

LAPORAN PRAKTIKUM
POSTTEST 4
ALGORITMA PEMROGRAMAN LANJUT



Disusun oleh:

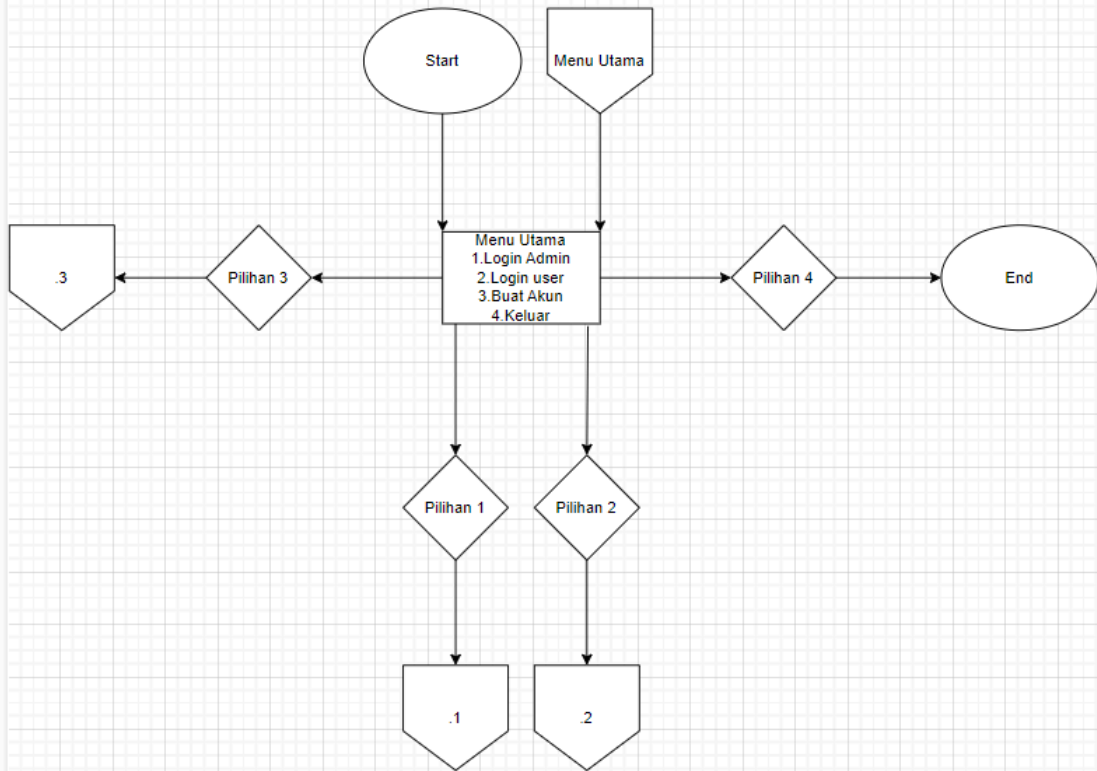
Muhammad Taufiqul Hafizh Muttaqin 2409106099

