

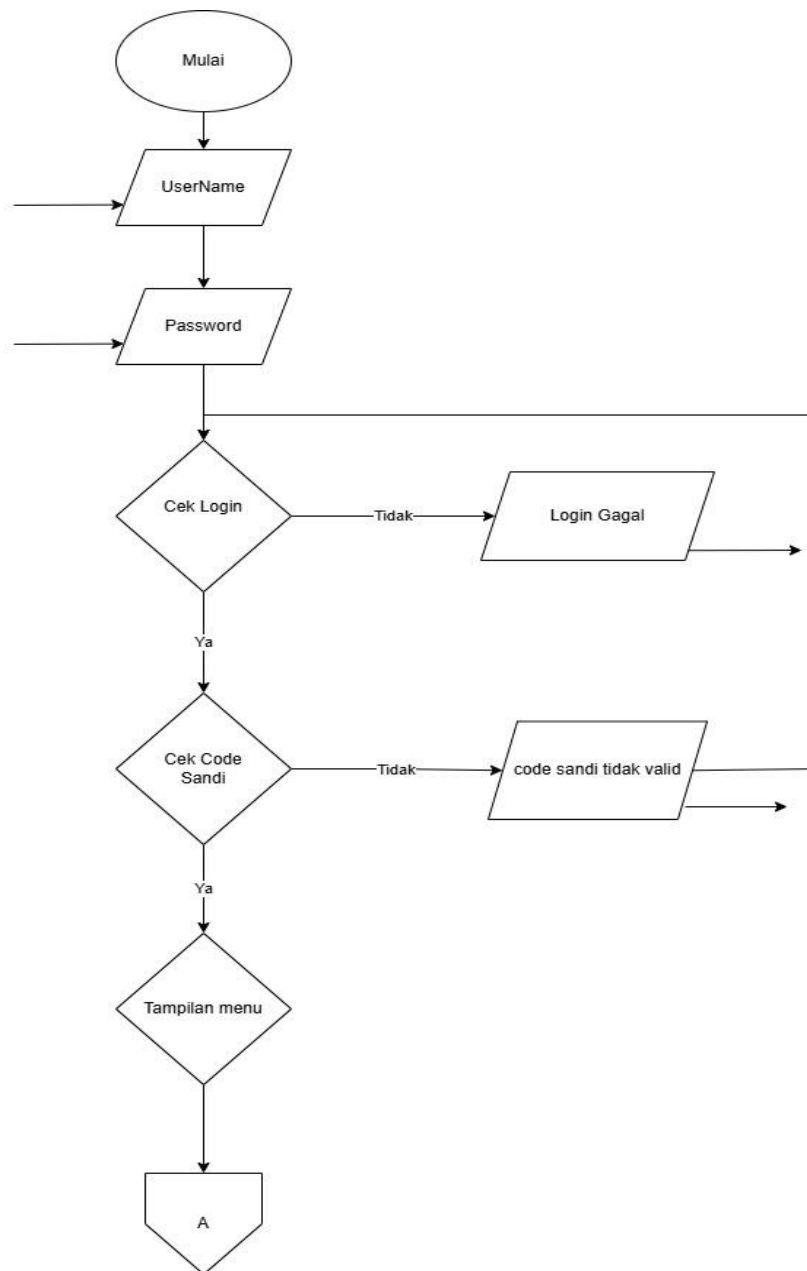
LAPORAN PRAKTIKUM
POSTTEST 1
ALGORITMA PEMROGRAMAN LANJUT

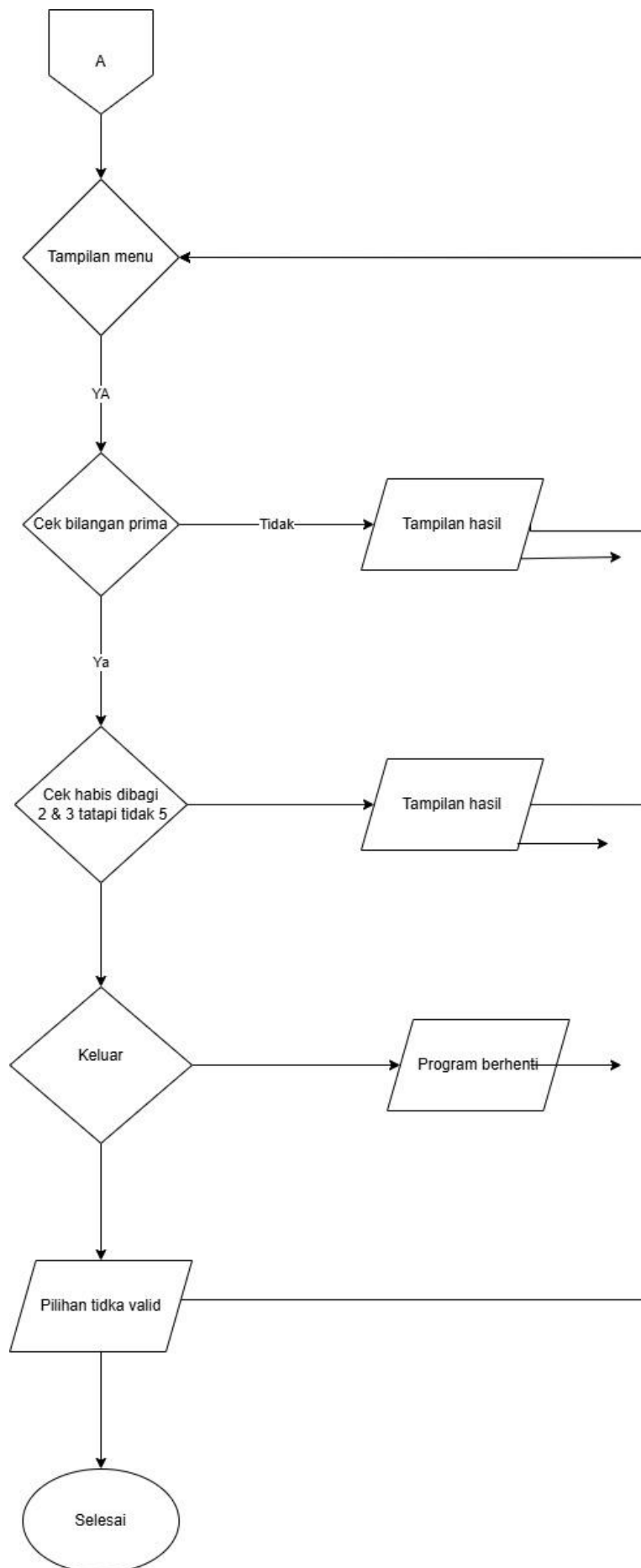


Disusun oleh:
Ade Pasiha Tangke Allo (2409106209)
Kelas (C2 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart





2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini bertujuan untuk memverifikasi login pengguna menggunakan username dan password yang telah ditentukan. Setelah berhasil login, pengguna diminta memasukkan kode sandi dalam rentang tertentu. Program kemudian memberikan dua fitur utama, yaitu mengecek apakah kode sandi tersebut merupakan bilangan prima atau apakah bilangan tersebut habis dibagi 2 dan 3 tetapi tidak oleh 5. Pengguna dapat memilih salah satu dari dua opsi tersebut dan program akan memberikan hasilnya. Program akan terus berjalan hingga pengguna memilih untuk keluar.

2.2. Penjelasan Alur & Algoritma

1. Algoritma

Program ini merupakan sistem login yang memungkinkan pengguna untuk memasukkan kode sandi dan mengecek beberapa sifat matematis dari angka tersebut. Berikut adalah alur lengkapnya:

1. Login Pengguna:

- Program meminta pengguna memasukkan username dan password (NIM).
