

Niky Tania Sari (18/SIB 1B)

TUGAS PRAKTIKUM ALSD JOBSHEET 9

PERCOBAAN 1

```
package Jobsheet9;

public class Mahasiswa18 {
    String nama, nim, kelas;
    int nilai;

    public Mahasiswa18(String nama, String nim, String kelas) {
        this.nama = nama;
        this.nim = nim;
        this.kelas = kelas;
        nilai = -1;
    }

    void tugasDinilai(int nilai) {
        this.nilai = nilai;
    }
}
```

```

StackTugasMahasiswa18.java

package Jobsheet9;

public class StackTugasMahasiswa18 {
    Mahasiswa18[] stack;
    int top;
    int size;

    public StackTugasMahasiswa18(int size) {
        this.size = size;
        stack = new Mahasiswa18[size];
        top = -1;
    }

    public boolean isFull() {
        if (top == size - 1) {
            return true;
        } else {
            return false;
        }
    }

    public boolean isEmpty() {
        if (top == -1) {
            return true;
        } else {
            return false;
        }
    }

    public void push(Mahasiswa18 mhs) {
        if (!isFull()) {
            top++;
            stack[top] = mhs;
        } else {
            System.out.println("Stack penuh! Tidak bisa menambahkan tugas lagi.");
        }
    }

    public Mahasiswa18 pop() {
        if (!isEmpty()) {
            Mahasiswa18 m = stack[top];
            top--;
            return m;
        } else {
            System.out.println("Stack kosong! Tidak ada tugas untuk dinilai");
            return null;
        }
    }

    public Mahasiswa18 peek() {
        if (!isEmpty()) {
            return stack[top];
        } else {
            System.out.println("Stack kosong! Tidak ada tugas yang dikumpulkan");
            return null;
        }
    }

    public void print() {
        for (int i = top; i >= 0; i--) { //yang pertama mengumpulkan ada di bawah
            System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);
        }
        System.out.println("");
    }

    public Mahasiswa18 bottom() {
        if (!isEmpty()) {
            return stack[0];
        } else {
            System.out.println("Stack kosong! Tidak ada tugas terbawah");
            return null;
        }
    }

    public int jumlahTugas() {
        return top + 1;
    }

    public String konversiDesimalKeBiner(int nilai) {
        StackKonversi18 stack = new StackKonversi18();
        while (nilai != 0) {
            int sisa = nilai % 2;
            stack.push(sisa);
            nilai = nilai / 2;
        }
        String biner = new String();
        while (!stack.isEmpty()) {
            biner += stack.pop();
        }
        return biner;
    }
}

