

LAPORAN HASIL PRAKTIKUM
ALGORITMA DAN STRUKTUR DATA
JOBSHEET 9



ATHAULLA HAFIZH

244107020030

TI 1 E

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

2024

2. Praktikum

Percobaan 1

Kode program class Mahasiswa05

```
package Jobsheet9;

public class Mahasiswa05 {

    String nim, nama, kelas;
    int nilai;

    Mahasiswa05() {
    }

    Mahasiswa05(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 class StackTugasMahasiswa05

```
package Jobsheet9;

public class StackTugasMahasiswa05 {

    Mahasiswa05[] stack;

    int size;

    int top;
```

```
public StackTugasMahasiswa05(int size) {
    this.size = size;
    stack = new Mahasiswa05[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(Mahasiswa05 mhs) {
    if (!isFull()) {
        top++;
        stack[top] = mhs;
    } else {
        System.out.println("Stack penuh! tidak bisa menambahkan tugas lagi!");
    }
}

public Mahasiswa05 pop() {
    if (!isEmpty()) {
        Mahasiswa05 m = stack[top];
        top--;
    }
}
```

```

        return m;
    } else {
        System.out.println("Stack kosong! tidak ada tugas untuk
dinilai");
        return null;
    }
}

public Mahasiswa05 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("");
}
}

```

Kode program class MahasiswaDemo05

```

package Jobsheet9;
import java.util.Scanner;
public class MahasiswaDemo05 {
    public static void main(String[] args) {
        StackTugasMahasiswa05 stack = new StackTugasMahasiswa05(5);
        Scanner scan = new Scanner(System.in);
        int pilih;
    }
}

```

```

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.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();
            Mahasiswa05 mhs = new Mahasiswa05(nama, nim,
kelas);

            stack.push(mhs);

            System.out.printf("Tugas %s berhasil
dikumpulkan\n", mhs.nama);

            break;
        case 2:
            Mahasiswa05 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);
            }

            break;
        case 3:
            Mahasiswa05 lihat = stack.peek();

```

```

        if (lihat != null) {
            System.out.println("Tugas terakhir
dikumpulkan oleh " + lihat.nama);
        }
        break;
    case 4:
        System.out.println("\nDaftar semua tugas:");
        System.out.println("Nama\tNIM\tNilai");
        stack.print();
        break;
    default:
        System.out.println("Pilihan tidak valid.");
    }
} while (pilih >= 1 && pilih <= 4);
}
}

```

Output

```

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	Nilai
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	Nilai
Dila	1001	1A
Erik	1002	1B

Pertanyaan

1. Kode program yang diganti

```
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. Sejumlah data yang diinputkan saat menginisiasi objek StackTugasMahasiswa05, kalau dalam contoh menginputkan 5

```
StackTugasMahasiswa05 stack = new StackTugasMahasiswa05(5);
```

3. Karena jika tidak ada kondisi tersebut, akan menyebabkan error (index out of bounds) apabila dalam suatu kasus sebuah array telah terisi penuh dan tetap dipaksa untuk mengisi elemen pada array tersebut.

4. Tambahkan method melihat tugas terbawah

```
public Mahasiswa05 peekBottom() {  
    if (!isEmpty()) {  
        return stack[0];  
    } else {  
        System.out.println("Stack kosong! Tidak ada tugas yang  
dikumpulkan");  
        return null;  
    }  
}
```

5. Tambah method menghitung jumlah tugas

```
int jumlahTugas() {  
    if (!isEmpty()) {  
        return top + 1;  
    } else {
```



```
        return 0;
    }
}
```

Percobaan 2

Kode program yang ditambahkan

```
public String konversiDesimalKeBiner(int nilai) {
    StackKonversi05 stack = new StackKonversi05();
    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;
}
```

