POLITEKNIK NEGERI MALANG

TEKNOLOGI INFORMASI

TEKNIK INFORMATIKA



#### Nama : Muhammad Nuril Huda

#### Kelas : TI-1A

#### No : 19

Mata Kuliah : Algoritma dan Struktur Data

## 2.1 Percobaan 1: Mahasiswa Mengumpulkan Tugas

##### 2.1.1 Kode Program

Class Mahasiswa 19

public class Mahasiswa19 {

    String nim;

    String nama;

    String kelas;

    int nilai;

    Mahasiswa19(){

    }

    Mahasiswa19(String nama, String nim, String kelas){

        this.nim = nim;

        this.nama = nama;

        this.kelas = kelas;

        nilai = -1;

    }

    void tugasDinilai(int nilai){

        this.nilai = nilai;

    }

}

Class StackTugasMahasiswa19

public class StackTugasMahasiswa19 {

    Mahasiswa19 [] stack;

    int top;

    int size;

    public StackTugasMahasiswa19 (int size){

        this.size = size;

        stack = new Mahasiswa19[size];

        top = -1;

    }

    public boolean isFull(){

        if(top == size - 1){

            return true;

        } else {

            return false;

        }

    }

public boolean isEmpety(){

        if(top == -1){

            return true;

        } else {

            return false;

        }

    }

    public void push(Mahasiswa19 mhs){

        if(!isFull()){

            top++;

            stack[top] = mhs;

        } else {

            System.out.println("Stack penuh! Tidak bisa menambahkan tugas lagi.");

        }

    }

    public Mahasiswa19 pop(){

        if (!isEmpety()){

            Mahasiswa19 m = stack[top];

            top--;

            return m;

        } else {

            System.out.println("Stack kosong! Tidak ada tugas untuk dinilai.");

            return null;

        }

    }

    public Mahasiswa19 peek(){

        if (!isEmpety()){

            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("");

    }

}

Class MahasiswaDemo19

import java.util.Scanner;

import java.util.Stack;

public class MahasiswaDemo19 {

    public static void main(String[] args) {

        StackTugasMahasiswa19 stack = new StackTugasMahasiswa19(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();

                    Mahasiswa19 mhs = new Mahasiswa19(nama, nim, kelas);

                    stack.push(mhs);

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

                    break;

case 2:

                    Mahasiswa19 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();

                        scan.nextLine();

                        dinilai.tugasDinilai(nilai);

                        System.out.printf("Nilai Tugas %s adalah %d\n", dinilai.nama, nilai);

                    }

                    break;

                case 3:

                    Mahasiswa19 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;

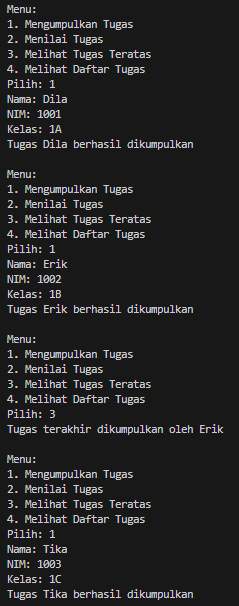
                default:

                    System.out.println("Pilihan tidak valid.");

            }

        } while (pilih >= 1 && pilih <= 4);

##### 2.1.2 Hasil Kode Program



##### 2.1.3 Pertanyaan

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

* Bagian yang perlu diperbaiki adalah method print pada class StackTugasMahasiswa19

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("");

    }

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

* 5

 StackTugasMahasiswa19 stack = new StackTugasMahasiswa19(5);

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

* Karena array stack memiliki kapasitas ukuran tetap dan tidak bisa menampung lebuh banyak dari kapasitasnya, sehingga kondisi !isFull berfungsing untuk memastikan hanya menambahkan data ketika masih ada ruang kosong didalam stack
* Kalau kondisi if-else dihapus kemudian ketika push() dipanggil pada saaat stack sudah penuh maka program akan eror

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

case 4:

      Mahasiswa19 see = stack.peek2();

      if (see != null) {

      System.out.println("Tugas pertama dikumpulkan oleh " + see.nama);

