

# **Laporan Hasil Praktikum Algoritma Dan Struktur Data**

## **Jobsheet 10**



Disusun Oleh :

Nama : Fadhil Taufiqurrachman

NIM : 244107020090

Kelas : Teknik Informatika 1E

**Program Studi Teknik Informatika**

**Jurusan Teknologi Informasi**

**Politeknik Negeri Malang 2025**

## 2.1 Percobaan 1: Mahasiswa Mengumpulkan Tugas

### 2.2.1 Kode Program

Kode program pada class Mahasiswa08 :

```
package Jobsheet10;

public class Mahasiswa08 {
    String nim, nama, kelas;
    int nilai;

    Mahasiswa08() {

    }

    Mahasiswa08(String nama, String nim, String kelas) {
        this.nama = nama;
        this.nim = nim;
        this.kelas = kelas;
        nilai = -1;
    }

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

Kode program pada class StackTugasMahasiswa08 :

```
package Jobsheet10;

public class StackTugasMahasiswa08 {
    Mahasiswa08[] stack;
    int size, top;

    public StackTugasMahasiswa08(int size) {
        this.size = size;
        stack = new Mahasiswa08[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(Mahasiswa08 mhs) {
    if (!isFull()) {
        top++;
        stack[top] = mhs;
    } else {
        System.out.println("Stack Penuh! Tidak Bisa Menambahkan
Tugas Lagi.");
    }
}

public Mahasiswa08 pop() {
    if (!isEmpty()) {
        Mahasiswa08 m = stack[top];
        top--;
        return m;
    } else {
        System.out.println("Stack Kosong! Tidak Ada Tugas Yang Untuk
Dinilai.");
        return null;
    }
}

public Mahasiswa08 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 = 0; i <= top; i++) {
        System.out.println(stack[i].nama + "\t" + stack[i].nim +
"\t" + stack[i].kelas);
    }
    System.out.println();
}
}

```



```

case 3:
    Mahasiswa08 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;
default:
    System.out.println("Pilihan Tidak Valid.");
}
} while (pilih >= 1 && pilih <= 4);
input.close();
}

```

### 2.2.2 Verifikasi

```

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 :
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
Pilih : 4
Daftar Semua Tugas
Nama    NIM    Kelas
Dila    1001    1A
Erik    1002    1B
Tika    1003    1C

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

```

```

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

Menu :
1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas
Pilih : 0
Pilihan Tidak Valid.

```

### 2.2.3 Pertanyaan

1. Memodifikasi pada method print yang ada pada class StackTugasMahasiswa08.

Karena sebelumnya pada kode tersebut tidak menerapkan konsep LIFO (Last In Last Out), maka dari itu harus di modifikasi agar sesuai dengan konsep. Berikut di bawah ini merupakan hasil kode program yang sudah dimodifikasi.

```

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

```

2. Terdapat pada class MahasiswaDemo08, berikut adalah kode programnya yang mendefinisikan banyaknya data yang dapat ditampung dalam sebuah stack.

```
StackTugasMahasiswa08 stack = new StackTugasMahasiswa08(5);
```

3. Untuk menghindari sebuah error yang bernama StackOverflow, yang artinya yaitu stack sudah penuh namun masih ingin menambahkan data atau melebihi jumlah yang dapat ditampung oleh sebuah stack. Maka dari itu, sebelum menambahkan data kita harus mengecek isFull terlebih dahulu untuk mencari tahu apakah stack sudah penuh atau belum.

4. Berikut adalah hasil modifikasi kode programnya.

**Class StackTugasMahasiswa08.**

```
public Mahasiswa08 bottom() {
    if (!isEmpty()) {
        return stack[0];
    } else {
        System.out.println("Stack Kosong! Tidak Ada Tugas Yang
Dikumpulkan.");
        return null;
    }
}
```

**Class MahasiswaDemo08.**

