ChiperText**





Gambar 9 Source kodingan berdasarkan modul. (Sumber: Penulis)

Kode ini adalah program C++ yang melakukan enkripsi dan dekripsi pada teks yang dimasukkan oleh pengguna menggunakan metode Caesar cipher. Berikut adalah penjelasan rinci tentang kode tersebut:

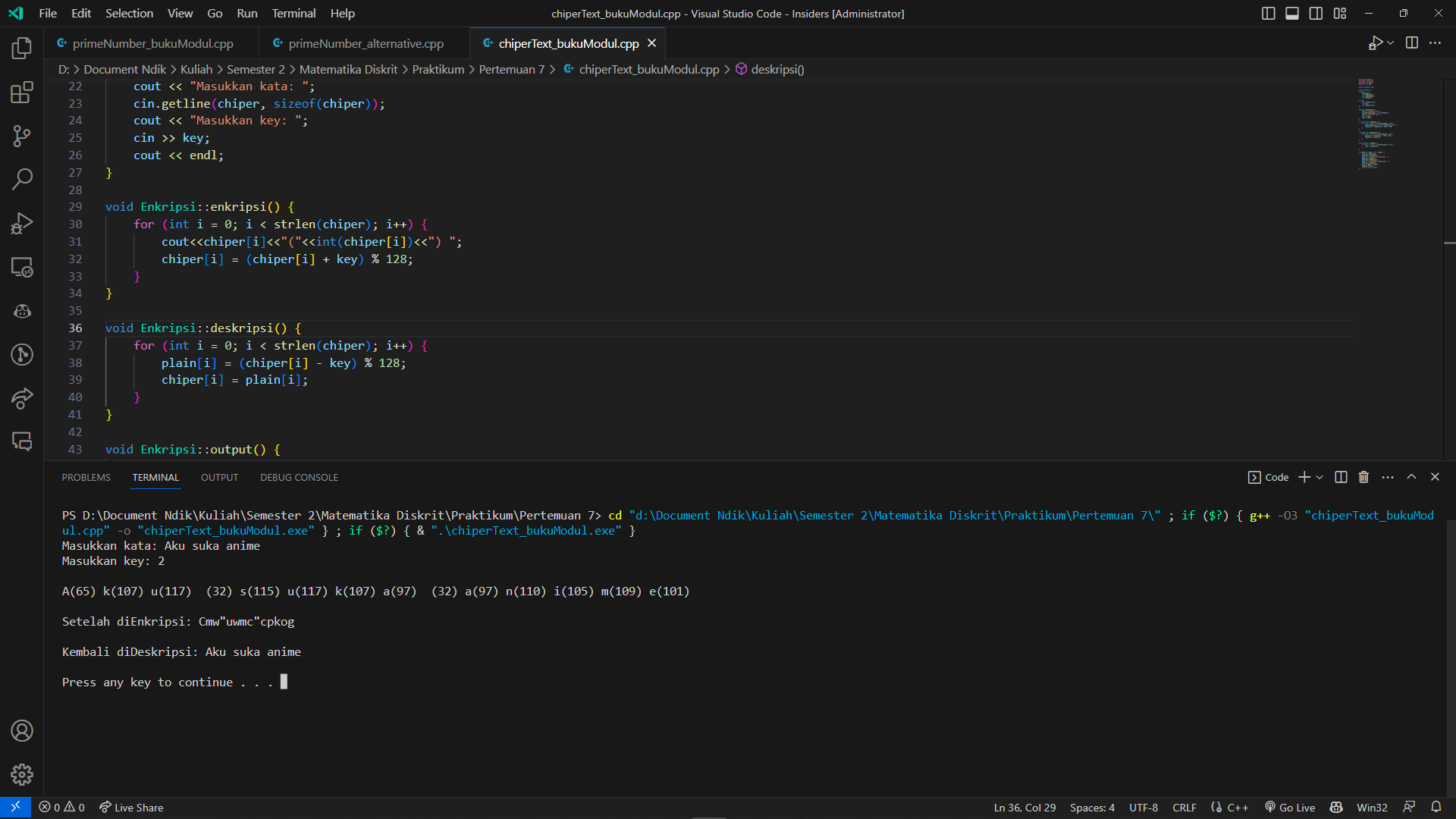
1. #include <cstdlib>, #include <iostream>, dan #include <string.h> adalah header files yang digunakan untuk mengakses fungsi-fungsi standar dalam C++.
2. #define maks 500 mendefinisikan konstanta bernama maks dengan nilai 500.
3. using namespace std; digunakan untuk mengakses fungsi dan objek dalam namespace std tanpa harus menambahkan prefix std::.
4. class Enkripsi mendefinisikan kelas bernama Enkripsi yang memiliki metode dan atribut berikut:

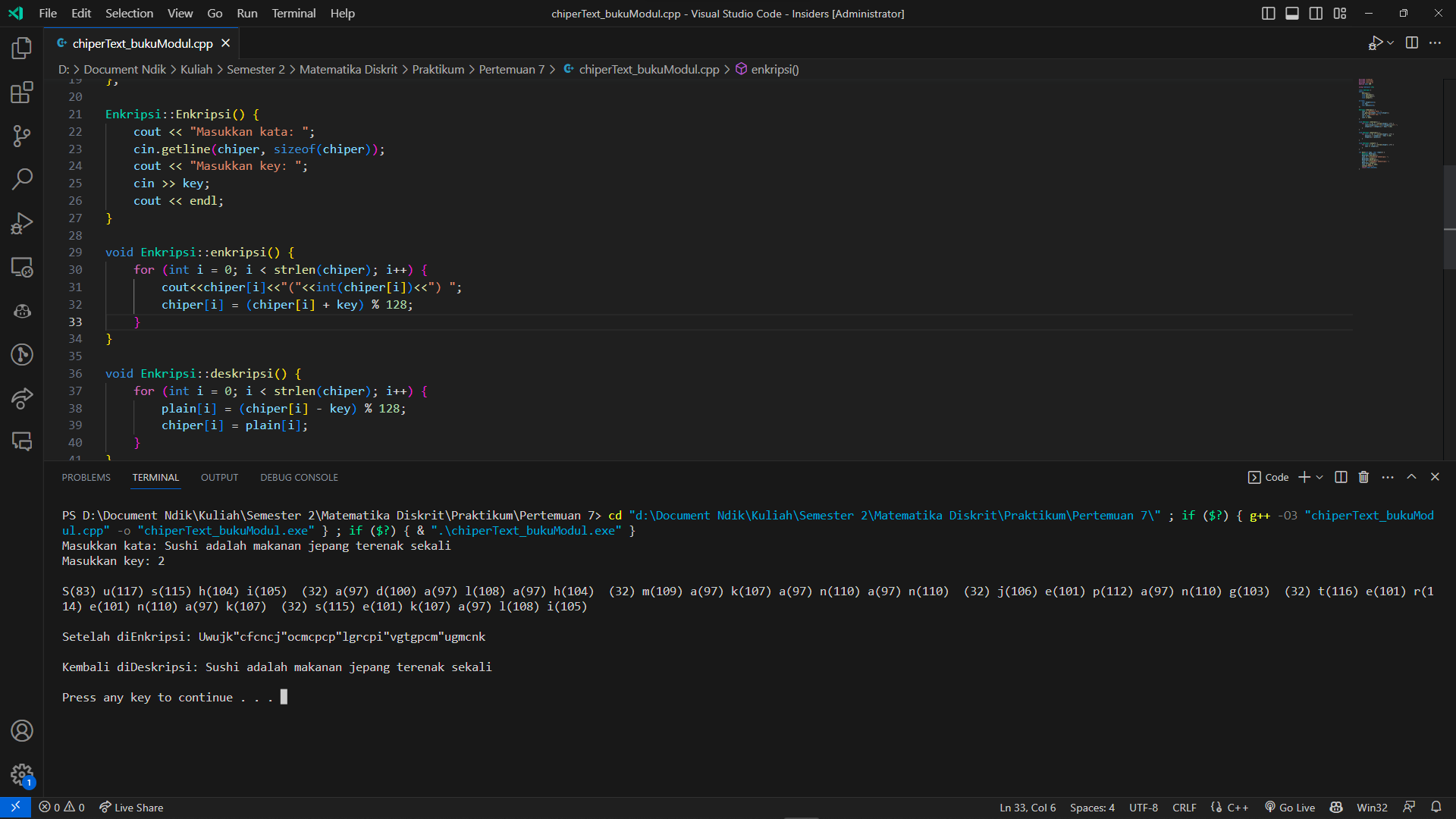
* Atribut chiper, key, dan plain yang digunakan untuk menyimpan teks yang akan dienkripsi, kunci enkripsi, dan teks yang telah didekripsi.
* Konstruktor Enkripsi() yang meminta pengguna untuk memasukkan teks dan kunci enkripsi.
* Metode enkripsi() yang mengenkripsi teks menggunakan metode Caesar cipher.
* Metode deskripsi() yang mendekripsi teks yang telah dienkripsi.
* Metode output() yang mencetak teks yang telah dienkripsi atau didekripsi.