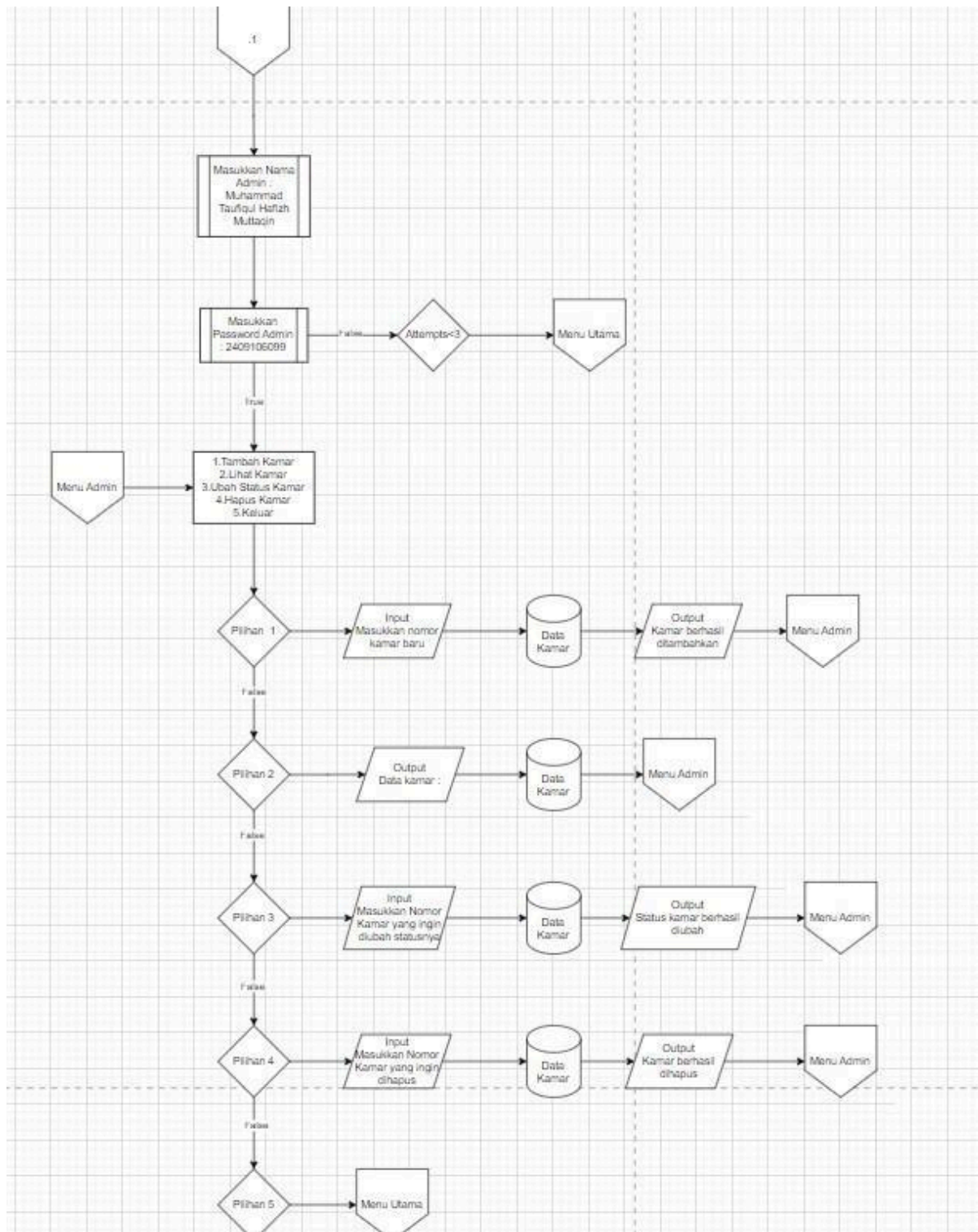
Kelas C1 C'24

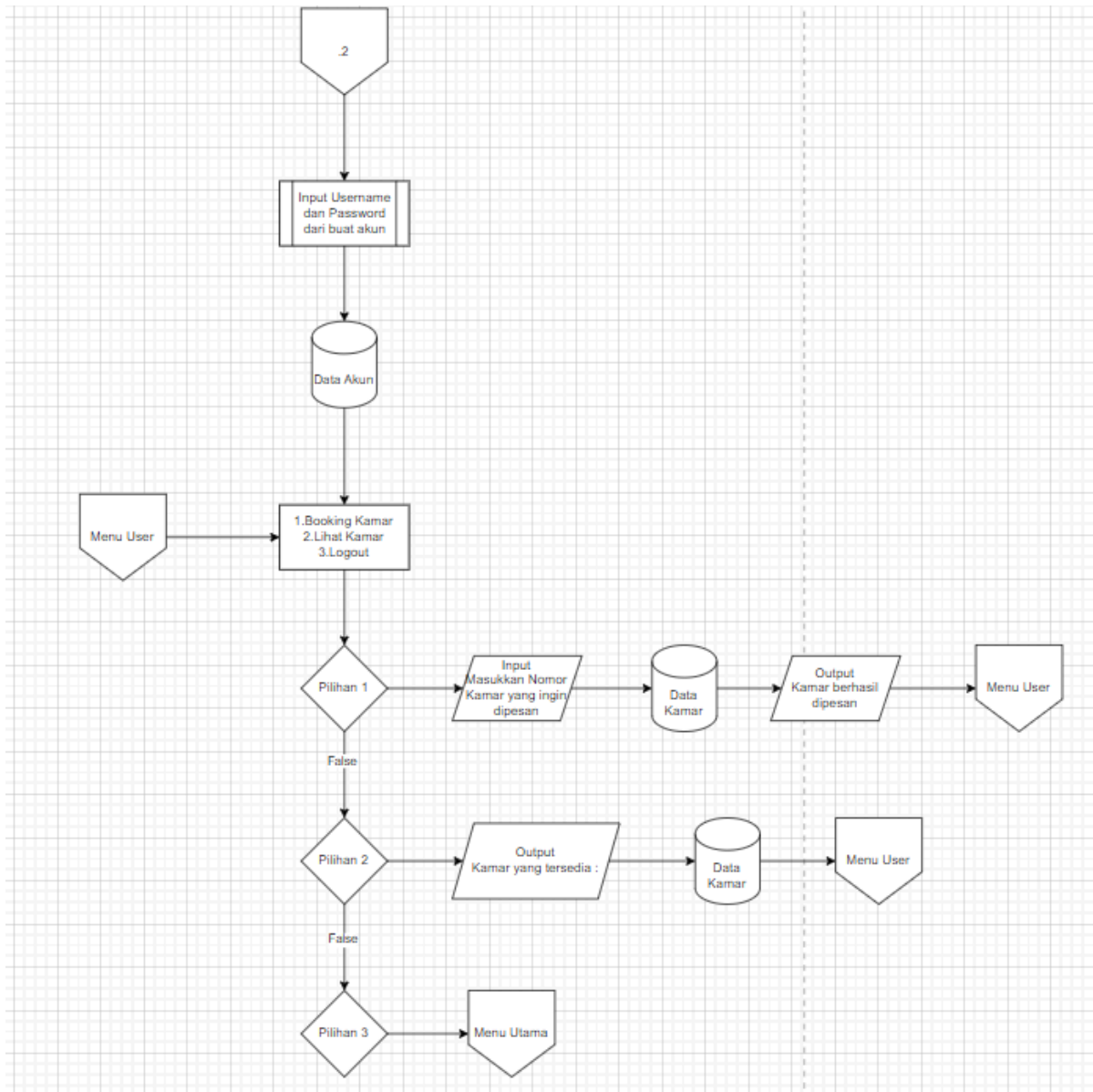
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

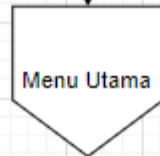
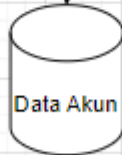
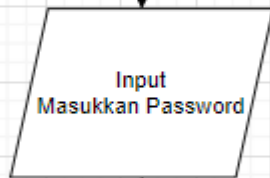
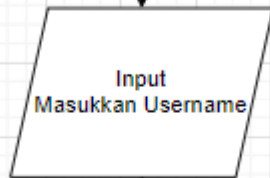
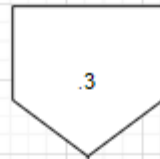
1. Flowchart

Gambar Flowchart:









2. Analisis Program

Deskripsi Singkat

Program

Program ini adalah Sistem Pemesanan Kamar Hotel yang memungkinkan admin untuk mengelola kamar hotel dan pengguna untuk memesan kamar.

Fitur utama:

1. Login Admin: Admin dapat menambahkan, melihat, mengubah status, dan menghapus kamar. dengan username menggunakan nama dan password menggunakan nim.
2. Login User: Pengguna dapat masuk ke akun mereka dan memesan kamar yang tersedia.
3. Registrasi: Pengguna dapat membuat akun baru.
4. Struktur Data: Menggunakan `struct` untuk menyimpan informasi kamar dan pengguna dalam array.

Program ini berbasis menu dengan sistem login yang membatasi akses admin dan pengguna ke fitur yang sesuai.

Note:

Dari program sebelumnya masing-masing fitur yang ada pada menu dipisahkan dari program utama menjadi sub program.

Fungsi wajib menggunakan parameter.

Wajib menggunakan prosedur.

3.Source Code

A. Library

`#include <iostream>` → Untuk input dan output (cin, cout).