```
case 5:
    Mahasiswa08 terbawah = stack.bottom();
    if (terbawah != null) {
        System.out.println("Tugas Terbawah Adalah " +
terbawah.nama);
    }
    break;
```

5. Berikut di bawah ini adalah kode programnya.

**Menambahkan kode program pada class StackTugasMahasiswa08 :**

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

Menambahkan kode program pada class MahasiswaDemo08 :

```
case 6:
    int total = stack.jumlahTugas();
    if (total > 0) {
        System.out.println("Total Tugas Yang Sudah
Dikumpulkan : " + total);
    } else {
        System.out.println("Tidak Ada Tugas Yang
Dikumpulkan.");
    }
    break;
```

## 6. Push Dan Commit Ke Akun GitHub.

```
PS C:\File Kuliah\Semester 2\Praktikum Algoritma Dan Struktur Data\GitHub\Praktikum-ASD> cd Jobsheet10
PS C:\File Kuliah\Semester 2\Praktikum Algoritma Dan Struktur Data\GitHub\Praktikum-ASD\Jobsheet10> git add Mahasiswa08.java StackTugasMahasiswa08.java Mahasis
waDemo08.java
PS C:\File Kuliah\Semester 2\Praktikum Algoritma Dan Struktur Data\GitHub\Praktikum-ASD\Jobsheet10> git commit -m "Percobaan 1"
[main c859b7f] Percobaan 1
3 files changed, 171 insertions(+)
create mode 100644 Jobsheet10/Mahasiswa08.java
create mode 100644 Jobsheet10/MahasiswaDemo08.java
create mode 100644 Jobsheet10/StackTugasMahasiswa08.java
PS C:\File Kuliah\Semester 2\Praktikum Algoritma Dan Struktur Data\GitHub\Praktikum-ASD\Jobsheet10> git push origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 1.80 KiB | 167.00 KiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/FadhilTaufiqurrachman/Praktikum-ASD.git
b5ffff2..c859b7f  main -> main
```



## 2.2 Percobaan 2: Konversi Nilai Tugas Ke Biner

### 2.2.1 Kode Program

Menambahkan kode program pada class StackTugasMahasiswa08 :

```
public String konversiDesimalKeBiner(int nilai) {
    StackKonversi08 stack = new StackKonversi08();
    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;
}
```

Menambahkan kode program pada class MahasiswaDemo08, Penambahan yaitu pada teks berwarna :

```
case 2:
    Mahasiswa08 dinilai = stack.pop();
    if (dinilai != null) {
        System.out.println("Menilai Tugas Dari " +
dinilai.nama);
        System.out.print("Masukkan Nilai (0-100) : ");
        int nilai = input.nextInt();
        dinilai.tugasDinilai(nilai);
        System.out.printf("Nilai Tugas %s Adalah %d\n",
dinilai.nama, dinilai.nilai);
        String biner =
stack.konversiDesimalKeBiner(nilai);
        System.out.println("Nilai Biner Tugas : " +
biner);
    }
    break;
```

Kode program pada class StackKonversi08 :

```
package Jobsheet10;

public class StackKonversi08 {
    int[] tumpukanBiner;
    int size, top;

    public StackKonversi08() {
        this.size = 32;
        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;
        }
    }
}
```

### 2.3.2 Verifikasi

```
Menu :
1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas
5. Melihat Tugas Terbawah
6. Total Tugas Yang Sudah Dikumpulkan
Pilih : 2
Menilai Tugas Dari Tika
Masukkan Nilai (0-100) : 87
Nilai Tugas Tika Adalah 87
Nilai Biner Tugas : 1010111
```

### 2.3.3 Pertanyaan

1. Pertama tama yaitu menginstansiasi objek Bernama stack. Kemudian selama nilai lebih dari 0, maka secara berulang kali akan membagi bilangan desimal dengan 2 dan menyimpan sisa hasil baginya ke dalam sebuah stack. Setelah proses pembagian selesai, sisa-sisa hasil bagi diambil dari stack dan digabungkan menjadi sebuah string. Karena stack memiliki sifat LIFO, maka urutan pengambilan sisa hasil bagi akan terbalik, dan akan menghasilkan representasi biner yang benar.
2. Program akan tetap berjalan tidak ada yang berbeda dari sebelumnya, hasilnya tetap sama dengan ketika while(nilai>0). Namun terdapat satu celah yaitu ketika kita menginputkan nilai suatu tugas dengan bilangan negatif, maka hasilnya tidak akan sesuai dengan hasil yang seharusnya, Di depan bilangan biner 1 akan terdapat negatifnya.