      }

      break;

public Mahasiswa19 peek2(){

        if (!isEmpety()){

            return stack [0];

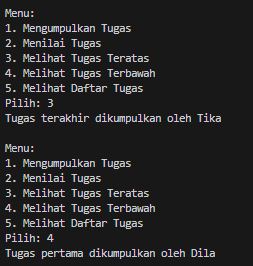
        } else {

            System.out.println("Stack kosong! Tidak ada tugas yang dikumpulkan.");

            return null;

        }

    }



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

public void banyakTugas (){

        if (!isEmpety()){

            int banyaktugas = top+1;

            System.out.println("Jumlah tugas yang dikumpulkan: "+banyaktugas);

        } else {

            System.out.println("Stack kosong! Tidak ada tugas yang dikumpulkan.");

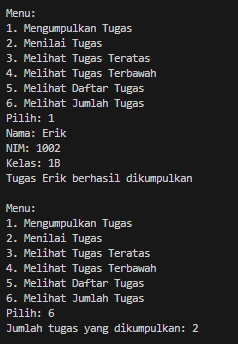
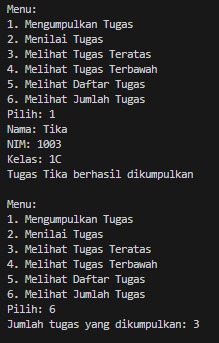
        }

    }

case 6:

stack.banyakTugas();

break;



## 2.2 Percobaan 2: Konversi Nilai Tugas ke Biner

##### 2.2.1 Kode Program

Class StackKonversi19

public class StackKonversi19 {

    int [] tumpukanBiner;

    int size;

    int top;

    public StackKonversi19 (){

        this.size = 32;

        tumpukanBiner = new int[size];

        top = -1;

    }

    public boolean isEmpety(){

        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 (isEmpety()){

            System.out.println("Stack Kosong");

            return -1;

        } else {

            int data = tumpukanBiner[top];

            top--;

            return data;

        }

    }

}

Class MahasiswaDemo19

 case 2:

Mahasiswa19 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();

scan.nextLine();

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;

Class StackTugasMahasiswa19

public String konversiDesimalKeBiner (int nilai){

        StackKonversi19 stack = new StackKonversi19();

        while (nilai >0){

            int sisa = nilai %2;

            stack.push (sisa);

            nilai = nilai / 2;

        }

        String biner = new String();

        while (!stack.isEmpety()){

            biner += stack.pop();

        }

        return biner;

    }

##### 2.2.2 Hasil Kode Program

##### 2.2.3 Pertanyaan

1. Jelaskan alur kerja dari method konversiDesimalKeBiner!

* Membuat stack kosong

Membuat objek StackKonversi19 untuk menampung sisa pembagian.

* Mengisi stack dengan sisa pembagian
* Selama nilai > 0, lakukan:

Hitung sisa = nilai % 2.

Masukkan sisa ke dalam stack dengan push(sisa).

Bagi nilai dengan 2 (nilai = nilai / 2).

* Membuat string kosong untuk hasil

Inisialisasi variabel biner sebagai string kosong.

* Mengeluarkan isi stack
* Selama stack tidak kosong:

Ambil elemen paling atas stack dengan pop().

Tambahkan hasil pop() ke string biner.

* Mengembalikan hasil

Kembalikan string biner yang berisi bilangan dalam bentuk biner.

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

* Jika hanya mengganti nilai menjadi kode tanpa mengubah hal lain, program akan eror. Karena kode tidak didefinisikan didalam method komversiDesimalKeBiner

## 2.4 Latihan Praktikum

##### 2.4.1 Kode Program

Class Surat19

public class Surat19 {

    String idSurat;

    String namaMahasiswa;

    String kelas;

    char jenisIzin;

    int durasi;

    Surat19(){

    }

    Surat19(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;

    }

}

Class StackSurat19

public class StackSurat19 {

    Surat19[] stack;

    int size;

    int top;

    public StackSurat19(int size) {

        this.size = size;

        stack = new Surat19[size];

        top = -1;