`#include <iomanip>` → Untuk format output yang lebih rapi.

`#include <string>` → Untuk menggunakan tipe data string.

`using namespace std;` → Agar tidak perlu menuliskan `std::` sebelum fungsi-fungsi standar.

Source Code:

```
#include <iostream>
#include <iomanip>
#include <string>

using namespace std;
```

B. Fitur Dipisahkan

`tambahKamar` → Subprogram untuk menambah kamar

`lihatKamar` → Subprogram untuk melihat kamar

`ubahStatusKamar` → Subprogram untuk mengubah status kamar

`loginUser loginAdmin buatAKun` → Subprogram Login

`bookingKamar hapusKamar` → Subprogram memesan kamar

`hapusKamar` → Subprogram menghapus kamar

Source Code:

```
int loginUser(User daftarUser[], int userCount, string user, string
pass) {

    for (int i = 0; i < userCount; i++) {

        if (daftarUser[i].username == user && daftarUser[i].password
== pass) {

            return i;

        }

    }

    return -1;

}

void buatAkun(User daftarUser[], int &userCount) {

    cout << "\nMasukkan Username: "; cin >>
daftarUser[userCount].username;

    cout << "Masukkan Password: "; cin >>
daftarUser[userCount].password;
```



```

        userCount++;

        cout << "Akun berhasil dibuat!" << endl;
    }

void tambahKamar(Kamar daftarKamar[], int &kamarCount) {

    cout << "Masukkan nomor kamar baru: "; cin >>
daftarKamar[kamarCount].nomor;

    daftarKamar[kamarCount].status = 1;

    kamarCount++;

    cout << "Kamar berhasil ditambahkan!" << endl;
}

void lihatKamar(Kamar daftarKamar[], int kamarCount) {

    cout << "\nDaftar Kamar:\n";

    cout << "+-----+-----+\n";

    cout << "| Nomor | Status | \n";

    cout << "+-----+-----+\n";

    for (int i = 0; i < kamarCount; i++) {

        cout << "| " << setw(5) << daftarKamar[i].nomor << " | ";

        cout << setw(10) << (daftarKamar[i].status ? "Tersedia" :
"Terpesan") << " | \n";

    }

    cout << "+-----+-----+\n";
}

void ubahStatusKamar(Kamar daftarKamar[], int kamarCount, int nomor)

```

```

{

    for (int i = 0; i < kamarCount; i++) {

        if (daftarKamar[i].nomor == nomor) {

            daftarKamar[i].status = 1 - daftarKamar[i].status;

            cout << "Status kamar berhasil diubah!" << endl;

            return;

        }

    }

    cout << "Kamar tidak ditemukan!" << endl;

}

void hapusKamar(Kamar daftarKamar[], int &kamarCount, int nomor) {

    for (int i = 0; i < kamarCount; i++) {

        if (daftarKamar[i].nomor == nomor) {

            for (int j = i; j < kamarCount - 1; j++) {

                daftarKamar[j] = daftarKamar[j + 1];

            }

            kamarCount--;

            cout << "Kamar berhasil dihapus!" << endl;

            return;

        }

    }

    cout << "Kamar tidak ditemukan!" << endl;

}

```

```

void bookingKamar(Kamar daftarKamar[], int kamarCount, int nomor) {
    for (int i = 0; i < kamarCount; i++) {
        if (daftarKamar[i].nomor == nomor && daftarKamar[i].status ==
1) {
            daftarKamar[i].status = 0;

            cout << "Kamar berhasil dipesan!" << endl;

            return;
        }
    }

    cout << "Kamar tidak tersedia atau tidak ditemukan!" << endl;
}

void lihatKamarTersedia(Kamar daftarKamar[], int kamarCount) {
    cout << "\nKamar yang tersedia:\n";

    for (int i = 0; i < kamarCount; i++) {
        if (daftarKamar[i].status == 1) {

            cout << "+-----+-----+\n";

            cout << "Kamar " << daftarKamar[i].nomor << "
tersedia.\n";

            cout << "+-----+-----+\n";

        }
    }
}