```

```

MahasiswaDemo18.java

package Jobsheet9;

import java.util.Scanner;

public class MahasiswaDemo18 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int pilih;

        StackTugasMahasiswa18 stack = new StackTugasMahasiswa18(5);

        do {
            System.out.println("\nMenu:");
            System.out.println("1. Mengumpulkan Tugas");
            System.out.println("2. Menilai Tugas");
            System.out.println("3. Melihat Tugas Teratas");
            System.out.println("4. Melihat Daftar Tugas");
            System.out.println("5. Melihat Tugas Terbawah");
            System.out.println("6. Jumlah Tugas Terkumpul");
            System.out.print("Pilih : ");
            pilih = scan.nextInt();
            scan.nextLine();

            switch (pilih) {
                case 1:
                    System.out.print("Nama : ");
                    String nama = scan.nextLine();
                    System.out.print("NIM : ");
                    String nim = scan.nextLine();
                    System.out.print("Kelas: ");
                    String kelas = scan.nextLine();
                    Mahasiswa18 mhs = new Mahasiswa18(nama, nim, kelas);
                    stack.push(mhs);
                    System.out.printf("Tugas %s berhasil dikumpulkan\n", mhs.nama);
                    break;

                case 2:
                    Mahasiswa18 dinilai = stack.pop();
                    if (dinilai != null) {
                        System.out.println("Menilai tugas dari " + dinilai.nama);
                        System.out.print("Masukkan nilai (0-100) : ");
                        int nilai = scan.nextInt();
                        dinilai.tugasDinilai(nilai);
                        System.out.printf("Nilai Tugas %s adalah %d\n", dinilai.nama, nilai);
                        String biner = stack.konversiDesimalKeBiner(nilai);
                        System.out.println("Nilai Biner Tugas: " + biner);
                    }
                    break;

                case 3:
                    Mahasiswa18 lihat = stack.peek();
                    if (lihat != null) {
                        System.out.println("Tugas terakhir dikumpulkan oleh " + lihat.nama);
                    }
                    break;

                case 4:
                    System.out.println("Daftar semua tugas:");
                    System.out.println("Nama\tNIM\tKelas");
                    stack.print();
                    break;

                case 5:
                    Mahasiswa18 terbawah = stack.bottom();
                    if (terbawah != null) {
                        System.out.println("Tugas pertama dikumpulkan oleh " + terbawah.nama);
                    }
                    break;

                case 6:
                    System.out.println("Jumlah tugas terkumpul " + stack.jumlahTugas());
                    break;

                default:
                    System.out.println("Pilihan tidak valid.");
            }
        } while (pilih >= 1 && pilih <= 4);
    }
}

```

Menu:

1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas

Pilih : 1

Nama : Dila

NIM : 1001

Kelas: 1A

Tugas Dila berhasil dikumpulkan

Menu:

1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas

Pilih : 1

Nama : Erik

NIM : 1002

Kelas: 1B

Tugas Erik berhasil dikumpulkan

Menu:

Menu:

1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas

Pilih : 3

Tugas terakhir dikumpulkan oleh Erik

Menu:

1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas

Pilih : 1

Nama : Tika

NIM : 1003

Kelas: 1C

Tugas Tika berhasil dikumpulkan

Menu:

1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas

```
Menu:
1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas
Pilih : 4
```

```
Daftar semua tugas:
Nama    NIM    Kelas
Tika    1003    1C
Erik    1002    1B
Dila    1001    1A
```

```
Menu:
1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas
Pilih : 2
Menilai tugas dari Tika
Masukkan nilai (0-100) : 87
Nilai Tugas Tika adalah 87
```

```
Masukkan nilai (0-100) : 87
Nilai Tugas Tika adalah 87
```

```
Menu:
1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas
Pilih : 4
Daftar semua tugas:
Nama    NIM    Kelas
Erik    1002    1B
Dila    1001    1A
```

https://github.com/nikyrania/asd_jobsheet9/tree/main/Jobsheet9

Pertanyaan

1. Lakukan perbaikan pada kode program, sehingga keluaran yang dihasilkan sama dengan verifikasi hasil percobaan! Bagian mana yang perlu diperbaiki?

Jawaban:

Di ClassStackTugasMahasiswa18

Yang sebelumnya:

```

public void print() {
    for (int i = 0; i ≤ top; i++) {
        System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);
    }
    System.out.println(x:"");
}

```

Menjadi:

```

public void print() {
    for (int i = top; i ≥ 0; i--) { //yang pertama mengumpulkan ada di bawah
        System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);
    }
    System.out.println(x:"");
}

```

2. Berapa banyak data tugas mahasiswa yang dapat ditampung di dalam Stack? Tunjukkan potongan kode programnya!

Jawaban:

Potongan kode yang menentukan kapasitas, yang ditampung di stack ditentukan oleh variabel size yang diset melalui konstruktor

```

public StackTugasMahasiswa18(int size) {
    this.size = size;
    stack = new Mahasiswa18[size];
    top = -1;
}

```

Pada demo, stack dibuat dengan kapasitas 5

```

StackTugasMahasiswa18 stack = new StackTugasMahasiswa18(size:5);

