

**LAPORAN TUGAS PRAKTIKUM  
PEMROGRAMAN ALGORITMA DAN PEMROGRAMAN**

**TUGAS PEKAN 7**

**Disusun Oleh:  
Muhammad Aufa Rafiki  
2511531012**

**Dosen Pengampu: Dr. Wahyudi S.T.M.T  
Asisten Praktikum: Muhammad Zaki Al Hafiz**



**DEPARTEMEN INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS**

**2025**

## 1. Akun.java

PSEUDOCODE

Judul
Program Membuat Class Data dengan nama Class Akun (Class ini bertugas sebagai blueprint untuk menyimpan data akun.)
Deklarasi
String username; String password; String email; Integer pinAngka;
Algoritma
1. METHOD setUsername(String user) 2.   this.username = user 3. AKHIR METHOD 4. METHOD setPassword(String pass) 5.   this.password = pass 6. AKHIR METHOD 7. METHOD setEmail(String mail) 8.   this.email = mail 9. AKHIR METHOD 10. METHOD setPinAngka(int pin) 11.   this.pinAngka = pin 12. AKHIR METHOD 13. METHOD getUsername() 14.   RETURN username 15. AKHIR METHOD 16. METHOD getUsername() 17.   RETURN username 18. AKHIR METHOD 19. METHOD getEmail() 20.   RETURN email 21. AKHIR METHOD 22. METHOD getPinAngka() 23.   RETURN pinAngka 24. AKHIR METHOD 25. METHOD isPasswordValid() 26.   RETURN (panjang dari password >= 8) 27. AKHIR METHOD 28. METHOD isEmailValid() 29.   RETURN (email mengandung "@" DAN email mengandung ".") 30. AKHIR METHOD 31. AKHIR CLASS

## Kode Program

```
1 package pekan7_2511531012;
2
3 public class Akun {
4     private String username;
5     private String password;
6     private String email;
7     private int pinAngka;
8     public void setUsername(String username) {
9         this.username = username;
10    }
11    public void setPassword(String password) {
12        this.password = password;
13    }
14    public void setEmail(String email) {
15        this.email = email;
16    }
17    public void setPinAngka(int pinAngka) {
18        this.pinAngka = pinAngka;
19    }
20    public String getUsername() {
21        return username;
22    }
23    public String getPassword() {
24        return password;
25    }
26    public String getEmail() {
27        return email;
28    }
29    public int getPinAngka() {
30        return pinAngka;
31    }
32    public boolean isPasswordValid() {
33        return this.password.length() >= 8;
34    }
35    public boolean isEmailValid() {
36        return this.email.contains "@" && this.email.contains ".";
37    }
38
39 }
```