```

→ Subprogram untuk melihat kamar

```

int main() {
    int pilihan;
    string user, pass;
    int loginIndex = -1;
    int isAdmin = 0;
    int nomor;

    while (true) {
        cout << "\\n=== Sistem Pemesanan Kamar Hotel ===\\n";
        cout << "1. Login Admin\\n2. Login User\\n3. Buat Akun\\n4.
Keluar\\nPilihan: ";
        cin >> pilihan;

        switch (pilihan) {
            case 1: {
                int attempt = 0;
                cin.ignore();
                while (attempt < 3) {
                    cout << "\\nMasukkan Nama Admin: "; getline(cin, user);
                    cout << "Masukkan Password Admin: "; cin >> pass;
                    cin.ignore();
                    if (loginAdmin(user, pass)) {
                        isAdmin = 1;
                        break;
                    } else {
                        cout << "Login gagal! Percobaan ke-" << (attempt + 1)
<< " dari 3.\\n";
                        attempt++;
                    }
                }
                if (attempt == 3) cout << "Gagal login sebagai admin, kembali
ke menu utama.\\n";
                break;
            }

            case 2:
                cout << "\\nMasukkan Username: "; cin >> user;
                cout << "Masukkan Password: "; cin >> pass;
                loginIndex = loginUser(daftarUser, userCount, user, pass);
                if (loginIndex == -1) cout << "Login gagal!" << endl;
                else cout << "Login berhasil!" << endl;

```

```

        break;

    case 3:
        buatAkun(daftarUser, userCount);
        break;

    case 4:
        return 0;
}

while (isAdmin) {
    cout << "\n=== Menu Admin ===\n";
    cout << "1. Tambah Kamar\n2. Lihat Kamar\n3. Ubah Status
Kamar\n4. Hapus Kamar\n5. Keluar\nPilihan: ";
    cin >> pilihan;

    switch (pilihan) {
        case 1: tambahKamar(daftarKamar, kamarCount); break;
        case 2: lihatKamar(daftarKamar, kamarCount); break;
        case 3:
            cout << "Masukkan Nomor Kamar: "; cin >> nomor;
            ubahStatusKamar(daftarKamar, kamarCount, nomor); break;
        case 4:
            cout << "Masukkan Nomor Kamar: "; cin >> nomor;
            hapusKamar(daftarKamar, kamarCount, nomor); break;
        case 5: isAdmin = 0; cout << "Logout admin berhasil.\n";
break;
    }
}

while (loginIndex != -1) {
    cout << "\n=== Menu User ===\n";
    cout << "1. Booking Kamar\n2. Lihat Kamar\n3. Logout\nPilihan: ";
    cin >> pilihan;

    switch (pilihan) {
        case 1:
            cout << "Masukkan Nomor Kamar yang ingin dipesan: "; cin
>> nomor;

            bookingKamar(daftarKamar, kamarCount, nomor); break;
        case 2: lihatKamarTersedia(daftarKamar, kamarCount); break;
        case 3: loginIndex = -1; cout << "Logout user berhasil.\n";
break;
    }
}

```



C. Fungsi Menggunakan Parameter

Fungsi loginAdmin menerima dua parameter yaitu user dan pass

Fungsi loginUser menerima array user, jumlah user, serta username dan password sebagai parameter.

Source Code:

```
bool loginAdmin(string user, string pass)
```

```
int loginUser(User daftarUser[], int userCount, string user, string pass)
```

D. Fungsi menggunakan prosedur

Nama Prosedur	Fungsi
<code>tambahKamar(...)</code>	Menambahkan data kamar baru
<code>hapusKamar(...)</code>	Menghapus Kamar
<code>buatAkun(...)</code>	Membuat akun user baru
<code>lihatKamar(...)</code>	Menampilkan seluruh kamar
<code>ubahStatusKamar(...)</code>	Mengubah status kamar (tersedia ↔ terpesan)
<code>bookingKamar(...)</code>	User memesan kamar
<code>lihatKamarTersedia(...)</code>	Menampilkan hanya kamar yang tersedia