```

3. Mengapa perlu pengecekan kondisi !isFull() pada method push? Kalau kondisi if-else tersebut dihapus, apa dampaknya?

Jawaban:

Perlu karena untuk mencegah penambahan elemen saat stack sudah penuh, agar tidak terjadi overflow (ArrayIndexOutOfBoundsException). Jika dihapus, maka ketika stack penuh dan masih tetap melakukan push, program akan error karena melebihi kapasitas

```

Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 5
at Jebheest0.StackTugasMahasiswa18.push(StackTugasMahasiswa18.java:20)

```

4. Modifikasi kode program pada class MahasiswaDemo dan StackTugasMahasiswa sehingga pengguna juga dapat melihat mahasiswa yang pertama kali mengumpulkan tugas melalui operasi lihat tugas terbawah!

Jawaban:

Di StackTugasMahasiswa18

```

public Mahasiswa18 bottom() {
    if (!isEmpty()) {
        return stack[0];
    } else {
        System.out.println(x:"Stack kosong! Tidak ada tugas terbawah");
        return null;
    }
}

```

Di MahasiswaDemo18

```
case 5:
    Mahasiswa18 terbawah = stack.bottom();
    if (terbawah != null) {
        System.out.println("Tugas pertama dikumpulkan oleh " + terbawah.nama);
    }
```

5. Tambahkan method untuk dapat menghitung berapa banyak tugas yang sudah dikumpulkan saat ini, serta tambahkan operasi menunya!

Jawaban:

Di StackTugasMahasiswa18

```
public int jumlahTugas() {
    return top + 1;
}
```

Di MahasiswaDemo18

```
case 6:
    System.out.println("Jumlah tugas terkumpul " + stack.jumlahTugas());
    break;
```

6. https://github.com/nikytania/asd_jobsheet9/tree/main/Jobsheet9

PERCOBAAN 2

```
StackKonversi18.java

package Jobsheet9;

public class StackKonversi18 {
    int[] tumpukanBiner;
    int top;
    int size;

    public StackKonversi18() {
        this.size = 32; //asumsi 32 bit
        tumpukanBiner = new int[size];
        top = -1;
    }

    public boolean isEmpty() {
        return top == -1;
    }

    public boolean isFull() {
        return top == size - 1;
    }

    public void push(int data) {
        if (isFull()) {
            System.out.println("Stack penuh");
        } else {
            top++;
            tumpukanBiner[top] = data;
        }
    }

    public int pop() {
        if (isEmpty()) {
            System.out.println("Stack kosong.");
            return -1;
        } else {
            int data = tumpukanBiner[top];
            top--;
            return data;
        }
    }
}
```



```

StackTugasMahasiswa18.java

package Jobsheet9;

public class StackTugasMahasiswa18 {
    Mahasiswa18[] stack;
    int top;
    int size;

    public StackTugasMahasiswa18(int size) {
        this.size = size;
        stack = new Mahasiswa18[size];
        top = -1;
    }

    public boolean isFull() {
        if (top == size - 1) {
            return true;
        } else {
            return false;
        }
    }

    public boolean isEmpty() {
        if (top == -1) {
            return true;
        } else {
            return false;
        }
    }

    public void push(Mahasiswa18 mhs) {
        if (!isFull()) {
            top++;
            stack[top] = mhs;
        } else {
            System.out.println("Stack penuh! Tidak bisa menambahkan tugas lagi.");
        }
    }

    public Mahasiswa18 pop() {
        if (!isEmpty()) {
            Mahasiswa18 m = stack[top];
            top--;
            return m;
        } else {
            System.out.println("Stack kosong! Tidak ada tugas untuk dinilai");
            return null;
        }
    }

    public Mahasiswa18 peek() {
        if (!isEmpty()) {
            return stack[top];
        } else {
            System.out.println("Stack kosong! Tidak ada tugas yang dikumpulkan");
            return null;
        }
    }

    public void print() {
        for (int i = top; i >= 0; i--) { //yang pertama mengumpulkan ada di bawah
            System.out.println(stack[i].nama + "\t" + stack[i].nim + "\t" + stack[i].kelas);
        }
        System.out.println("");
    }

    public Mahasiswa18 bottom() {
        if (!isEmpty()) {
            return stack[0];
        } else {
            System.out.println("Stack kosong! Tidak ada tugas terbawah");
            return null;
        }
    }

    public int jumlahTugas() {
        return top + 1;
    }

    public String konversiDesimalKeBiner(int nilai) {
        StackKonversi18 stack = new StackKonversi18();
        while (nilai != 0) {
            int sisa = nilai % 2;
            stack.push(sisa);
            nilai = nilai / 2;
        }
        String biner = new String();
        while (!stack.isEmpty()) {
            biner += stack.pop();
        }
        return biner;
    }
}