## 2. Kelas Utama

PSEUDOCODE

Judul

Program Membuat Class Utama

Deklarasi

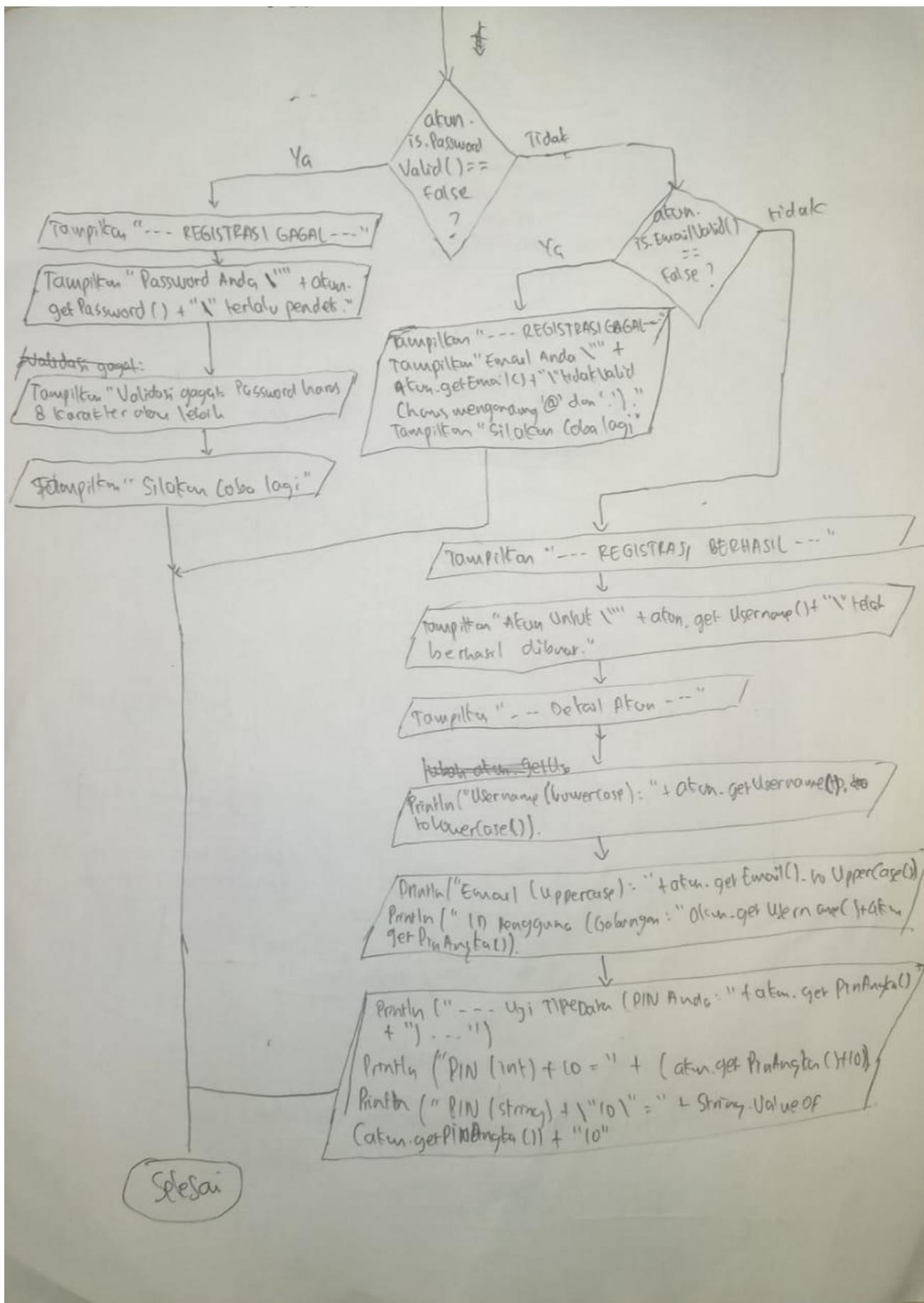
String username,  
String password,  
String email,  
String pinInput,  
Integer pin,  
String pinString,

Algoritma

1. input ← BUAT Scanner BARU
2. akun ← BUAT Akun BARU
3. CETAK "===== REGISTRASI AKUN BARU ====="
4. CETAK "Masukkan Username: "
5. BACA username
6. PANGGIL akun.setUsername(username)
7. CETAK "Masukkan Password: "
8. BACA password
9. PANGGIL akun.setPassword(password)
10. CETAK "Masukkan Email: "
11. BACA email
12. PANGGIL akun.setEmail(email)
13. CETAK "Masukkan PIN (6 digit): "
14. BACA pinInput
15. pin ← KONVERSI pinInput KE INTEGER
16. PANGGIL akun.setPinAngka(pin)
17. CETAK BARIS
18. JIKA NOT akun.isPasswordValid() MAKA
  - CETAK "REGISTRASI GAGAL"
  - CETAK "Password Anda " + akun.getPassword() + " terlalu pendek."
  - CETAK "Validasi gagal: Password harus 8 karakter atau lebih."
  - CETAK "Silakan coba lagi."
19. LAIN JIKA NOT akun.isEmailValid() MAKA
  - CETAK "REGISTRASI GAGAL"
  - CETAK "Email Anda " + akun.getEmail() + " tidak valid (harus mengandung '@' dan '.')."
  - CETAK "Silakan coba lagi."
20. LAIN (Semua Valid) MAKA
  - CETAK "REGISTRASI BERHASIL"
  - CETAK "Akun untuk " + akun.getUsername() + " telah berhasil dibuat."
  - CETAK "--- Detail Akun ---"
  - CETAK "Username (Lowercase): " + toLowerCase(akun.getUsername())
  - CETAK "Email (Uppercase) : " + toUpperCase(akun.getEmail())
  - CETAK "ID Pengguna (Gabungan): " + concat(akun.getUsername(),  
toString(akun.getPinAngka()))
  - CETAK "--- Uji Tipe Data (PIN Anda: " + akun.getPinAngka() + ") ---"
  - CETAK "PIN (int) + 10 = " + (akun.getPinAngka() + 10)
  - pinString ← toString(akun.getPinAngka())
  - CETAK "PIN(String) + ""10"" = " + concat(pinString, "10")
21. AKHIR JIKA
22. TUTUP input
23. AKHIR ALGORITMA