Source Code:

```
// ===== FUNGSI & PROSEDUR =====

// Fungsi login admin
bool loginAdmin(string user, string pass) {
    return (user == "Muhammad Taufiquel Hafizh Muttaqin" && pass == "2409106099");
}

// Fungsi login user
int loginUser(User daftarUser[], int userCount, string user, string pass) {
    for (int i = 0; i < userCount; i++) {
        if (daftarUser[i].username == user && daftarUser[i].password == pass) {
            return i;
        }
    }
    return -1;
}

// Prosedur membuat akun
void buatAkun(User daftarUser[], int &userCount) {
    cout << "\nMasukkan Username: ";
    cin >> daftarUser[userCount].username;
    cout << "Masukkan Password: ";
    cin >> daftarUser[userCount].password;
    userCount++;
    cout << "Akun berhasil dibuat!" << endl;
}
```



```

// Prosedur menambah kamar
void tambahKamar(Kamar daftarKamar[], int &kamarCount) {
    cout << "Masukkan nomor kamar baru: ";
    cin >> daftarKamar[kamarCount].nomor;
    daftarKamar[kamarCount].status = 1;
    kamarCount++;
    cout << "Kamar berhasil ditambahkan!" << endl;
}

// Prosedur melihat daftar kamar
void lihatKamar(Kamar daftarKamar[], int kamarCount) {
    cout << "\nDaftar Kamar:\n";
    cout << "+-----+-----+\n";
    cout << "| Nomor | Status | \n";
    cout << "+-----+-----+\n";
    for (int i = 0; i < kamarCount; i++) {
        cout << "| " << setw(5) << daftarKamar[i].nomor << " | ";
        cout << setw(10) << (daftarKamar[i].status ? "Tersedia" : "Terpesan") << "
\n";
    }
    cout << "+-----+-----+\n";
}

// Prosedur mengubah status kamar
void ubahStatusKamar(Kamar daftarKamar[], int kamarCount, int nomor) {
    for (int i = 0; i < kamarCount; i++) {
        if (daftarKamar[i].nomor == nomor) {
            daftarKamar[i].status = 1 - daftarKamar[i].status;
            cout << "Status kamar berhasil diubah!" << endl;
            return;
        }
    }
    cout << "Kamar tidak ditemukan!" << endl;
}

// ✓ Prosedur menghapus kamar (contoh utama)
void hapusKamar(Kamar daftarKamar[], int &kamarCount, int nomor) {
    for (int i = 0; i < kamarCount; i++) {
        if (daftarKamar[i].nomor == nomor) {
            for (int j = i; j < kamarCount - 1; j++) {
                daftarKamar[j] = daftarKamar[j + 1];
            }
            kamarCount--;
            cout << "Kamar berhasil dihapus!" << endl;
            return;
        }
    }
    cout << "Kamar tidak ditemukan!" << endl;
}

// Prosedur untuk booking kamar oleh user
void bookingKamar(Kamar daftarKamar[], int kamarCount, int nomor) {
    for (int i = 0; i < kamarCount; i++) {
        if (daftarKamar[i].nomor == nomor && daftarKamar[i].status == 1) {
            daftarKamar[i].status = 0;
        }
    }
}

```

```

        cout << "Kamar berhasil dipesan!" << endl;
        return;
    }
}
cout << "Kamar tidak tersedia atau tidak ditemukan!" << endl;
}

// Prosedur menampilkan kamar yang tersedia
void lihatKamarTersedia(Kamar daftarKamar[], int kamarCount) {
    cout << "\nKamar yang tersedia:\n";
    for (int i = 0; i < kamarCount; i++) {
        if (daftarKamar[i].status == 1) {
            cout << "+-----+-----+\n";
            cout << "Kamar " << daftarKamar[i].nomor << " tersedia.\n";
            cout << "+-----+-----+\n";
        }
    }
}

// ===== PROGRAM UTAMA =====
int main() {
    int pilihan;
    string user, pass;
    int loginIndex = -1;
    int isAdmin = 0;
    int nomor;

    while (true) {
        cout << "\n=== Sistem Pemesanan Kamar Hotel ===\n";
        cout << "1. Login Admin\n2. Login User\n3. Buat Akun\n4. Keluar\nPilihan: ";

        cin >> pilihan;

        switch (pilihan) {
            case 1: {
                int attempt = 0;
                cin.ignore();
                while (attempt < 3) {
                    cout << "\nMasukkan Nama Admin: "; getline(cin, user);
                    cout << "Masukkan Password Admin: "; cin >> pass;
                    cin.ignore();
                    if (loginAdmin(user, pass)) {
                        isAdmin = 1;
                        break;
                    } else {
                        cout << "Login gagal! Percobaan ke-" << (attempt + 1) << "
dari 3.\n";
                        attempt++;
                    }
                }
                if (attempt == 3) cout << "Gagal login sebagai admin, kembali ke
menu utama.\n";
                break;
            }