```

```

MahasiswaDemo18.java

package Jobsheet9;

import java.util.Scanner;

public class MahasiswaDemo18 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int pilih;

        StackTugasMahasiswa18 stack = new StackTugasMahasiswa18(5);

        do {
            System.out.println("\nMenu:");
            System.out.println("1. Mengumpulkan Tugas");
            System.out.println("2. Menilai Tugas");
            System.out.println("3. Melihat Tugas Teratas");
            System.out.println("4. Melihat Daftar Tugas");
            System.out.println("5. Melihat Tugas Terbawah");
            System.out.println("6. Jumlah Tugas Terkumpul");
            System.out.print("Pilih : ");
            pilih = scan.nextInt();
            scan.nextLine();

            switch (pilih) {
                case 1:
                    System.out.print("Nama : ");
                    String nama = scan.nextLine();
                    System.out.print("NIM : ");
                    String nim = scan.nextLine();
                    System.out.print("Kelas: ");
                    String kelas = scan.nextLine();
                    Mahasiswa18 mhs = new Mahasiswa18(nama, nim, kelas);
                    stack.push(mhs);
                    System.out.printf("Tugas %s berhasil dikumpulkan\n", mhs.nama);
                    break;

                case 2:
                    Mahasiswa18 dinilai = stack.pop();
                    if (dinilai != null) {
                        System.out.println("Menilai tugas dari " + dinilai.nama);
                        System.out.print("Masukkan nilai (0-100) : ");
                        int nilai = scan.nextInt();
                        dinilai.tugasDinilai(nilai);
                        System.out.printf("Nilai Tugas %s adalah %d\n", dinilai.nama, nilai);
                        String biner = stack.konversiDesimalKeBiner(nilai);
                        System.out.println("Nilai Biner Tugas: " + biner);
                    }
                    break;

                case 3:
                    Mahasiswa18 lihat = stack.peek();
                    if (lihat != null) {
                        System.out.println("Tugas terakhir dikumpulkan oleh " + lihat.nama);
                    }
                    break;

                case 4:
                    System.out.println("Daftar semua tugas:");
                    System.out.println("Nama\tNIM\tKelas");
                    stack.print();
                    break;

                case 5:
                    Mahasiswa18 terbawah = stack.bottom();
                    if (terbawah != null) {
                        System.out.println("Tugas pertama dikumpulkan oleh " + terbawah.nama);
                    }
                    break;

                case 6:
                    System.out.println("Jumlah tugas terkumpul " + stack.jumlahTugas());
                    break;

                default:
                    System.out.println("Pilihan tidak valid.");
            }
        } while (pilih >= 1 && pilih <= 4);
    }
}

```

Menu:

```
1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas
5. Melihat Tugas Terbawah
6. Jumlah Tugas Terkumpul
Pilih : 2
Menilai tugas dari Tika
Masukkan nilai (0-100) : 87
Nilai Tugas Tika adalah 87
Nilai Biner Tugas: 1010111
```

https://github.com/nikytonia/asd_jobsheet9/tree/main/Jobsheet9

Pertanyaan

1. Jelaskan alur kerja dari method konversiDesimalKeBiner!

Jawaban:

Inisialisasi stack kosong: StackKonversi18 dipakai untuk menyimpan hasil pembagian sisa (0 atau 1)

```
StackKonversi18 stack = new StackKonversi18();
```

Perulangan pembagian dengan 2:

Nilai desimal dibagi 2 berulang-ulang

Sisa pembagian (nilai % 2) dimasukkan ke stack

Nilai terus dibagi 2 hingga hasilnya 0

```
while (nilai > 0) {
    int sisa = nilai % 2;
    stack.push(sisa);
    nilai = nilai / 2;
}
```

Konversi ke string:

Setelah selesai, stack berisi angka-angka biner dalam urutan terbalik (dari bawah ke atas)

Maka, program pop satu per satu dari stack untuk menyusun urutan biner yang benar

```
String biner = new String();
while (!stack.isEmpty()) {
    biner += stack.pop();
}
```

Return hasil biner: setelah semua angka dikeluarkan dari stack, string hasilnya dikembalikan

```
return biner;
```

2. Pada method konversiDesimalKeBiner, ubah kondisi perulangan menjadi while (kode != 0), bagaimana hasilnya? Jelaskan alasannya!

Jawaban:

```
StackKonversi10 Stack = ne  
while (nilai != 0) {  
    int sisa = nilai % 2;
```

Jika kondisi perulangan diubah menjadi *while (nilai != 0)*, hasilnya tetap sama. Hal ini karena nilai yang digunakan merupakan bilangan bulat positif (antara 0 hingga 100), sehingga baik menggunakan kondisi *nilai > 0* maupun *nilai != 0* akan memberikan proses perulangan yang identik.

Selama nilai lebih dari nol, kedua kondisi tersebut akan tetap menjalankan perulangan. Sebaliknya, ketika nilai sudah sama dengan nol, keduanya akan menghentikan perulangan. Dengan demikian, perubahan kondisi tersebut tidak memengaruhi hasil akhir dari proses konversi bilangan desimal ke biner