## FLOWCHART





## KODE PROGRAM

```
1 package pekan7_2511531012;
2 import java.util.Scanner;
3 public class tugasAlproPekan7_2511531012 {
4
5     public static void main(String[] args) {
6         Scanner input = new Scanner(System.in);
7         Akun akun = new Akun();
8
9         System.out.println("===== REGISTRASI AKUN BARU =====");
10        System.out.print("Masukkan Username: ");
11        String username = input.nextLine();
12        akun.setUsername(username);
13
14        System.out.print("Masukkan Password: ");
15        String password = input.nextLine();
16        akun.setPassword(password);
17
18        System.out.print("Masukkan Email: ");
19        String email = input.nextLine();
20        akun.setEmail(email);
21
22        System.out.print("Masukkan PIN (6 digit): ");
23        int pin = Integer.parseInt(input.nextLine());
24        akun.setPinAngka(pin);
25
26        System.out.println();
27
28        if (!akun.isPasswordValid()) {
29            System.out.println("--- REGISTRASI GAGAL ---");
30            System.out.println("Password Anda '" + akun.getPassword() + "' terlalu pendek.");
31            System.out.println("Validasi gagal: Password harus 8 karakter atau lebih.");
32            System.out.println("Silakan coba lagi.");
33
34    } else if (!akun.isEmailValid()) {
35        System.out.println("--- REGISTRASI GAGAL ---");
36        System.out.println("Email Anda '" + akun.getEmail() + "' tidak valid (harus mengandung '@' dan '.').");
37        System.out.println("Silakan coba lagi.");
38
39    } else {
40        System.out.println("--- REGISTRASI BERHASIL ---");
41        System.out.println("Akun untuk '" + akun.getUsername() + "' telah berhasil dibuat.");
42        System.out.println();
43        System.out.println("--- Detail Akun ---");
44        System.out.println("Username (Lowercase): " + akun.getUsername().toLowerCase());
45        System.out.println("Email (Uppercase) : " + akun.getEmail().toUpperCase());
46        System.out.println("ID Pengguna (Gabungan): " + akun.getUsername() + akun.getPinAngka());
47        System.out.println();
48        System.out.println("--- Uji Tipe Data (PIN Anda: " + akun.getPinAngka() + ") ---");
49        System.out.println("PIN (int) + 10 = " + (akun.getPinAngka() + 10));
50        System.out.println("PIN (String) + \"10\" = " + String.valueOf(akun.getPinAngka() + 10));
51    }
52    input.close();
53
54 }
55
56 }
```

