

Jobsheet 9

Algoritma dan Struktur Data (Stack)

Hernanda Rizka Utami / 244107060075 / SIB – 1B

Percobaan 1

Commit dan Push ke Github

Name	Last commit message	Last commit date
..		
Mahasiswa12.class	Percobaan 1 js9	2 minutes ago
Mahasiswa12.java	Percobaan 1 js9	2 minutes ago
MahasiswaDemo12.class	Percobaan 1 js9	2 minutes ago
MahasiswaDemo12.java	Percobaan 1 js9	2 minutes ago
StackTugasMahasiswa12.class	Percobaan 1 js9	2 minutes ago
StackTugasMahasiswa12.java	Percobaan 1 js9	2 minutes ago

Compile dan run

```
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
```

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:

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

Menjadi:

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

Output:







	<pre> 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 Dilla 1001 1A </pre>
2. Berapa banyak data tugas mahasiswa yang dapat ditampung di dalam Stack? Tunjukkan potongan kode programnya!	<p>Ada 5 data mahasiswa yang dapat ditampung pada stack.</p> <pre> StackTugasMahasiswa12 stack = new StackTugasMahasiswa12(5); </pre>
3. Mengapa perlu pengecekan kondisi !isFull() pada method push ? Kalau kondisi if-else tersebut dihapus, apa dampaknya?	<p>Karena untuk proses mengecek atau untuk memastikan bahwa masih ada ruang di dalam stack sebelum menambahkan data baru.</p> <p>Kondisi yang terjadi jika if-else dihapus adalah program akan crash dan error karena ketika menghapus kondisi pengecekan dan program langsung melakukan proses</p> <pre> top++; stack[top] = mhs; </pre> <p>maka ketika <code>top == size - 1</code> atau stack dalam kondisi penuh, <code>top++</code> akan membuat <code>top == size</code>, dan program akan mencoba mengakses <code>stack[size]</code> padahal index maksimal adalah <code>size - 1</code>.</p>
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!	<p>Modifikasi class StackTugasMahasiswa</p> <pre> public Mahasiswa12 bottom(){ if(!isEmpty()){ return stack[0]; }else{ System.out.println("Stack kosong! Tidak ada tugas yang dikumpulkan"); return null; } } </pre> <p>Modifikasi class MahasiswaDemo</p>

	<pre> case 5: Mahasiswa12 bawah = stack.bottom(); if(bawah != null){ System.out.println("Tugas pertama dikumpulkan oleh " + bawah.nama); } break; </pre> <p>Output</p> <pre> Menu: 1. Mengumpulkan Tugas 2. Menilai Tugas 3. Melihat Tugas Teratas 4. Melihat Daftar Tugas 5. Melihat Tugas Terbawah Pilih: 4 Daftar semua tugas: Nama Nim kelas Tika 1003 1C Erik 1002 1B Dilla 1001 1A </pre> <pre> Menu: 1. Mengumpulkan Tugas 2. Menilai Tugas 3. Melihat Tugas Teratas 4. Melihat Daftar Tugas 5. Melihat Tugas Terbawah Pilih: 5 Tugas pertama dikumpulkan oleh Dilla </pre>
<p>5. Tambahkan method untuk dapat menghitung berapa banyak tugas yang sudah dikumpulkan saat ini, serta tambahkan operasi menunya!</p>	<p>Modifikasi class StackTugasMahasiswa</p> <pre> public int jumlahTugas(){ return top + 1; } </pre> <p>Modifikasi class MahasiswaDemo</p> <pre> case 6: int jumlah = stack.jumlahTugas(); System.out.printf("Jumlah tugas yang dikumpulkan: %d\n", jumlah); break; </pre>

