

LAPORAN PRAKTIKUM

“POST TEST 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:

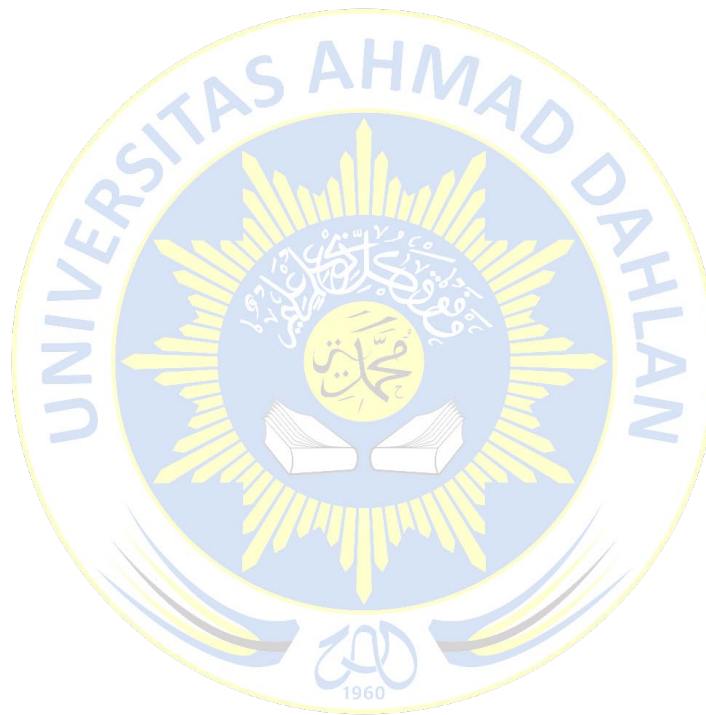
Mohammad Farid Hendianto 2200018401

Selasa 12.00-13.30

**PROGRAM STUDI INFORMATIKA
UNIVERSITAS AHMAD DAHLAN
FAKULTAS TEKNOLOGI INDUSTRI
TAHUN 2023**

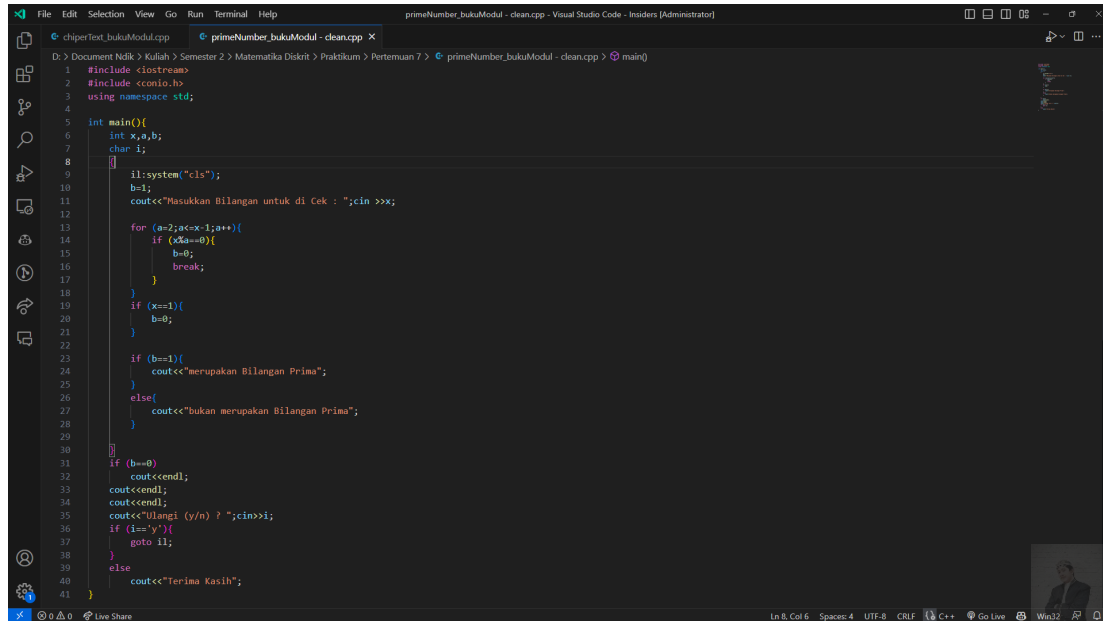
DAFTAR SOAL

1. Analisa lah program bilangan prima, lalu tunjukkan dan jelaskan pada code program bagian mana yang bertugas untuk menentukan bilangan tersebut prima atau bukan?..... 3
2. Analisa lah program Caesar Chiper, lalu tunjukkan dan jelaskan pada code program bagian mana yang bertugas untuk menjalankan proses enkripsi dan deskripsi?..... 11