PRAKTIKUM

```
package Jobsheet9;

public class Surat18 {
    String idSurat;
    String namaMahasiswa;
    String kelas;
    char jenisIzin;
    int durasi;

    public Surat18(String idSurat, String namaMahasiswa, String kelas, char jenisIzin, int durasi) {
        this.idSurat = idSurat;
        this.namaMahasiswa = namaMahasiswa;
        this.kelas = kelas;
        this.jenisIzin = jenisIzin;
        this.durasi = durasi;
    }

    public void tampilkanSurat() {
        System.out.println("ID Surat      : " + idSurat);
        System.out.println("Nama        : " + namaMahasiswa);
        System.out.println("Kelas      : " + kelas);
        System.out.println("Jenis Izin  : " + (jenisIzin == 'S' ? "Sakit" : "Izin"));
        System.out.println("Durasi (hari): " + durasi);
    }
}
```

```
StackSurat18.java

package Jobsheet9;

public class StackSurat18 {
    Surat18[] stack;
    int top;
    int size;

    public StackSurat18(int size) {
        this.size = size;
        stack = new Surat18[size];
        top = -1;
    }

    public boolean isFull() {
        return top == size - 1;
    }

    public boolean isEmpty() {
        return top == -1;
    }

    public void push(Surat18 s) {
        if (!isFull()) {
            stack[++top] = s;
        } else {
            System.out.println("Stack penuh! Tidak bisa menerima surat lagi");
        }
    }

    public Surat18 pop() {
        if (!isEmpty()) {
            return stack[top--];
        } else {
            System.out.println("Stack kosong! Tidak ada surat yang diproses");
            return null;
        }
    }

    public Surat18 peek() {
        if (!isEmpty()) {
            return stack[top];
        } else {
            System.out.println("Stack kosong! Tidak ada surat");
            return null;
        }
    }

    public void cariSurat(String nama) {
        boolean ditemukan = false;
        for (int i = 0; i <= top; i++) {
            if (stack[i].namaMahasiswa.equalsIgnoreCase(nama)) {
                System.out.println("Surat ditemukan:");
                stack[i].tampilkanSurat();
                ditemukan = true;
                break;
            }
        }
        if (!ditemukan) {
            System.out.println("Surat dari mahasiswa tersebut tidak ditemukan");
        }
    }
}
```

```

SuratMain18.java

package Jobsheet9;
import java.util.Scanner;

public class SuratMain18 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        StackSurat18 stack = new StackSurat18(5);

        int pilihan;
        do {
            System.out.println("\nMENU:");
            System.out.println("1. Terima Surat Izin");
            System.out.println("2. Proses Surat Izin");
            System.out.println("3. Lihat Surat Izin Terakhir");
            System.out.println("4. Cari Surat");
            System.out.println("0. Keluar");
            System.out.print("Pilih menu: ");
            pilihan = sc.nextInt();
            sc.nextLine();

            switch (pilihan) {
                case 1:
                    System.out.print("ID Surat: ");
                    String id = sc.nextLine();
                    System.out.print("Nama Mahasiswa: ");
                    String nama = sc.nextLine();
                    System.out.print("Kelas: ");
                    String kelas = sc.nextLine();
                    System.out.print("Jenis Izin (S/I): ");
                    char jenis = sc.nextLine().toUpperCase().charAt(0);
                    System.out.print("Durasi Izin (hari): ");
                    int durasi = sc.nextInt();
                    sc.nextLine();

                    Surat18 surat = new Surat18(id, nama, kelas, jenis, durasi);
                    stack.push(surat);
                    System.out.println("Surat berhasil ditambahkan");
                    break;

                case 2:
                    Surat18 suratDiproses = stack.pop();
                    if (suratDiproses != null) {
                        System.out.println("Surat yang diproses:");
                        suratDiproses.tampilkanSurat();
                    }
                    break;

                case 3:
                    Surat18 suratTerakhir = stack.peek();
                    if (suratTerakhir != null) {
                        System.out.println("Surat terakhir yang masuk:");
                        suratTerakhir.tampilkanSurat();
                    }
                    break;

                case 4:
                    System.out.print("Masukkan nama mahasiswa: ");
                    String cariNama = sc.nextLine();
                    stack.cariSurat(cariNama);
                    break;

                case 0:
                    System.out.println("Program selesai");
                    break;

                default:
                    System.out.println("Pilihan tidak valid");
            }
        } while (pilihan != 0);
    }
}

```

MENU:

1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
0. Keluar

Pilih menu: 1

ID Surat: 123

Nama Mahasiswa: Maura

Kelas: 2B

Jenis Izin (S/I): S

Durasi Izin (hari): 3

Surat berhasil ditambahkan

MENU:

1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
0. Keluar

Pilih menu: 1

ID Surat: 345

Nama Mahasiswa: Nana

Kelas: 4E

Nama Mahasiswa: Nana

Kelas: 4E

Jenis Izin (S/I): I

Durasi Izin (hari): 4

Surat berhasil ditambahkan

MENU:

1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
0. Keluar

Pilih menu: 3

Surat terakhir yang masuk:

ID Surat : 345

Nama : Nana

Kelas : 4E

Jenis Izin : Izin

Durasi (hari): 4

MENU:

1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat


```
MENU:
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
0. Keluar
Pilih menu: 4
Masukkan nama mahasiswa: Maura
Surat ditemukan:
ID Surat      : 123
Nama          : Maura
Kelas        : 2B
Jenis Izin    : Sakit
Durasi (hari): 3
```

```
MENU:
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
0. Keluar
Pilih menu: 0
Program selesai
PS C:\Praktikum ASD> █
```

https://github.com/nikytania/asd_jobsheet9/tree/main/Jobsheet9