```
package Jobsheet9;

public class StackKonversi05 {
    int[] tumpukanBiner;
    int size;
    int top;

    public StackKonversi05() {
        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;
        }
    }
}

```

Pertanyaan

1. Proses Konversi:

- Selama nilai lebih besar dari 0, maka :
 - Hitung sisa dari nilai ketika dibagi 2 ($\text{sisa} = \text{nilai} \% 2$). Sisa ini akan menjadi digit biner (0 atau 1).
 - Simpan sisa tersebut ke dalam stack dengan memanggil `stack.push(sisa)`.
 - Perbarui nilai dengan hasil pembagian nilai dengan 2 ($\text{nilai} = \text{nilai} / 2$).

Membangun String Biner:

- Setelah semua digit biner disimpan dalam stack, sebuah string kosong biner diinisialisasi.

- Selama stack tidak kosong, digit biner diambil dari stack (dengan `stack.pop()`) dan ditambahkan ke string biner. Ini dilakukan untuk membalik urutan digit, karena digit biner yang dihasilkan dari proses konversi disimpan dalam urutan terbalik.

2. Jika kita mengubah kondisi perulangan dari `while (nilai > 0)` menjadi `while (nilai != 0)`, maka secara fungsional tidak ada perubahan yang signifikan dalam konteks ini, karena kedua kondisi tersebut akan menghasilkan hasil yang sama selama nilai adalah bilangan bulat non-negatif.

Namun, jika nilai diubah menjadi negatif, maka `while (nilai > 0)` tidak akan pernah dieksekusi, sedangkan `while (nilai != 0)` akan tetap dieksekusi. Ini bisa menyebabkan loop yang tidak diinginkan atau kesalahan, tergantung pada bagaimana nilai negatif ditangani dalam konteks konversi biner.

Latihan Praktikum

Kode program Surat05

```
package Jobsheet9;

public class Surat05 {

    String idSurat,namaMahasiswa,kelas;

    char jenisIzin;

    int durasi;

    boolean statusCek;

    Surat05() {

    }

    Surat05(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;

        this.statusCek = false;

    }

}
```

```
void suratDicek() {  
    this.statusCek = true;  
}  
}
```

Kode program StackSurat05

```
package Jobsheet9;  
public class StackSurat05 {  
    Surat05[] stack;  
    int top;  
    int size;  
  
    public StackSurat05(int size) {  
        this.size = size;  
        stack = new Surat05[size];  
        top = -1;  
    }  
    public boolean isFull() {  
        return top == size - 1;  
    }  
    public boolean isEmpty() {  
        return top == -1;  
    }  
  
    public void push(Surat05 srt) {  
        if (!isFull()) {  
            top++;  
            stack[top] = srt;  
        } else {  
            System.out.println("Stack penuh! Tidak bisa menambahkan surat lagi.");  
        }  
    }  
    public Surat05 pop() {  
        if (!isEmpty()) {
```

```

        Surat05 s = stack[top];
        top--;
        return s;
    } else {
        System.out.println("Stack kosong! Tidak ada surat untuk
dicek.");
        return null;
    }
}

public Surat05 peek() {
    if (!isEmpty()) {
        return stack[top];
    } else {
        System.out.println("Stack kosong! Tidak ada surat yang
dikumpulkan.");
        return null;
    }
}
}

```

Kode program MainSurat05

```

package Jobsheet9;
import java.util.Scanner;
public class MainSurat05 {
    public static void main(String[] args) {
        StackSurat05 stack = new StackSurat05(10);
        Scanner input = new Scanner(System.in);
        int menu;
        Surat05[] srtTerverif = new Surat05[10];
        int jmlVerif = 0;

        do {
            System.out.println("\n===== MENU =====");
            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. Keluar Program");
        System.out.print("Pilih : ");
        menu = input.nextInt();
        input.nextLine();

        switch (menu) {
            case 1:
                System.out.println();
                System.out.print("ID Surat : ");
                String idSurat = input.nextLine();
                System.out.print("Nama Mahasiswa : ");
                String namaMahasiswa = input.nextLine();
                System.out.print("Kelas : ");
                String kelas = input.nextLine();
                System.out.print("Jenis perizinan (I/S) : ");
                char jenisIzin = input.next().charAt(0);
                input.nextLine();
                System.out.print("Lama waktu izin : ");
                int durasi = input.nextInt();
                input.nextLine();

                Surat05 srt = new Surat05(idSurat,
namaMahasiswa, kelas, jenisIzin, durasi);
                stack.push(srt);

                System.out.printf("Surat %s berhasil
dikumpulkan\n", srt.namaMahasiswa);
                break;

            case 2:
                Surat05 dicek = stack.pop();
                if (dicek != null) {
                    System.out.println();
                    System.out.println("Memverifikasi surat dari
" + dicek.namaMahasiswa);
                    dicek.suratDicek();
                    Surat05 temp = dicek;

```

```

        srtTerverif[jmlVerif] = temp;