1. Analisa lah program bilangan prima, lalu tunjukkan dan jelaskan pada code program bagian mana yang bertugas untuk menentukan bilangan tersebut prima atau bukan?

Sebelum saya memberikan bagian kutipan program dimana yang bertugas untuk menentukan bilangan tersebut prima atau bukan, berikut adalah full source kodingan sebelumnya untuk mencari bilangan tersebut apakah prima atau bukan berdasarkan modul yang sudah diperbaiki tanpa merubah bentuk kodingannya.



```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std;
4
5 int main(){
6     int x,a,b;
7     char i;
8
9     il:system("cls");
10    b=1;
11    cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
12
13    for (a=2; a<=x-1; a++){
14        if (x%a==0){
15            b=0;
16            break;
17        }
18    }
19    if (b==1){
20        b=0;
21    }
22
23    if (b==1){
24        cout<<"merupakan Bilangan Prima";
25    }
26    else{
27        cout<<"bukan merupakan Bilangan Prima";
28    }
29
30    if (b==0)
31        cout<<endl;
32    cout<<endl;
33    cout<<endl;
34    cout<<"Ulangi (y/n) ? ";cin>>i;
35    if (i=="y"){
36        goto il;
37    }
38    else
39        cout<<"Terima Kasih";
40
41

```

Gambar 1 Source Kodingan dari modul yang sudah diperbaiki. (Sumber: Penulis)

Berikut adalah analisis menurut algoritma yang digunakan dalam program tersebut untuk menentukan apakah suatu bilangan merupakan bilangan prima atau bukan:

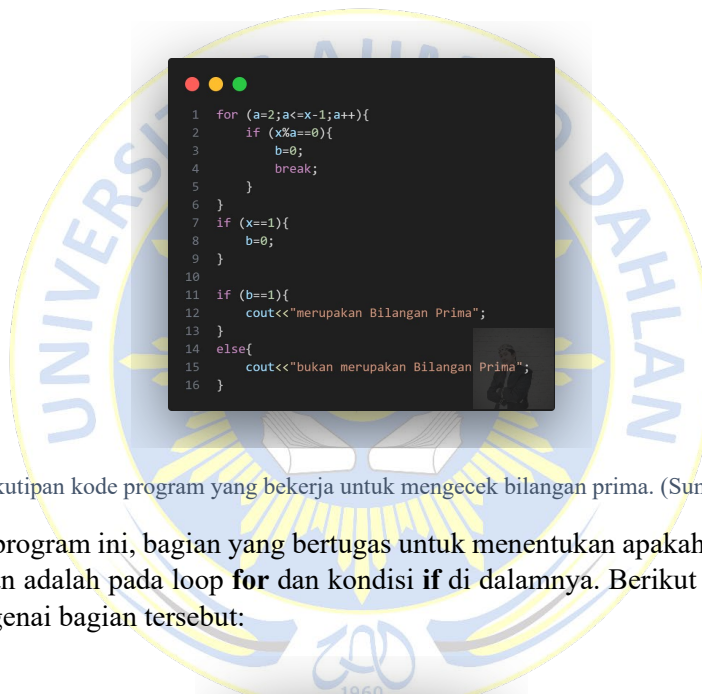
- 1) Inisialisasi variabel:
 - **x**: bilangan yang akan dicek apakah prima atau bukan.
 - **a**: variabel untuk iterasi dalam loop **for**.
 - **b**: variabel penanda yang akan digunakan untuk menentukan apakah **x** prima atau bukan. Inisialisasi dengan nilai **1**, yang mengindikasikan bahwa **x** dianggap prima hingga bukti sebaliknya ditemukan.
- 2) Input bilangan **x** yang akan dicek.
- 3) Cek apakah **x** sama dengan **1**. Jika ya, set **b** menjadi **0**, karena **1** bukan bilangan prima.
- 4) Lakukan loop **for** dengan variabel **a** mulai dari **2** hingga **x-1**:
 - a. Cek apakah **x** habis dibagi oleh **a** (yaitu, $x \% a == 0$).
 - b. Jika ya, set **b** menjadi **0**, karena **x** bukan bilangan prima, dan keluar dari loop menggunakan **break**.
- 5) Setelah loop selesai, cek nilai variabel **b**:
 - Jika **b** masih **1**, maka **x** merupakan bilangan prima.