## OUTPUT

```
===== REGISTRASI AKUN BARU =====
Masukkan Username: aufa
Masukkan Password: 1234
Masukkan Email: aufa@ku.ko
Masukkan PIN (6 digit): 123456

--- REGISTRASI GAGAL ---
Password Anda "1234" terlalu pendek.
Validasi gagal: Password harus 8 karakter atau lebih.
Silakan coba lagi.
```

```
===== REGISTRASI AKUN BARU =====
Masukkan Username: aufa
Masukkan Password: 12345678
Masukkan Email: aufa@ku
Masukkan PIN (6 digit): 121212

--- REGISTRASI GAGAL ---
Email Anda "aufa@ku" tidak valid (harus mengandung '@' dan '.').
Silakan coba lagi.
```

```
===== REGISTRASI AKUN BARU =====
Masukkan Username: aufa
Masukkan Password: 12345678
Masukkan Email: aufa@ku.ko
Masukkan PIN (6 digit): 121212
|
--- REGISTRASI BERHASIL ---
Akun untuk "aufa" telah berhasil dibuat.

--- Detail Akun ---
Username (Lowercase): aufa
Email (Uppercase) : AUFA@KU.KO
ID Pengguna (Gabungan): aufa121212

--- Uji Tipe Data (PIN Anda: 121212) ---
PIN (int) + 10 = 121222
PIN (String) + "10" = 12121210
```

#### PENJELASAN SINGKAT KEDUA PROGRAM

Program registrasi akun ini beroperasi berdasarkan prinsip Pemrograman Berorientasi Objek (OOP) dengan membagi fungsionalitas ke dalam dua kelas utama: Akun.java sebagai kelas data (blueprint) dan tugasAlproPekan7\_2511531012.java sebagai kelas pengendali alur utama. Kelas Akun berfungsi sebagai wadah untuk menyimpan informasi akun seperti username, password, email, dan PIN. Data-data ini dilindungi sebagai atribut privat, yang berarti hanya bisa diakses atau diubah melalui metode publik yang disediakan oleh kelas itu sendiri, yaitu setter (untuk mengisi data) dan getter (untuk mengambil data). Selain penyimpanan data, Akun juga menyediakan metode helper isPasswordValid() dan isEmailValid(). Metode isPasswordValid() bertanggung jawab memeriksa apakah panjang password yang diberikan memenuhi standar minimal (8 karakter), sementara isEmailValid() memvalidasi format email dengan memastikan adanya karakter '@' dan '.' dalam alamat email. Kedua metode ini mengembalikan nilai boolean (true/false) yang menunjukkan hasil validasi.

Kelas tugasAlproPekan7\_2511531012, di sisi lain, adalah titik awal eksekusi program. Kelas ini bertanggung jawab untuk berinteraksi langsung dengan pengguna. Pada awalnya, ia menginisialisasi Scanner untuk menerima masukan dari konsol dan membuat objek Akun baru untuk setiap registrasi. Selanjutnya, program meminta pengguna memasukkan detail akun (username, password, email, dan

PIN), dan setiap masukan tersebut kemudian disimpan ke dalam objek Akun melalui panggilan metode setter yang relevan. Setelah semua data terkumpul, Registrasi memulai proses validasi menggunakan struktur percabangan if-else if-else. Pertama, ia memanggil akun.isPasswordValid() untuk mengecek apakah password sudah memenuhi kriteria. Jika password tidak valid, program akan langsung menampilkan pesan "REGISTRASI GAGAL" dan mengakhiri eksekusi. Namun, jika password valid, program akan melanjutkan untuk memeriksa validitas email dengan memanggil akun.isEmailValid(). Jika email tidak valid, program juga akan menampilkan pesan kegagalan dan berakhir. Hanya jika kedua validasi (password dan email) berhasil, program akan memasuki blok else, di mana ia mencetak pesan "REGISTRASI BERHASIL" beserta detail akun lengkap. Dalam fase ini, Registrasi juga mendemonstrasikan beberapa manipulasi string (seperti mengubah username menjadi huruf kecil atau email menjadi huruf besar, serta menggabungkan username dan PIN), dan menunjukkan perbedaan antara operasi aritmatika integer dan konkatenasi string saat berinteraksi dengan nilai PIN. Akhirnya, program menutup Scanner dan mengakhiri prosesnya. Secara keseluruhan, program ini menggambarkan bagaimana dua kelas dapat berkolaborasi, dengan satu kelas mengelola data dan logikanya, sementara kelas lain mengendalikan alur interaksi dan presentasi hasil.