```
Menu :
1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas
5. Melihat Tugas Terbawah
6. Total Tugas Yang Sudah Dikumpulkan
Pilih : 2
Menilai Tugas Dari Tika
Masukkan Nilai (0-100) : -87
Nilai Tugas Tika Adalah -87
Nilai Biner Tugas : -10-10-1-1-1
```

### 3. Push Dan Commit Ke Akun GitHub.

```
PS C:\File Kuliah\Semester 2\Praktikum Algoritma Dan Struktur Data\GitHub\Praktikum-ASD\Jobsheet10> git add StackTugasMahasiswa08.java StackKonversi08.java Mah
asiswaDemo08.java
PS C:\File Kuliah\Semester 2\Praktikum Algoritma Dan Struktur Data\GitHub\Praktikum-ASD\Jobsheet10> git commit -m "Percobaan 2"
[main 296dea5] Percobaan 2
3 files changed, 56 insertions(+)
create mode 100644 Jobsheet10/StackKonversi08.java
PS C:\File Kuliah\Semester 2\Praktikum Algoritma Dan Struktur Data\GitHub\Praktikum-ASD\Jobsheet10> git push origin main
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 1.05 KiB | 214.00 KiB/s, done.
Total 6 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/FadhilTaufiqurrachman/Praktikum-ASD.git
c859b7f..296dea5 main -> main
```

## 2.4 Latihan Praktikum

### 2.4.1 Kode Program

Kode program pada class Surat08 :

```
package Jobsheet10;

public class Surat08 {
    String idSurat, namaMahasiswa, kelas;
    char jenisIzin;
    int durasi;

    Surat08() {

    }

    Surat08(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 Mahasiswa    : " + namaMahasiswa);
        System.out.println("Kelas          : " + kelas);
        System.out.println("Jenis Izin      : " + (jenisIzin == 'S' ?
"Sakit" : "Izin Keperluan Lain"));
        System.out.println("Durasi          : " + durasi + " Hari");
        System.out.println();
    }
}
```

### Kode program pada class StackSurat08 :

```
package Jobsheet10;

public class StackSurat08 {
    Surat08[] data;
    int top, size;

    public StackSurat08(int size) {
        this.size = size;
        data = new Surat08[size];
        top = -1;
    }

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

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

    public void push(Surat08 surat) {
        if (!isFull()) {
            top++;
            data[top] = surat;
            System.out.println("Surat Berhasil Ditambahkan!");
        } else {
            System.out.println("Stack Penuh! Tidak Bisa Menambahkan Surat Lagi.");
        }
    }
}
```

```

public Surat08 pop() {
    if (!isEmpty()) {
        Surat08 surat = data[top];
        top--;
        return surat;
    } else {
        System.out.println("Stack Kosong! Tidak Ada Surat Yang Dapat
Diambil.");
        return null;
    }
}

public Surat08 peek() {
    if (!isEmpty()) {
        return data[top];
    } else {
        System.out.println("Stack Kosong! Tidak Ada Surat Yang Dapat
Dilihat.");
        return null;
    }
}

public void cariSurat(String namaMahasiswa) {
    boolean ditemukan = false;
    if (isEmpty()) {
        System.out.println("Stack Kosong! Tidak Ada Surat Yang Dapat
Dicari.");
        return;
    }
    for (int i = 0; i <= top; i++) {
        if (data[i].namaMahasiswa.equalsIgnoreCase(namaMahasiswa)) {
            data[i].tampilkanSurat();
            ditemukan = true;
        }
    }
    if (!ditemukan) {
        System.out.println("Surat Dengan Nama Mahasiswa " +
namaMahasiswa + " Tidak Ditemukan.");
    }
}
}

```

### Kode program pada class SuratMain08 :