    }

    public boolean isFull() {

        return top == size - 1;

    }

    public boolean isEmpety() {

        return top == -1;

    }

public void push(Surat19 srt) {

        if (!isFull()) {

            top++;

            stack[top] = srt;

        } else {

            System.out.println("Stack penuh! Tidak bisa menambahkan surat lagi.");

        }

    }

    public Surat19 pop() {

        if (!isEmpety()) {

            Surat19 s = stack[top];

            top--;

            return s;

        } else {

            System.out.println("Stack kosong! Tidak ada surat.");

            return null;

        }

    }

    public Surat19 peek() {

        if (!isEmpety()) {

            return stack[top];

        } else {

            System.out.println("Stack kosong! Tidak ada surat di atas.");

            return null;

        }

    }

public void cariSuratBerdasarkanNama(String nama) {

        boolean ditemukan = false;

        for (int i = top; i >= 0; i--) {

            if (stack[i].namaMahasiswa.equalsIgnoreCase(nama)) {

                System.out.println("Surat ditemukan:");

                System.out.println("ID: " + stack[i].idSurat);

                System.out.println("Nama: " + stack[i].namaMahasiswa);

                System.out.println("Kelas: " + stack[i].kelas);

                System.out.println("Jenis Izin: " + stack[i].jenisIzin);

                System.out.println("Durasi: " + stack[i].durasi + " hari");

                ditemukan = true;

                break;

            }

        }

        if (!ditemukan) {

            System.out.println("Surat tidak ditemukan.");

        }

    }

}

Class SuratDemo19

import java.util.Scanner;

public class SuratDemo19 {

    public static void main(String[] args) {

        StackSurat19 stack = new StackSurat19(5);

        Scanner scan = new Scanner(System.in);

        int pilih;

do {

            System.out.println("\nMenu:");

            System.out.println("1. Terima Surat Izin");

            System.out.println("2. Proses Surat Izin");

            System.out.println("3. Melihat Surat Izin Teratas");

            System.out.println("4. Cari Surat");

            System.out.print("Pilih: ");

            pilih = scan.nextInt();

            scan.nextLine(); // Membersihkan newline

            switch (pilih) {

                case 1:

                    System.out.print("ID Surat: ");

                    String idSurat = scan.nextLine();

                    System.out.print("Nama Mahasiswa: ");

                    String namaMahasiswa = scan.nextLine();

                    System.out.print("Kelas: ");

                    String kelas = scan.nextLine();

                    System.out.print("Jenis Surat (S/I): ");

                    char jenisizin = scan.nextLine().charAt(0); // ambil satu karakter

                    System.out.print("Durasi: ");

                    int durasi = scan.nextInt();

                    scan.nextLine();

                    Surat19 mhs = new Surat19(idSurat, namaMahasiswa, kelas, jenisizin, durasi);

                    stack.push(mhs);

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

                    break;

case 2:

                    Surat19 diproses = stack.pop();

                    if (diproses != null) {

                        System.out.println("Surat dari " + diproses.namaMahasiswa + " telah diproses.");

                    }

                    break;

                case 3:

                    Surat19 teratas = stack.peek();

                    if (teratas != null) {

                        System.out.println("Surat Teratas:");

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

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

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

                        System.out.println("Jenis Izin: " + teratas.jenisIzin);

                        System.out.println("Durasi: " + teratas.durasi + " hari");

                    }

                    break;

                case 4:

                    System.out.print("Masukkan nama mahasiswa yang dicari: ");

                    String cariNama = scan.nextLine();

                    stack.cariSuratBerdasarkanNama(cariNama);

                    break;

                default:

                    System.out.println("Pilihan tidak valid.");

                    break;

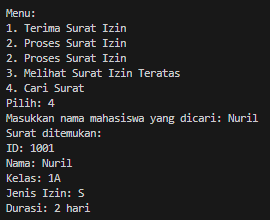
            }

        } while (pilih >= 1 && pilih <= 4);

    }

}

##### 2.4.2 Hasil Kode Program



## Link Github: <https://github.com/nurilhuda05/Algoritma-dan-Struktur-Data.git>