- Jika **b** sudah berubah menjadi **0**, maka **x** bukan merupakan bilangan prima.
- 6) Tampilkan hasil apakah **x** merupakan bilangan prima atau bukan.

Algoritma ini menggunakan metode pembagian untuk menentukan apakah suatu bilangan merupakan bilangan prima atau bukan. Dalam metode ini, kita mencoba membagi bilangan yang akan dicek (**x**) dengan semua bilangan antara **2** dan **x-1**. Jika **x** tidak habis dibagi oleh bilangan tersebut, maka **x** adalah bilangan prima. Jika **x** habis dibagi oleh salah satu bilangan tersebut, maka **x** bukan bilangan prima.

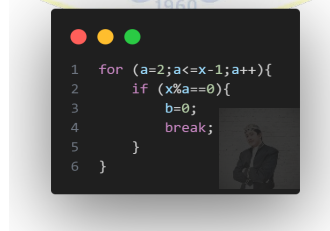
Dalam program yang diberikan, algoritma ini diimplementasikan menggunakan loop **for** dan kondisi **if** di dalamnya, serta pengecekan nilai **b** setelah loop selesai.

Berikut adalah kutipan kode untuk mengecek bilangan prima.



Gambar 2 kutipan kode program yang bekerja untuk mengecek bilangan prima. (Sumber: Penulis)

Dalam program ini, bagian yang bertugas untuk menentukan apakah bilangan tersebut prima atau bukan adalah pada loop **for** dan kondisi **if** di dalamnya. Berikut adalah penjelasan lebih rinci mengenai bagian tersebut:

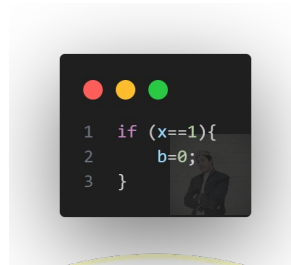


Gambar 3_Perulangan for. (Sumber: Penulis)

Loop **for** ini dimulai dengan nilai **a** yang diinisialisasi menjadi **2** dan berlanjut hingga **x-1**. Pada setiap iterasi, program akan memeriksa apakah **x** habis dibagi oleh **a** dengan menggunakan operator modulo (%). Jika **x % a == 0**, maka bilangan **x** dapat dibagi habis oleh

a, sehingga bukan merupakan bilangan prima. Dalam hal ini, variabel **b** diatur menjadi **0** dan loop dihentikan menggunakan perintah **break**.

Setelah loop **for** selesai, program akan memeriksa apakah **x** sama dengan **1**. Jika ya, maka **b** diatur menjadi **0**, karena **1** bukan merupakan bilangan prima.



Gambar 4 kondisi dimana apabila user memasukkan angka satu maka bukan termasuk angka prima. (Sumber: Penulis)

Kemudian, program akan memeriksa nilai dari variabel **b**. Jika **b** masih bernilai **1**, maka bilangan **x** merupakan bilangan prima. Jika **b** sudah berubah menjadi **0**, maka bilangan **x** bukan merupakan bilangan prima.



Gambar 5 Pengecekan terakhir nilai **b** untuk menentukan output apakah itu bilangan prima atau bukan. (Sumber: Penulis)

Dengan demikian, bagian yang bertanggung jawab untuk menentukan apakah bilangan tersebut prima atau bukan adalah loop **for** dan kondisi **if** di dalamnya, serta pengecekan nilai **b** setelah loop selesai.