```
package Jobsheet10;

import java.util.Scanner;
public class SuratMain08 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        StackSurat08 stack = new StackSurat08(10);
        int pilihan;

        do {
            System.out.println("===== Menu Program Surat Mahasiswa
=====");
            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("5. Lihat Semua Surat Yang Belum
Diproses.");
            System.out.println("0. Keluar.");
            System.out.println("=====
==");

            System.out.print("Pilih Menu : ");
            pilihan = input.nextInt();
            input.nextLine();

            switch (pilihan) {
                case 1:
                    System.out.println("\n----- Terima Surat Izin -----
-");
                    System.out.print("Masukkan ID
Surat          : ");
                    String idSurat = input.nextLine();
                    System.out.print("Masukkan Nama
Mahasiswa      : ");
                    String namaMahasiswa = input.nextLine();
                    System.out.print("Masukkan
Kelas         : ");
                    String kelas = input.nextLine();
                    char jenisIzin;
                    do {
                        System.out.print("Masukkan Jenis Izin (S
(Sakit), I (Izin)) : ");
                        jenisIzin =
Character.toUpperCase(input.nextLine().charAt(0));
                        if (jenisIzin != 'S' && jenisIzin != 'I') {
                            System.out.println("Input Tidak Valid. Harap
Masukkan 'S' Atau 'I'.");
                        }
                    } while (jenisIzin != 'S' && jenisIzin != 'I');
                    System.out.print("Masukkan Durasi
(Hari)         : ");
                    int durasi = input.nextInt();
                    input.nextLine();
                    Surat08 suratBaru = new Surat08(idSurat,
namaMahasiswa, kelas, jenisIzin, durasi);
                    stack.push(suratBaru);
                    System.out.println();
                    break;
            }
        } while (pilihan != 0);
    }
}
```

```

        case 2:
            System.out.println("\n----- Proses Surat Izin -----");
            Surat08 suratDiproses = stack.pop();
            if (suratDiproses != null) {
                System.out.println("Surat Berikut Telah Diproses
/ Diverifikasi :");
                suratDiproses.tampilkanSurat();
            }
            System.out.println();
            break;
        case 3:
            System.out.println("\n----- Lihat Surat Izin
Terakhir -----");
            Surat08 suratTerakhir = stack.peek();
            if (suratTerakhir != null) {
                System.out.println("Surat Izin Terakhir
(Teratas) Adalah :");
                suratTerakhir.tampilkanSurat();
            }
            System.out.println();
            break;
        case 4:
            System.out.println("\n----- Cari Surat -----");
            System.out.print("Masukkan Nama Mahasiswa Yang
Dicari : ");
            String namaCari = input.nextLine();
            stack.cariSurat(namaCari);
            System.out.println();
            break;
        case 5:
            System.out.println("\n----- Lihat Semua Surat Yang
Belum Diproses -----");
            if (stack.isEmpty()) {
                System.out.println("Tidak Ada Surat Yang
Tersimpan.");
            } else {
                System.out.println("Daftar Surat Izin Yang
Tersimpan :");
                for (int i = stack.top; i >= 0; i--) {
                    stack.data[i].tampilkanSurat();
                }
            }
            System.out.println();
            break;
        case 0:
            System.out.println("Keluar Dari Program, Terima
Kasih!\n");
            break;
        default:
            System.out.println("Pilihan Tidak Valid! Silakan
Coba Lagi.\n");
    }
    while (pilihan != 0);
}

```



## 2.4.2 Hasil Output

```
===== Menu Program Surat Mahasiswa =====
1. Terima Surat Izin.
2. Proses Surat Izin.
3. Lihat Surat Izin Terakhir.
4. Cari Surat.
5. Lihat Semua Surat Yang Belum Diproses.
0. Keluar.
=====
Pilih Menu : 1

----- Terima Surat Izin -----
Masukkan ID Surat           : 001
Masukkan Nama Mahasiswa     : Aruna Prabandari
Masukkan Kelas              : 1J
Masukkan Jenis Izin (S (Sakit), I (Izin)) : I
Masukkan Durasi (Hari)      : 5
Surat Berhasil Ditambahkan!