- Jika username atau password salah, program akan menampilkan pesan error dan langsung berhenti.
- Jika login berhasil, program melanjutkan ke tahap berikutnya.

2. Memasukkan Kode Sandi:

- Program meminta pengguna memasukkan kode sandi dalam rentang 100 hingga 99999.
- Jika kode sandi tidak valid (di luar rentang tersebut), program menampilkan pesan error dan meminta input ulang.

3. Menampilkan Menu Pilihan:

- Setelah kode sandi valid dimasukkan, program menampilkan menu dengan 3 opsi:

1. Cek apakah kode sandi adalah bilangan prima.
 2. Cek apakah kode sandi habis dibagi 2 dan 3 tetapi tidak oleh 5.
 3. Keluar dari program.
4. Memproses Pilihan Pengguna:
- Jika memilih opsi 1, program akan memeriksa apakah kode sandi adalah bilangan prima.
 - Jika memilih opsi 2, program akan memeriksa apakah kode sandi habis dibagi 2 dan 3 tetapi tidak oleh 5.
 - Jika memilih opsi 3, program akan berhenti.
 - Jika pengguna memilih opsi yang tidak valid, program menampilkan pesan error dan meminta input ulang.
5. Pengulangan Menu:
- Setelah memilih opsi 1 atau 2, pengguna akan kembali ke menu utama.
 - Pengguna bisa terus melakukan pengecekan hingga memilih opsi 3 untuk keluar.