	<p>Output</p> <div> <pre> Menu: 1. Mengumpulkan Tugas 2. Menilai Tugas 3. Melihat Tugas Teratas 4. Melihat Daftar Tugas 5. Melihat Tugas Terbawah 6. Melihat Jumlah Tugas yang telah dikumpulkan Pilih: 4 Daftar semua tugas: Nama Nim kelas Tika 1003 1C Dilla 1001 1B </pre> </div> <div> <pre> Menu: 1. Mengumpulkan Tugas 2. Menilai Tugas 3. Melihat Tugas Teratas 4. Melihat Daftar Tugas 5. Melihat Tugas Terbawah 6. Melihat Jumlah Tugas yang telah dikumpulkan Pilih: 6 Jumlah tugas yang dikumpulkan: 2 </pre> </div>
<p>6. Commit dan push kode program ke Github</p>	<div> <div> <div></div> <div>MahasiswaDemo12.class</div> <div>P1 pertanyaan 5</div> <div>now</div> </div> <div> <div></div> <div>MahasiswaDemo12.java</div> <div>P1 pertanyaan 5</div> <div>now</div> </div> <div> <div></div> <div>StackTugasMahasiswa12.class</div> <div>P1 pertanyaan 5</div> <div>now</div> </div> <div> <div></div> <div>StackTugasMahasiswa12.java</div> <div>P1 pertanyaan 5</div> <div>now</div> </div> </div>

Percobaan 2

Commit dan Push ke Github

 MahasiswaDemo12.class	Percobaan 2	1 minute ago
 MahasiswaDemo12.java	Percobaan 2	1 minute ago
 StackKonversi12.class	Percobaan 2	1 minute ago
 StackKonversi12.java	Percobaan 2	1 minute ago
 StackTugasMahasiswa12.class	Percobaan 2	1 minute ago
 StackTugasMahasiswa12.java	Percobaan 2	1 minute ago

Compile dan run

Menu:

1. Mengumpulkan Tugas
2. Menilai Tugas
3. Melihat Tugas Teratas
4. Melihat Daftar Tugas
5. Melihat Tugas Terbawah
6. Melihat Jumlah Tugas yang telah dikumpulkan

Pilih: 2

Menilai tugas dari Tika

Masukkan nilai (0-100): 78

Nilai Tugas Tika adalah 78

Nilai Biner Tugas: 1001110

Pertanyaan

1. Jelaskan alur kerja dari method **konversiDesimalKeBiner!**

- ```
StackKonversi12 stack = new StackKonversi12();
```

Membuat objek stack untuk menyimpan sisa pembagian 2 dari nilai decimal.

```
while(nilai > 0){
 int sisa = nilai % 2;
 stack.push(sisa);
 nilai = nilai / 2;
}
```
- Perulangan utama dari konversi decimal ke biner, selama nilai decimal masih lebih dari 0 akan melakukan pembagian dengan 2.

Int sisa ini untuk mengambil sisa pembagian 2 dari nilai yang akan menjadi digit biner.

Sisa pembagian akan dimasukkan ke dalam stack.

Lalu proses nilai dibagi 2 untuk loop selanjutnya.

```
String biner = new String();
while (!stack.isEmpty()){
 biner += stack.pop();
}
return biner;
```
- Untuk membuat string kosong yang akan menyimpan hasil biner akhir.

Perulangan disini akan mengambil semua isi stack yaitu digit-digit biner.

Biner+= untuk mengambil elemen paling atas dari stack dan akan menggabungkan ke string biner yang akan membentuk urutan digit biner secara lengkap.

Lalu mengembalikan hasil konversi decimal ke biner dalam bentuk string.

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

Jika perulangan dalam **konversiDesimalKeBiner** diganti menjadi perulangan **while(nilai !=0)** hasilnya akan tetap sama dan tetap bisa dijalankan programnya. Karena selama nilai yang diinput positif maka nilai > 0 dan nilai != 0 akan sama masuk ke dalam perulangan, tetapi jika nilai adalah angka negatif maka nilai !=0 akan masuk ke perulangan dan bisa menyebabkan infinite loop karena nilai negatif dibagi 2 terus tidak akan pernah jadi nol.