Berikut adalah outputnya.

```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main() { // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

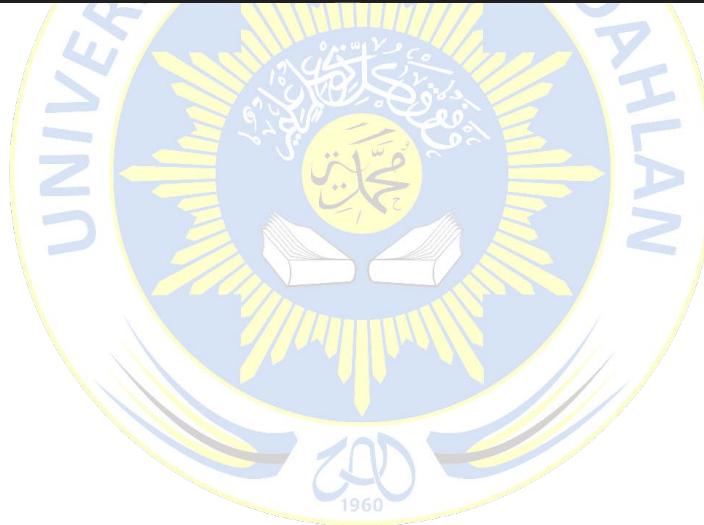
```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE

Masukkan Bilangan untuk di Cek : 3
merupakan Bilangan Prima

Ulangi (y/n) ?

Ln 17, Col 14 Spaces: 4 UTF-8 CRLF C++ Go Live Win32



```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main(){ // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE

Masukkan Bilangan untuk di Cek : 2
merupakan Bilangan Prima

Ulangi (y/n) ?

Ln 21, Col 10 Spaces: 4 UTF-8 CRLF C++ Go Live Win32

```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main(){ // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE

Masukkan Bilangan untuk di Cek : 4
bukan merupakan Bilangan Prima

Ulangi (y/n) ?

Ln 17, Col 14 Spaces: 4 UTF-8 CRLF C++ Go Live Win32

```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main(){ // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

```

Masukkan Bilangan untuk di Cek : 5
merupakan Bilangan Prima

Ulangi (y/n) ?

```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main(){ // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

```

Masukkan Bilangan untuk di Cek : 6
bukan merupakan Bilangan Prima

Ulangi (y/n) ?


```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main(){ // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE

Masukkan Bilangan untuk di Cek : 7
merupakan Bilangan Prima

Ulangi (y/n) ?

```

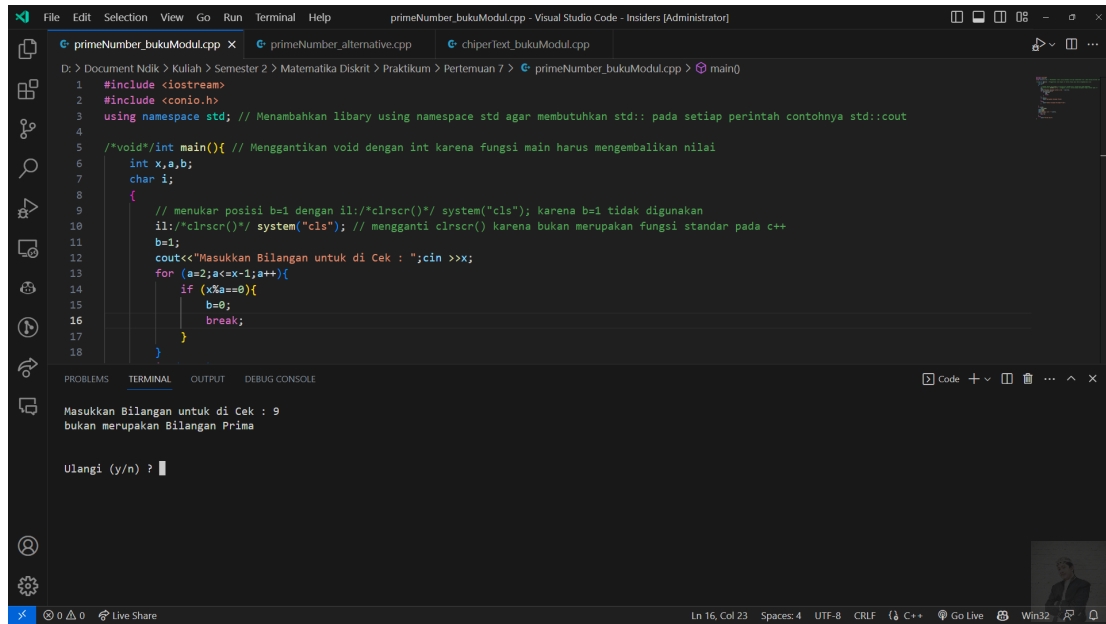
1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main(){ // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE

Masukkan Bilangan untuk di Cek : 8
bukan merupakan Bilangan Prima

Ulangi (y/n) ?