3. penjelasan block code

1. Import library

```
#include <iostream>
#include <string>
using namespace std;
```

- Menggunakan library <iostream> untuk unput output
- Mengubakan <string> untuk menangani string username dan password

2. Fungsi cek Bilangan prima

```
bool isPrime(int num) {
```

```

    if (num < 2) return false;
    for (int i = 2; i * i <= num; i++) {
        if (num % i == 0) return false;
    }
    return true;
}

```

- Mengecek apakah num adalah bilangan prima dengan metode trial division.

3. Fungsi cek bilangan cek habis dibagi 2 dan 3 tetapi tidak oleh 5

```

bool isDivisible(int num) {
    return (num % 2 == 0 && num % 3 == 0 && num % 5 != 0);
}

```

- Mengecek apakah num habis dibagi 2 dan 3 tetapi tidak oleh 5

4. Process login

```

cout << "Masukkan Username: ";
getline(cin, username);
cout << "Masukkan Password (NIM): ";
getline(cin, password);

if (username != "AdePasihaTangkeAllo" || password != "2409106109") {
    cout << "Login gagal! Username atau password salah." << endl;
    return 0;
}

```

- Meminta input username dan password
- Jika salah, tampilan pesan error dan hentikan program

5. Input kode Sandi

```

6. Cout << "\nMasukkan kode sandi (antara 100 dan 99999): ";
7.     Cin >> code;
8.
9.     if (code < 100 || code > 99999) {
10.        cout << "Kode sandi tidak valid! Harus antara 100 dan 99999."
<< endl;

```

```
11.         continue;
12.     }
```

- Memastikan kode sandi berada dalam rentang 100 hingga 99999

6. Menu pilihan

```
cout << "\nMenu:\n";
    cout << "1. Cek apakah kode sandi adalah bilangan prima" << endl;
    cout << "2. Cek apakah kode sandi habis dibagi 2 dan 3 tetapi
tidak oleh 5" << endl;
    cout << "3. Keluar" << endl;
    cout << "Pilih opsi: ";
    cin >> choice;
```

- menampilkan menu kepada pengguna

7. Switch Case untuk memproses pilihan

```
8. switch (choice) {
9.     case 1:
10.         if (isPrime(code))
11.             cout << "Kode sandi adalah bilangan prima." <<
endl;
12.         else
13.             cout << "Kode sandi bukan bilangan prima." <<
endl;
14.         break;
15.     case 2:
16.         if (isDivisible(code))
17.             cout << "Kode sandi memenuhi syarat habis dibagi
2 dan 3 tetapi tidak oleh 5." << endl;
18.         else
19.             cout << "Kode sandi tidak memenuhi syarat
tersebut." << endl;
20.         break;
21.     case 3:
22.         cout << "Program berhenti." << endl;
23.         return 0;
24.     default:
25.         cout << "Pilihan tidak valid!" << endl;
```

- memproses pilihan pengguna dengan switch case

3. Source Code

```
#include <iostream>
#include <string>
using namespace std;

// Fungsi untuk mengecek apakah bilangan adalah prima
bool isPrime(int num) {
    if (num < 2) return false;
    for (int i = 2; i * i <= num; i++) {
        if (num % i == 0) return false;
    }
    return true;
}

// Fungsi untuk mengecek apakah bilangan memenuhi syarat habis dibagi 2 dan 3
// tetapi tidak oleh 5
bool isDivisible(int num) {
    return (num % 2 == 0 && num % 3 == 0 && num % 5 != 0);
}

int main() {
    string username, password;
    int code;

    // Login
    cout << "Masukkan Username: ";
    getline(cin, username);
    cout << "Masukkan Password (NIM): ";
    getline(cin, password);

    if (username != "AdePasihatangkeAllo" || password != "2409106109") {
        cout << "Login gagal! Username atau password salah." << endl;
        return 0;
    }

    while (true) {
        cout << "\nMasukkan kode sandi (antara 100 dan 99999): ";
        cin >> code;

        if (code < 100 || code > 99999) {
            cout << "Kode sandi tidak valid! Harus antara 100 dan 99999." <<
endl;
            continue;
        }

        int choice;
        do {
```



```

        cout << "\nMenu:\n";
        cout << "1. Cek apakah kode sandi adalah bilangan prima" << endl;
        cout << "2. Cek apakah kode sandi habis dibagi 2 dan 3 tetapi
tidak oleh 5" << endl;
        cout << "3. Keluar" << endl;
        cout << "Pilih opsi: ";
        cin >> choice;

        switch (choice) {
            case 1:
                if (isPrime(code))
                    cout << "Kode sandi adalah bilangan prima." << endl;
                else
                    cout << "Kode sandi bukan bilangan prima." << endl;
                break;
            case 2:
                if (isDivisible(code))
                    cout << "Kode sandi memenuhi syarat habis dibagi 2 dan
3 tetapi tidak oleh 5." << endl;
                else
                    cout << "Kode sandi tidak memenuhi syarat tersebut."
<< endl;
                break;
            case 3:
                cout << "Program berhenti." << endl;
                return 0;
            default:
                cout << "Pilihan tidak valid!" << endl;
        }
    } while (true);
}
return 0;
}

// /Ade ganteng