        System.out.printf("Surat dari mahasiswa %s
sudah di verifikasi.\n", dicek.namaMahasiswa);

        jmlVerif++;
    }

    break;

case 3:
    Surat05 lihat = stack.peek();
    if (lihat != null) {
        System.out.println("\nSurat terakhir
dikumpulkan oleh : " + lihat.namaMahasiswa);
    } else {
        System.out.println("\nTidak ada surat yang
dikumpulkan.");
    }

    break;

case 4:
    System.out.println();
    System.out.print("Masukkan nama mahasiswa yang
ingin dicari : ");

    String namaCari = input.nextLine();
    boolean find = false;
    int posisi = -1;
    for (int j = 0; j < stack.size; j++) {
        if (stack.stack[j] != null &&
stack.stack[j].namaMahasiswa.equalsIgnoreCase(namaCari)) {
            find = true;
            posisi = j;
            break;
        }
    }

    if (!find) {
        for (int j = 0; j < srtTerverif.length; j++)
        {
            if (srtTerverif[j] != null &&
srtTerverif[j].namaMahasiswa.equalsIgnoreCase(namaCari)) {
                find = true;

```

```

                posisi = j;
                break;
            }
        }
    }
    if (find) {
        if (posisi >= 0 && posisi < stack.size &&
stack.stack[posisi] != null) {
            Surat05 suratDitemukan =
stack.stack[posisi];

            System.out.println("\nSurat Ditemukan :
");

            System.out.println("ID Surat : " +
suratDitemukan.idSurat);

            System.out.println("Nama Mahasiswa : " +
suratDitemukan.namaMahasiswa);

            System.out.println("Kelas : " +
suratDitemukan.kelas);

            System.out.println("Jenis Perizinan : "
+ suratDitemukan.jenisIzin);

            System.out.println("Lama Waktu Izin : "
+ suratDitemukan.durasi);

        } else if (posisi >= 0 && posisi <
srtTerverif.length && srtTerverif[posisi] != null) {
            Surat05 suratDitemukan =
srtTerverif[posisi];

            System.out.println("\nSurat Ditemukan :
");

            System.out.println("ID Surat : " +
suratDitemukan.idSurat);

            System.out.println("Nama Mahasiswa : " +
suratDitemukan.namaMahasiswa);

            System.out.println("Kelas : " +
suratDitemukan.kelas);

            System.out.println("Jenis Perizinan : "
+ suratDitemukan.jenisIzin);

            System.out.println("Lama Waktu Izin : "
+ suratDitemukan.durasi);

        }
    } else {
        System.out.println("\nSurat Mahasiswa " +
namaCari + " tidak ditemukan dalam daftar surat.");
    }
}

```



```

        break;

        case 5:
            System.out.println("\nTerima kasih, program
selesai.");
            break;

        default:
            System.out.println("\nPilihan tidak valid,
silakan coba lagi.");
    }
    } while (menu != 5);
}
}

```

Output

```

===== MENU =====
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar Program
Pilih : 1

ID Surat : 101
Nama Mahasiswa : Athaulla Hafizh
Kelas : 1E
Jenis perizinan (I/S) : I
Lama waktu izin : 2
Surat Athaulla Hafizh berhasil dikumpulkan

===== MENU =====
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar Program
Pilih : 1

ID Surat : 102
Nama Mahasiswa : Hafizh Athaulla
Kelas : 1E
Jenis perizinan (I/S) : S
Lama waktu izin : 2

```

```
Surat Hafizh Athaulla berhasil dikumpulkan

===== MENU =====
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar Program
Pilih : 3

Surat terakhir dikumpulkan oleh : Hafizh Athaulla

===== MENU =====
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar Program
Pilih : 2

Memverifikasi surat dari Hafizh Athaulla
Surat dari mahasiswa Hafizh Athaulla sudah di verifikasi.

===== MENU =====
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar Program
```

```
Pilih : 2

Memverifikasi surat dari Athaulla Hafizh
Surat dari mahasiswa Athaulla Hafizh sudah di verifikasi.

===== MENU =====
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar Program
Pilih : 4

Masukkan nama mahasiswa yang ingin dicari : Athaulla Hafizh

Surat Ditemukan :
ID Surat : 101
Nama Mahasiswa : Athaulla Hafizh
Kelas : 1E
Jenis Perizinan : I
Lama Waktu Izin : 2
```

```
===== MENU =====
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar Program
Pilih : 5

Terima kasih, program selesai.
```