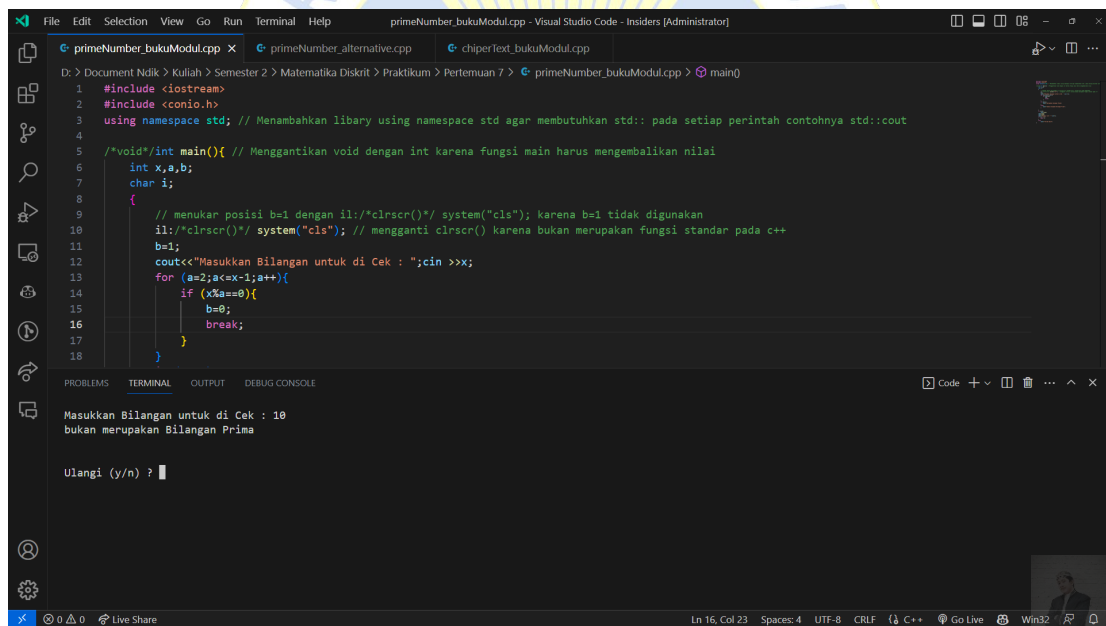
```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main(){ // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

```

Masukkan Bilangan untuk di Cek : 9
bukan merupakan Bilangan Prima

Ulangi (y/n) ?



```

1 #include <iostream>
2 #include <conio.h>
3 using namespace std; // Menambahkan library using namespace std agar membutuhkan std:: pada setiap perintah contohnya std::cout
4
5 /*void*/int main(){ // Menggantikan void dengan int karena fungsi main harus mengembalikan nilai
6     int x,a,b;
7     char i;
8     {
9         // menukar posisi b=1 dengan il:/*clrscr()*/ system("cls"); karena b=1 tidak digunakan
10        il:/*clrscr()*/ system("cls"); // mengganti clrscr() karena bukan merupakan fungsi standar pada c++
11        b=1;
12        cout<<"Masukkan Bilangan untuk di Cek : ";cin >>x;
13        for (a=2;a<=x-1;a++){
14            if (x%a==0){
15                b=0;
16                break;
17            }
18        }
19    }
20 }

```

Masukkan Bilangan untuk di Cek : 10
bukan merupakan Bilangan Prima

Ulangi (y/n) ?