```

4. Uji Coba dan Hasil Output

4.1 Uji Coba

A. Pengujian Login

Senario 1.1: Login Berhasil

Masukkan Username: AdePasihaTangkeAllo

Masukkan Password (NIM): 2409106109

Masukkan kode sandi (antara 100 dan 99999): 113

Skenario 1.2 Login Gagal Atau tidak valid

Masukkan Username: AdePasihaTangkeAllo

Masukkan Password (NIM): 12345678

Kode sandi tidak valid! Harus antara 100 dan 99999.

Masukan kode sandi (antara 100 dan 99999): 250

Jika input kurang dari 100 atau lebih dari 99999, program meminta input ulang.

B. Pengujian Cek menu

Skenario 2.1 Memeriksa Bilangan Prima

Input pilih 1:

Masukkan kode sandi (antara 100 dan 99999): 113

Menu:

1. Cek apakah kode sandi adalah bilangan prima
2. Cek apakah kode sandi habis dibagi 2 dan 3 tetapi tidak oleh 5
3. Keluar Pilih opsi: 1 Kode sandi adalah bilangan prima.

Pilihan Opsi 113

Kode sandi adalah bilangan Prima.

- User masukkan 113 sebagai kode sandi.
- Memilih opsi 1 untuk mengecek bilangan Prima
- 113 adalah bilangan prima, jadi program mengecek hasilnya.

Skenario 2.2 Cek Bilangan Yang Habis dibagi 2 dan 3 Tapi Tidak Boleh 5

Masukan Username : AdePasihaTangkeAllo

Masukan Password (Nim): 2409106109

Masukkan kode sandi(antara 100 dan 99999): 102

Menu:

- 1, Cek apakah kode sandi adalah bilangan prima
2. Cek apakah kode sandi habis dibagi 2 dan 3 tapi tidak boleh 5
3. Keluar

Pilihan opsi:2

Kode sandi memenuhi syarat habis dibagi 2 dan 3 tapi tidak boleh 5

- User memasukan 102.
- Memilih opsi 2.
- 102 adalah syarat (karena $102 \% 2 == 0$, $102 \% 3 == 0$, dan $102 \% 5 != 0$)

.Skenario 2.3 Keluar Dari Program

Masukan Username: AdePasihaTangkeAllo

Masukan Password: 2409106109

Masukan Kata Sandi (antara 100 dan 99999): 200

Menu:

- 1, Cek apakah kode sandi adalah bilangan prima
2. Cek apakah kode sandi habis dibagi 2 dan 3 tapi tidak boleh 5
3. Keluar
 - User memilih opsi 3 untuk keluar, dan program berhenti.

4.2 Hasil Output

A. Login dan Cek Bilangan Prima

```
Masukkan Username: AdePasihaTangkeAllo
Masukkan Password (NIM): 2409106109

Masukkan kode sandi (antara 100 dan 99999): 113

Menu:
1. Cek apakah kode sandi adalah bilangan prima
2. Cek apakah kode sandi habis dibagi 2 dan 3 tetapi tidak oleh 5
3. Keluar
Pilih opsi: 1
Kode sandi adalah bilangan prima.
```

Gambar 4.1 Login dan cek bilangan prima

B. Cek Apakah Kode Sandi Habis dibagi 2 dan 3 tetapi tidak oleh 5

```
Masukkan Username: AdePasihaTangkeAllo
Masukkan Password (NIM): 2409106109

Masukkan kode sandi (antara 100 dan 99999): 102

Menu:
1. Cek apakah kode sandi adalah bilangan prima
2. Cek apakah kode sandi habis dibagi 2 dan 3 tetapi tidak oleh 5
3. Keluar
Pilih opsi: 2
Kode sandi memenuhi syarat habis dibagi 2 dan 3 tetapi tidak oleh 5.
```