```

```

        case 2:
            cout << "\nMasukkan Username: ";
            cin >> user;
            cout << "Masukkan Password: ";
            cin >> pass;
            loginIndex = loginUser(daftarUser, userCount, user, pass);
            if (loginIndex == -1)
                cout << "Login gagal!" << endl;
            else
                cout << "Login berhasil!" << endl;
            break;

        case 3:
            buatAkun(daftarUser, userCount);
            break;

        case 4:
            return 0;
    }

    // Menu admin
    while (isAdmin) {
        cout << "\n=== Menu Admin ===\n";
        cout << "1. Tambah Kamar\n2. Lihat Kamar\n3. Ubah Status Kamar\n4. Hapus Kamar\n5. Keluar\nPilihan: ";
        cin >> pilihan;

        switch (pilihan) {
            case 1: tambahKamar(daftarKamar, kamarCount); break;
            case 2: lihatKamar(daftarKamar, kamarCount); break;
            case 3:
                cout << "Masukkan Nomor Kamar: ";
                cin >> nomor;
                ubahStatusKamar(daftarKamar, kamarCount, nomor);
                break;
            case 4:
                cout << "Masukkan Nomor Kamar: ";
                cin >> nomor;
                hapusKamar(daftarKamar, kamarCount, nomor); //  pemanggilan
                prosedur hapus kamar
                break;
            case 5:
                isAdmin = 0;
                cout << "Logout admin berhasil.\n";
                break;
        }
    }

    // Menu user
    while (loginIndex != -1) {
        cout << "\n=== Menu User ===\n";
        cout << "1. Booking Kamar\n2. Lihat Kamar\n3. Logout\nPilihan: ";
        cin >> pilihan;

        switch (pilihan) {

```

```
        case 1:
            cout << "Masukkan Nomor Kamar yang ingin dipesan: ";
            cin >> nomor;
            bookingKamar(daftarKamar, kamarCount, nomor);
            break;
        case 2:
            lihatKamarTersedia(daftarKamar, kamarCount);
            break;
        case 3:
            loginIndex = -1;
            cout << "Logout user berhasil.\n";
            break;
    }
}
}
```

4. Uji Coba dan Hasil Output

4.1 Uji Coba Login Admin

Uji Coba Login Gagal Menggunakan Username Nama dan Password NIM serta Percobaan jika salah 3x salah maka kembali ke menu utama.

Gambar Output :

```
=== Sistem Pemesanan Kamar Hotel ===
1. Login Admin
2. Login User
3. Buat Akun
4. Keluar
Pilihan: 1

Masukkan Nama Admin: a
Masukkan Password Admin: a
Login gagal! Percobaan ke-1 dari 3.

Masukkan Nama Admin: a
Masukkan Password Admin: a
Login gagal! Percobaan ke-2 dari 3.

Masukkan Nama Admin: a
Masukkan Password Admin: a
Login gagal! Percobaan ke-3 dari 3.
Gagal login sebagai admin, kembali ke menu utama.

=== Sistem Pemesanan Kamar Hotel ===
1. Login Admin
2. Login User
3. Buat Akun
4. Keluar
Pilihan: █
```

Uji Coba Login Berhasil Menggunakan Username Nama dan Password NIM

Gambar Output :

```
=== Sistem Pemesanan Kamar Hotel ===
```

1. Login Admin
2. Login User
3. Buat Akun
4. Keluar

Pilihan: 1

Masukkan Nama Admin: Muhammad Taufiqul Hafizh Muttaqin

Masukkan Password Admin: 2409106099

```
=== Menu Admin ===
```

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Kamar(Tersedia/Terpesan)
4. Hapus Kamar
5. Keluar