Gambar 6 Output hasil cek bilangan Prima

2. Analisa lah program Caesar Chiper, lalu tunjukkan dan jelaskan pada code program bagian mana yang bertugas untuk menjalankan proses enkripsi dan deskripsi?

```

1 #include <cstdlib>
2 #include <iostream>
3 #include <string.h>
4 #define maks 500
5
6 using namespace std;
7
8 class Enkripsi {
9 public:
10     Enkripsi();
11     void enkripsi();
12     void deskripsi();
13     void output();
14
15 private:
16     char chiper[maks];
17     int key;
18     char plain[maks];
19 };
20
21 Enkripsi::Enkripsi() {
22     cout << "Masukkan kata: ";
23     cin.getline(chiper, sizeof(chiper));
24     cout << "Masukkan key: ";
25     cin >> key;
26     cout << endl;
27 }
28
29 void Enkripsi::enkripsi() {
30     for (int i = 0; i < strlen(chiper); i++) {
31         cout<<chiper[i]<<(" " <<int(chiper[i])<<(" "));
32         chiper[i] = (chiper[i] + key) % 128;
33     }
34 }
35
36 void Enkripsi::deskripsi() {
37     for (int i = 0; i < strlen(chiper); i++) {
38         plain[i] = (chiper[i] - key) % 128;
39         chiper[i] = plain[i];
40     }
41 }
42
43 void Enkripsi::output() {
44     for (int i = 0; i < strlen(chiper); i++) {
45         cout << chiper[i];
46     }
47 }
48
49 int main(int argc, char *argv[]) {
50     Enkripsi enkripsi;
51     enkripsi.enkripsi();
52     cout << "\n\nSetelah diEnkripsi: ";
53     enkripsi.output();
54     enkripsi.deskripsi();
55     cout << "\n\nEmail diDeskripsi: ";
56     enkripsi.output();
57     cout << endl << endl;
58     system("PAUSE");
59     return EXIT_SUCCESS;
60 }

```

Gambar 7 Source kodingan berdasarkan modul. (Sumber: Penulis)

Chiper teks adalah teks yang dihasilkan setelah proses enkripsi pada teks asli (plain teks) menggunakan suatu algoritma kriptografi. Dalam konteks program Caesar Cipher ini, chiper teks adalah teks yang dihasilkan setelah setiap karakter dalam teks asli dienkripsi dengan menambahkan suatu nilai kunci (key) tertentu.

Algoritma chiper teks pada program Caesar Cipher ini adalah sebagai berikut:

- 1) Pengguna memasukkan teks asli (plain teks) dan kunci (key) yang akan digunakan untuk enkripsi.
- 2) Program akan mengiterasi setiap karakter dalam teks asli.
- 3) Untuk setiap karakter, program akan menambahkan kunci (key) ke nilai ASCII karakter tersebut.
- 4) Hasil penjumlahan tersebut akan dimodulo dengan 128 untuk memastikan karakter hasil enkripsi tetap dalam rentang ASCII yang valid.
- 5) Karakter hasil enkripsi akan menggantikan karakter asli dalam teks, sehingga menghasilkan chiper teks.
- 6) Proses ini diimplementasikan dalam fungsi void Enkripsi::enkripsi() pada kode program yang telah

Program Caesar Cipher ini menggunakan bahasa pemrograman C++. Berikut adalah penjelasan mengenai bagian kode yang bertugas untuk menjalankan proses enkripsi dan deskripsi:

1) Enkripsi (*Encryption*):

Proses enkripsi terjadi dalam fungsi **void Enkripsi::enkripsi()**. Fungsi ini mengiterasi setiap karakter dalam string **chiper** dan mengenkripsi karakter tersebut dengan menambahkan **key** yang telah dimasukkan oleh pengguna. Setelah itu, hasil penjumlahan dimodulo dengan 128 untuk memastikan karakter hasil enkripsi tetap dalam rentang ASCII yang valid. Ingat, rumus enkripsi ASCII (bukan extended ASCII) untuk adalah sebagai berikut:

Berikut adalah potongan kode yang menjalankan enkripsi:

$$\text{Dekripsi: } c_i = D(p_i) = (p_i - \text{key}) \bmod 128$$



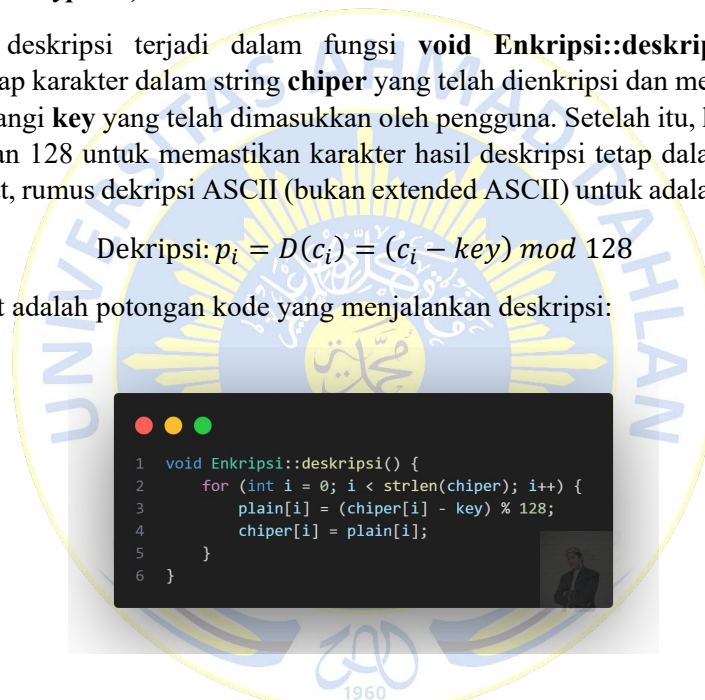
Gambar 8 fungsi enkripsi() pada class Enkripsi. (Sumber: Penulis)

2) Deskripsi (Decryption):

Proses deskripsi terjadi dalam fungsi **void Enkripsi::deskripsi()**. Fungsi ini mengiterasi setiap karakter dalam string **chiper** yang telah dienkripsi dan mendeskripsikannya dengan mengurangi **key** yang telah dimasukkan oleh pengguna. Setelah itu, hasil pengurangan dimodulo dengan 128 untuk memastikan karakter hasil deskripsi tetap dalam rentang ASCII yang valid. Ingat, rumus dekripsi ASCII (bukan extended ASCII) untuk adalah sebagai berikut:

$$\text{Dekripsi: } p_i = D(c_i) = (c_i - \text{key}) \bmod 128$$

Berikut adalah potongan kode yang menjalankan deskripsi:



Gambar 9 fungsi deskripsi() pada class Enkripsi. (Sumber: Penulis)

Dalam program ini, enkripsi dan deskripsi dijalankan secara berurutan dalam fungsi **main()**. Pertama, objek **Enkripsi** bernama **Deskripsi** dibuat, kemudian fungsi **enkripsi()** dipanggil untuk mengenkripsi teks, dan hasilnya ditampilkan. Selanjutnya, fungsi **deskripsi()** dipanggil untuk mendeskripsikan teks yang telah dienkripsi, dan hasilnya ditampilkan kembali.

Berikut adalah contoh outputnya

```

20
21 Enkripsi::enkripsi() {
22     cout << "Masukkan kata: ";
23     cin.getline(chiper, sizeof(chiper));
24     cout << "Masukkan key: ";
25     cin >> key;
26     cout << endl;
27 }
28
29 void Enkripsi::enkripsi() {
30     for (int i = 0; i < strlen(chiper); i++) {
31         cout<<chiper[i]<<(" " <<int(chiper[i])<<(" "));
32         chiper[i] = (chiper[i] + key) % 128;
33     }
34 }
35
36 void Enkripsi::deskripsi() {
37     for (int i = 0; i < strlen(chiper); i++) {
38         plain[i] = (chiper[i] - key) % 128;
39         chiper[i] = plain[i];
40     }
41 }

```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE

```

PS C:\Users\Ndik> cd "d:\Document Ndik\Kuliah\Semester 2\Matematika Diskrit\Praktikum\Pertemuan 7"; if ($?) { g++ -O3 "chiperText_bukuModul.cpp" -o "chiperText_bukuModul.exe" }; if ($?) { & ".\chiperText_bukuModul.exe" }
Masukkan kata: Matdis menyenangkan
Masukkan key: -2
M(77) a(97) t(116) d(100) i(105) s(115) (32) m(109) e(101) n(110) y(121) e(101) n(110) a(97) n(110) g(103) k(107) a(97) n(110)
Setelah diEnkripsi: K_rhpgakclwcl_lai_l
Kembali diDeskripsi: Matdis menyenangkan
Press any key to continue . . .