Gambar 4.2 Cek apakah kode dan sandi habis dibagi 2 dan 3 tetapi tidak boleh 5

C. Keluar

```
Masukkan Username: AdePasihaTangkeAllo
Masukkan Password (NIM): 2409106109

Masukkan kode sandi (antara 100 dan 99999): 200

Menu:
1. Cek apakah kode sandi adalah bilangan prima
2. Cek apakah kode sandi habis dibagi 2 dan 3 tetapi tidak oleh 5
3. Keluar
Pilih opsi: 3
```

Gambar 4.1 keluar

5. Git

```
hp@LAPTOP-QCIVAFNM MINGW64 /d/GitHub/Praktikum-APL
$ git config --global user.email "pasihaade@gmail.com"

hp@LAPTOP-QCIVAFNM MINGW64 /d/GitHub/Praktikum-APL
$ git init
Initialized empty Git repository in D:/GitHub/Praktikum-APL/.git/

hp@LAPTOP-QCIVAFNM MINGW64 /d/GitHub/Praktikum-APL (master)
$ git add .

hp@LAPTOP-QCIVAFNM MINGW64 /d/GitHub/Praktikum-APL (master)
$ git branch -M main

hp@LAPTOP-QCIVAFNM MINGW64 /d/GitHub/Praktikum-APL (main)
$ git remote add origin https://github.com/adepasiha/Praktikum-Apl.git

hp@LAPTOP-QCIVAFNM MINGW64 /d/GitHub/Praktikum-APL (main)
$ git commit -m "Ade ganteng"
[main (root-commit) 207074d] Ade ganteng
1 file changed, 74 insertions(+)
create mode 100644 Post-Test/Pots-Test-1/2409106109-AdePasihaTAngkeAllo-PT-1.cpp

hp@LAPTOP-QCIVAFNM MINGW64 /d/GitHub/Praktikum-APL (main)
$ git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 1.04 KiB | 533.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/adepasiha/Praktikum-Apl.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

hp@LAPTOP-QCIVAFNM MINGW64 /d/GitHub/Praktikum-APL (main)
$ |
```

Gambar 5 Tampilan Git Bash

1. Mengatur Email Global Untuk Git

\$ git config --global user "pasihaade@gmail.com"

-Perintah ini digunakan untuk mengatur email global Git, yang akan digunakan dalam setiap commit yang dibuat oleh pengguna.

2. Menginstal Repository Git

\$ Git init

-Peintah ini menginisialisasi (membuat) Repository Git dalam folder Praktikum APL

Jika sudah ada repository Git folder tersebut maka Git hanya akan mengingatkan bahwa repository sudah ada

3. Menambahkan File ke Staging Area

\$ git add.

- Perintah ini menambahkan semua file yang ada didalam folder ke staging area
- Staging area adalah tempat sementara sebuah file dikomit kedalam repository

4. Menambahkan Remote Repository (Gagal Karena Sudah ada)

\$ git Remote add origin <https://Github.com/adepasiha/praltikum-apl>

- perintah ini digunakan untuk menambahkan repository remote dengan nama origin
- Error: “remote origin already exists”, ini terjadi karena sebelumnya sudah ada repository remote yang sudah dikaitkan dengan nama origin.

5. Membuat Commit dengan pesan “Update”

\$ git commit -m “Update”

- Perintah ini menyimpan perubahan dalam repository dengan commit dan pesan “Update”.

-File yang dicommit:

- Post-test/Post-test-1/2409106109-AdePasihaTangkeAllo-PT-1.cpp
- Post-test/Post-test-1/2409106109-AdePasihaTangkeAllo-PT-1.exe

6. Mendorong (Push) Perubahan ke Repository Remote

\$ git push -u origin main

- Perintah ini mengunggah (push) perubahan ke repository remote pada branch main
- Karena ini adalah push pertama, flag -u digunakan untuk mengatur branch local main agar terhubung dengan branch main di remote repository.
- Proses ini terjadi:
 - Menghitung objek (Counting objects: 6).
 - Mengeompresi objek sebelum mengunggahnya.
 - Menulis (mengunggah) objek ke github.
 - Menampilkan informasi bahwa branch main sekarang dilacak oleh remote repository origin/main.