### Class Surat12

```
public class Surat12 {
 String idSurat, namaMahasiswa, kelas;
 char jenisIzin;
 int durasi;

 public Surat12(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 StackSurat12

```
public class StackSurat12 {
 Surat12[] stack;
 int top, size;

 public StackSurat12(int size) {
 this.size = size;
 stack = new Surat12[size];
 top = -1;
 }

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

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

 public void push(Surat12 surat) {
 if (!isFull()) {
 stack[++top] = surat;
 } else {
 System.out.println("Tumpukan surat penuh!");
 }
 }

 public Surat12 pop() {
 if (!isEmpty()) {
 return stack[top--];
 } else {
 return null;
 }
 }
}
```

```
 System.out.println("Tidak ada surat untuk diproses.");
 return null;
 }
}

public Surat12 peek() {
 if (!isEmpty()) {
 return stack[top];
 } else {
 System.out.println("Tidak ada surat dalam tumpukan.");
 return null;
 }
}

public boolean cariSurat(String nama) {
 for (int i = 0; i <= top; i++) {
 if (stack[i].namaMahasiswa.equalsIgnoreCase(nama)) {
 return true;
 }
 }
 return false;
}
}
```



```
 int durasi = scan.nextInt();
 scan.nextLine();
 stack.push(new Surat12(id, nama, kelas, jenis, durasi));
 System.out.println("Surat berhasil diterima.");
 break;

 case 2:
 Surat12 diproses = stack.pop();
 if (diproses != null) {
 System.out.println("Surat diproses dari: " + diproses.namaMahasiswa);
 }
 break;

 case 3:
 Surat12 terakhir = stack.peek();
 if (terakhir != null) {
 System.out.println("Surat Terakhir:");
 System.out.println("Nama: " + terakhir.namaMahasiswa);
 System.out.println("Jenis Izin: " + terakhir.jenisIzin);
 System.out.println("Durasi: " + terakhir.durasi + " hari");
 }
 break;

 case 4:
 System.out.print("Masukkan nama mahasiswa yang dicari: ");
 String cariNama = scan.nextLine();
 if (stack.cariSurat(cariNama)) {
 System.out.println("Surat dari " + cariNama + " ditemukan.");
 } else {
 System.out.println("Surat dari " + cariNama + " tidak ditemukan.");
 }
 break;
```

```

 case 5:
 System.out.println("Keluar dari program.");
 break;

 default:
 System.out.println("Pilihan tidak valid.");
 }
} while (pilih != 5);
}

```

## Output

### Menu 1

```

--- Menu Surat Izin Mahasiswa ---
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar
Pilih: 1
ID Surat: 1001
Nama Mahasiswa: Hernanda Rizka Utami
Kelas: 1B
Jenis Izin (S/I): I
Durasi (hari): 1
Surat berhasil diterima.

```

```

--- Menu Surat Izin Mahasiswa ---
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar
Pilih: 1
ID Surat: 1001
Nama Mahasiswa: Gempita Fitri Nurdini
Kelas: 1C
Jenis Izin (S/I): S
Durasi (hari): 2
Surat berhasil diterima.

```

```

--- Menu Surat Izin Mahasiswa ---
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar
Pilih: 1
ID Surat: 1003
Nama Mahasiswa: Tania Eka Putri
Kelas: 1B
Jenis Izin (S/I): I
Durasi (hari): 1
Surat berhasil diterima.

```

## Menu 2

```
--- Menu Surat Izin Mahasiswa ---
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar
Pilih: 2
Surat diproses dari: Tania Eka Putri
```

## Menu 3

```
--- Menu Surat Izin Mahasiswa ---
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar
Pilih: 3
Surat Terakhir:
Nama: Gempita Fitri Nurdini
Jenis Izin: S
Durasi: 2 hari
```

## Menu 4

```
--- Menu Surat Izin Mahasiswa ---
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar
Pilih: 4
Masukkan nama mahasiswa yang dicari: Hernanda
Surat dari Hernanda tidak ditemukan.
```

```
--- Menu Surat Izin Mahasiswa ---
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar
Pilih: 4
Masukkan nama mahasiswa yang dicari: Hernanda Rizka Utami
Surat dari Hernanda Rizka Utami ditemukan.
```

## Menu 5

```
--- Menu Surat Izin Mahasiswa ---
1. Terima Surat Izin
2. Proses Surat Izin
3. Lihat Surat Izin Terakhir
4. Cari Surat
5. Keluar
Pilih: 5
Keluar dari program.
```