```

```

22
23     cout << "Masukkan kata: ";
24     cin.getline(chiper, sizeof(chiper));
25     cout << "Masukkan key: ";
26     cin >> key;
27     cout << endl;
28 }
29
30 void Enkripsi::enkripsi() {
31     for (int i = 0; i < strlen(chiper); i++) {
32         cout<<chiper[i]<<(" " <<int(chiper[i])<<(" "));
33         chiper[i] = (chiper[i] + key) % 128;
34     }
35 }
36
37 void Enkripsi::deskripsi() {
38     for (int i = 0; i < strlen(chiper); i++) {
39         plain[i] = (chiper[i] - key) % 128;
40         chiper[i] = plain[i];
41     }
42 }
43 void Enkripsi::output() {

```

PROBLEMS TERMINAL OUTPUT DEBUG CONSOLE

```

PS D:\Document Ndik\Kuliah\Semester 2\Matematika Diskrit\Praktikum\Pertemuan 7> cd "d:\Document Ndik\Kuliah\Semester 2\Matematika Diskrit\Praktikum\Pertemuan 7"; if ($?) { g++ -O3 "chiperText_bukuModul.cpp" -o "chiperText_bukuModul.exe" }; if ($?) { & ".\chiperText_bukuModul.exe" }
Masukkan kata: Aku suka anime
Masukkan key: 2
A(65) k(107) u(117) (32) s(115) u(117) k(107) a(97) (32) a(97) n(110) i(105) m(109) e(101)
Setelah diEnkripsi: Cmw'uamc"cpkog
Kembali diDeskripsi: Aku suka anime
Press any key to continue . . .

```

```

1  primeNumber_bukuModul.cpp
2  primeNumber_alternative.cpp
3  chiperText_bukuModul.cpp X
4
5  D:\Document Ndik\Kuliah\Semester 2\Matematika Diskrit\Praktikum\Pertemuan 7> cd "d:\Document Ndik\Kuliah\Semester 2\Matematika Diskrit\Praktikum\Pertemuan 7" ; if ($?) { g++ -O3 "chiperText_bukuModul.cpp" -o "chiperText_bukuModul.exe" } ; if ($?) { & ".\chiperText_bukuModul.exe" }
6  Masukkan kata: "
7  Masukkan key: "
8  Masukkan key: 1
9
10 S(83) a(97) y(121) a(97) (32) b(98) e(101) r(114) m(109) a(97) i(105) n(110) (32) 6(71) e(101) n(110) s(115) h(104) i(105) n(110) (32) I(73) m(109) p(112) a(97) c(99) t(116) (32) d(100) a(97) n(110)
11 (32) H(72) o(111) n(110) k(107) a(97) i(105) : (58) (32) S(83) t(116) a(97) r(114) (32) R(82) a(97) i(105) l(108)
12
13 Setelah diEnkripsi: TbzbfcsnbjofHfotjofJnqduleboIpoibj;ITubsISbjw
14
15 Kembali diDeskripsi: Saya bermain Genshin Impact dan Honkai: Star Rail
16
17 Press any key to continue . . .

```

```

1  chiperText_bukuModul.cpp X
2
3  D:\Document Ndik\Kuliah\Semester 2\Matematika Diskrit\Praktikum\Pertemuan 7> cd "d:\Document Ndik\Kuliah\Semester 2\Matematika Diskrit\Praktikum\Pertemuan 7" ; if ($?) { g++ -O3 "chiperText_bukuModul.cpp" -o "chiperText_bukuModul.exe" } ; if ($?) { & ".\chiperText_bukuModul.exe" }
4  Masukkan kata: Sushi adalah makanan jepang terenak sekali
5  Masukkan key: "
6  Masukkan key: 2
7
8 S(83) a(117) s(115) h(104) i(105) (32) a(97) d(100) a(97) l(108) a(97) h(104) (32) m(109) a(97) k(107) a(97) n(110) a(97) n(110) (32) j(106) e(101) p(112) a(97) n(110) g(103) (32) t(116) e(101) r(114)
9 a(101) n(110) a(97) k(107) (32) s(115) a(101) k(107) a(97) l(108) i(105)
10
11 Setelah diEnkripsi: Uwujk*cfncj*ocmcpclgrcpil*vgtpcm*ugcnk
12
13 Kembali diDeskripsi: Sushi adalah makanan jepang terenak sekali
14
15 Press any key to continue . . .

```

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

Untuk mengakses kodingan, dapat melihat link github berikut.

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