Pilihan: █

4.2 Uji Coba Buat Akun dan Login User

Gambar Output:

```
=== Sistem Pemesanan Kamar Hotel ===
1. Login Admin
2. Login User
3. Buat Akun
4. Keluar
Pilihan: 3

Masukkan Username: apis
Masukkan Password: apis
Akun berhasil dibuat!

=== Sistem Pemesanan Kamar Hotel ===
1. Login Admin
2. Login User
3. Buat Akun
4. Keluar
Pilihan: 2

Masukkan Username: apis
Masukkan Password: api
Login gagal!

=== Sistem Pemesanan Kamar Hotel ===
1. Login Admin
2. Login User
3. Buat Akun
4. Keluar
Pilihan: 2

Masukkan Username: apis
Masukkan Password: apis
Login berhasil!

=== Menu User ===
1. Booking Kamar
2. Lihat Kamar
3. Logout
Pilihan: 1
```

4.3 Uji Coba Keluar Program

Gambar Output:

```
=== Sistem Pemesanan Kamar Hotel ===  
1. Login Admin  
2. Login User  
3. Buat Akun  
4. Keluar  
Pilihan: 4  
PS C:\Users\asus\Downloads\PostTest2>
```


4.4 Uji Coba CRUD Admin

Gambar Output:

```
=== Menu Admin ===
```

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Status Kamar
4. Hapus Kamar
5. Keluar

Pilihan: 1

Masukkan nomor kamar baru: 1

Kamar berhasil ditambahkan!

```
=== Menu Admin ===
```

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Status Kamar
4. Hapus Kamar
5. Keluar

Pilihan: 2

Daftar Kamar:

+	-----	+	-----	+
	Nomor		Status	
+	-----	+	-----	+
	1		Tersedia	
+	-----	+	-----	+

```
=== Menu Admin ===
```

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Status Kamar
4. Hapus Kamar
5. Keluar

Pilihan: 3

Masukkan Nomor Kamar yang ingin diubah statusnya: 1

Status kamar berhasil diubah!

```
=== Menu Admin ===
```

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Status Kamar
4. Hapus Kamar

=== Menu Admin ===

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Status Kamar
4. Hapus Kamar
5. Keluar

Pilihan: 4

Masukkan Nomor Kamar yang ingin dihapus: 1
Kamar berhasil dihapus!

=== Menu Admin ===

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Status Kamar
4. Hapus Kamar
5. Keluar

Pilihan: 2

Daftar Kamar:

Nomor	Status

=== Menu Admin ===

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Status Kamar
4. Hapus Kamar
5. Keluar

Pilihan: █

4.5 Uji Coba Lihat dan Booking Kamar User

Gambar Output:

```
=== Menu User ===
```

1. Booking Kamar
2. Lihat Kamar
3. Logout

Pilihan: 2

Kamar yang tersedia:

```
+-----+-----+
```

Kamar 1 tersedia.

```
+-----+-----+
```

```
+-----+-----+
```

Kamar 2 tersedia.

```
+-----+-----+
```

```
+-----+-----+
```

Kamar 3 tersedia.

```
+-----+-----+
```

```
=== Menu User ===
```

1. Booking Kamar
2. Lihat Kamar
3. Logout

Pilihan: 1

Masukkan Nomor Kamar yang ingin dipesan: 3

Kamar berhasil dipesan!

```
=== Menu User ===
```

1. Booking Kamar
2. Lihat Kamar
3. Logout

Pilihan: 2

Kamar yang tersedia:

```
+-----+-----+
```

Kamar 1 tersedia.

```
+-----+-----+
```

```
+-----+-----+
```

Kamar 2 tersedia.

```
+-----+-----+
```

Masukkan Nama Admin: Muhammad Taufiqul Hafizh Muttaqin
Masukkan Password Admin: 2409106099

=== Menu Admin ===

1. Tambah Kamar
2. Lihat Kamar
3. Ubah Status Kamar
4. Hapus Kamar
5. Keluar

Pilihan: 2

Daftar Kamar:

+-----+-----+-----+	
Nomor	Status
+-----+-----+-----+	
1	Tersedia
2	Tersedia
3	Terpesan
+-----+-----+-----+	

Tampilan Lihat Kamar di Menu Admin Jika Ada Kamar Yang Terpesan

GIT