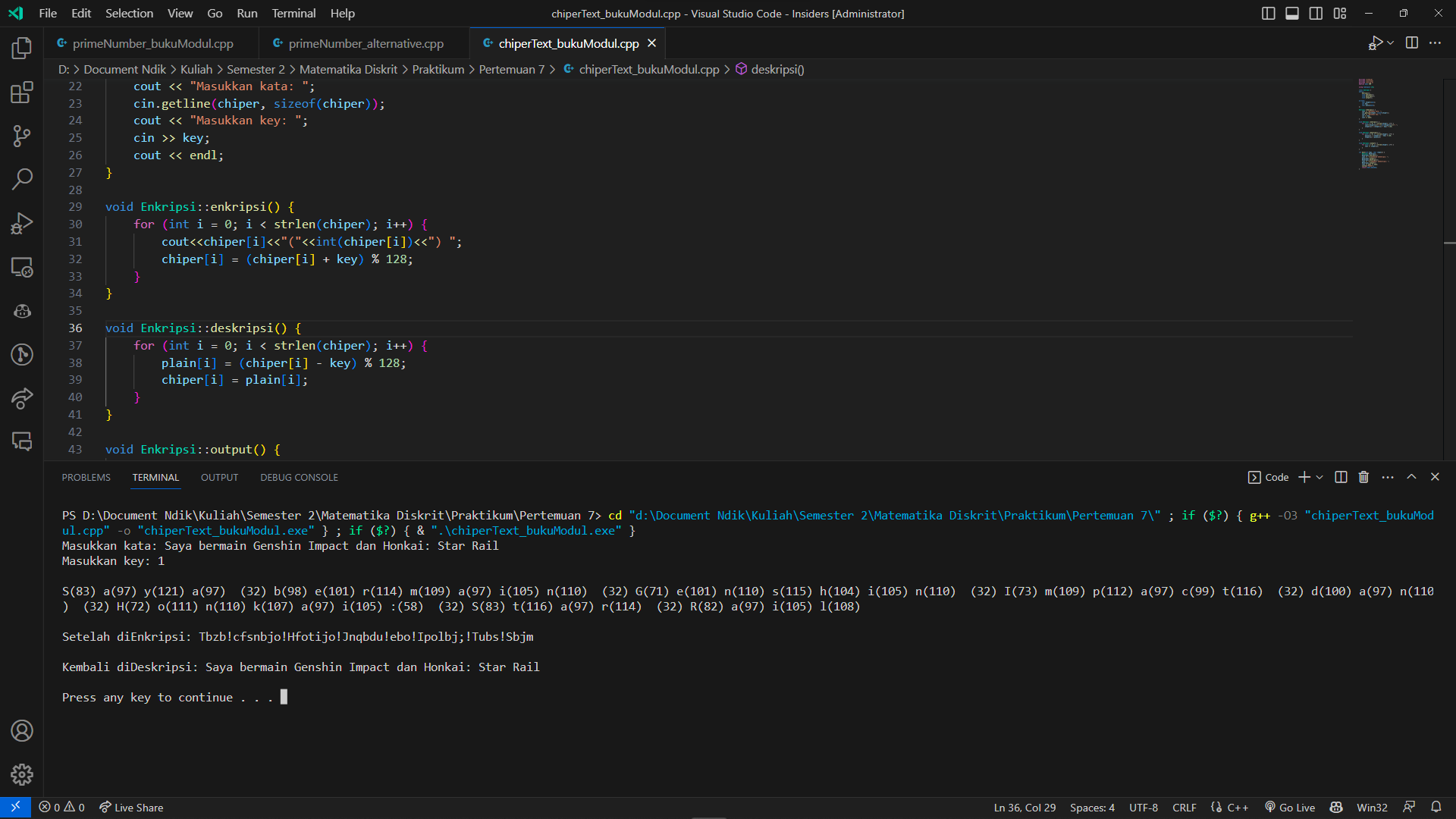
1. int main(int argc, char \*argv[]) adalah fungsi utama yang akan dieksekusi saat program dijalankan. Berikut adalah langkah-langkah yang dilakukan dalam fungsi main:

* Membuat objek Deskripsi dari kelas Enkripsi.
* Memanggil metode enkripsi() pada objek Deskripsi untuk mengenkripsi teks yang dimasukkan.
* Mencetak teks yang telah dienkripsi dengan memanggil metode output().
* Memanggil metode deskripsi() pada objek Deskripsi untuk mendekripsi teks yang telah dienkripsi.
* Mencetak teks yang telah didekripsi dengan memanggil metode output() lagi.
* Menjalankan perintah system("PAUSE") untuk memberi jeda sebelum program berakhir.
* Mengembalikan nilai EXIT\_SUCCESS untuk menandakan bahwa program berakhir dengan sukses.

Berikut adalah output kodingannya:







Gambar 10 Contoh kalimat yang ingin dienkripsi. (Sumber: Penulis)

Untuk mengakses hasil kodingan saya, dapat dilihat github berikut:

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