===== Menu Program Surat Mahasiswa =====
1. Terima Surat Izin.
2. Proses Surat Izin.
3. Lihat Surat Izin Terakhir.
4. Cari Surat.
5. Lihat Semua Surat Yang Belum Diproses.
0. Keluar.
=====
Pilih Menu : 1

----- Terima Surat Izin -----
Masukkan ID Surat           : 002
Masukkan Nama Mahasiswa     : Kanya Samudra
Masukkan Kelas              : 1L
Masukkan Jenis Izin (S (Sakit), I (Izin)) : S
Masukkan Durasi (Hari)      : 3
Surat Berhasil Ditambahkan!
```

===== Menu Program Surat Mahasiswa =====

1. Terima Surat Izin.
2. Proses Surat Izin.
3. Lihat Surat Izin Terakhir.
4. Cari Surat.
5. Lihat Semua Surat Yang Belum Diproses.
0. Keluar.

=====

Pilih Menu : 3

----- Lihat Surat Izin Terakhir -----

Surat Izin Terakhir (Teratas) Adalah :

ID Surat : 002  
Nama Mahasiswa : Kanya Samudra  
Kelas : 1L  
Jenis Izin : Sakit  
Durasi : 3 Hari

===== Menu Program Surat Mahasiswa =====

1. Terima Surat Izin.
2. Proses Surat Izin.
3. Lihat Surat Izin Terakhir.
4. Cari Surat.
5. Lihat Semua Surat Yang Belum Diproses.
0. Keluar.

=====

Pilih Menu : 4

----- Cari Surat -----

Masukkan Nama Mahasiswa Yang Dicari : Aruna Prabandari

ID Surat : 001  
Nama Mahasiswa : Aruna Prabandari  
Kelas : 1J  
Jenis Izin : Izin Keperluan Lain  
Durasi : 5 Hari

===== Menu Program Surat Mahasiswa =====

1. Terima Surat Izin.
2. Proses Surat Izin.
3. Lihat Surat Izin Terakhir.
4. Cari Surat.
5. Lihat Semua Surat Yang Belum Diproses.
0. Keluar.

=====

Pilih Menu : 2

----- Proses Surat Izin -----

Surat Berikut Telah Diproses / Diverifikasi :

ID Surat : 002  
Nama Mahasiswa : Kanya Samudra  
Kelas : 1L  
Jenis Izin : Sakit  
Durasi : 3 Hari

```
===== Menu Program Surat Mahasiswa =====
1. Terima Surat Izin.
2. Proses Surat Izin.
3. Lihat Surat Izin Terakhir.
4. Cari Surat.
5. Lihat Semua Surat Yang Belum Diproses.
0. Keluar.
=====
Pilih Menu : 5

----- Lihat Semua Surat Yang Belum Diproses -----
Daftar Surat Izin Yang Tersimpan :
ID Surat      : 001
Nama Mahasiswa : Aruna Prabandari
Kelas        : 1J
Jenis Izin    : Izin Keperluan Lain
Durasi        : 5 Hari

===== Menu Program Surat Mahasiswa =====
1. Terima Surat Izin.
2. Proses Surat Izin.
3. Lihat Surat Izin Terakhir.
4. Cari Surat.
5. Lihat Semua Surat Yang Belum Diproses.
0. Keluar.
=====
Pilih Menu : 0
Keluar Dari Program, Terima Kasih!
```