

JOBSHEET 6

Pemilihan 2

Nama: Rizqi Bagus Andrean

Absen: 25

Kelas: TI 1-D

1. Tujuan

- Mahasiswa memahami tentang operator logika
- Mahasiswa mampu menyelesaikan permasalahan dengan menggunakan sintaks pemilihan bersarang
- Mahasiswa mampu membuat sebuah program Java yang memanfaatkan sintaks pemilihan bersarang

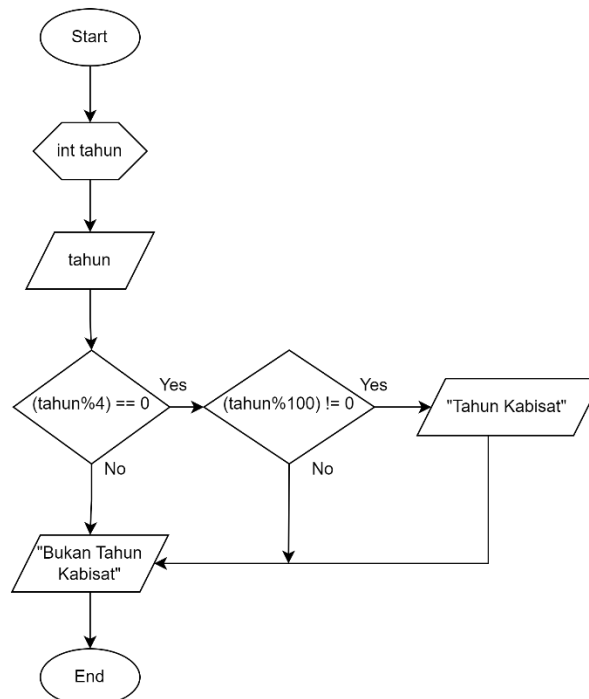
2. Praktikum

2.1 Percobaan 1

Waktu Percobaan 50 Menit

1. Menentukan tahun kabisat atau bukan, dengan ketentuan tahun kabisat adalah tahun kelipatan 4 dan bukan kelipatan 100.

Flowchart berikut adalah algoritma dari penentuan tahun kabisat



2. Selanjutnya kita akan membuat programnya. Buka text editor kemudian simpan dengan nama **Pemilihan2Percobaan1NoAbsen.java**
3. Buatlah struktur dasar java (membuat class dan program main).
4. Tambahkan import library Scanner.
5. Deklarasikan Scanner : beri nama Scanner dengan identitas Absen. Format input **Absen**
6. Tambahkan kode untuk menerima inputan dari keyboard untuk tahun.
7. Buatlah struktur kondisi seperti dibawah ini :

```
if ((tahun % 4) == 0) {
    if ((tahun % 100) != 0)
        System.out.println(x:"Tahun Kabisat");
    } else
        System.out.println(x:"Bukan Tahun Kabisat");
```

8. Jalankan program tersebut, maka outputnya adalah sebagai berikut :

```
Masukkan Tahun : 2004
Tahun Kabisat
```

9. Push dan commit hasil praktikum anda ke repository

Pertanyaan

1. Bagaimana outputnya ketika diberikan input tahun 2100 ? Jelaskan! Bagaimana agar output sesuai dengan ketentuan (Tahun 2100 bukan tahun kabisat)
2. Modifikasi program sesuai jawaban no 1 !

```
Pemilihan2Percobaan125.java > [?] Pemilihan2Percobaan125 > [?] main(String[])
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          System.out.print(s:"Masukkan tahun: ");
8          int tahun = sc.nextInt();
9
10         if (tahun % 4 == 0){
11             if (tahun % 100 != 0){
12                 System.out.println(tahun + " adalah tahun kabisat");
13             }else{
14                 System.out.println(tahun + " adalah bukan tahun kabisat");
15             }
16         } else {
17             System.out.println(tahun + " adalah bukan tahun kabisat");
18         }
19     }
20 }
21 }
```

3. Push dan commit hasil modifikasi anda ke repository !
4. Tahun 2000 adalah kelipatan 4 dan kelipatan 100, tetapi tahun 2000 merupakan tahun kabisat, ketentuan tambahan (pengecualian) adalah ketika tahun kelipatan 100 **dan juga** kelipatan 400 maka tahun tersebut merupakan tahun kabisat. Modifikasi program untuk menyesuaikan ketentuan tersebut ! (selesaikan tanpa menggunakan operator logika)

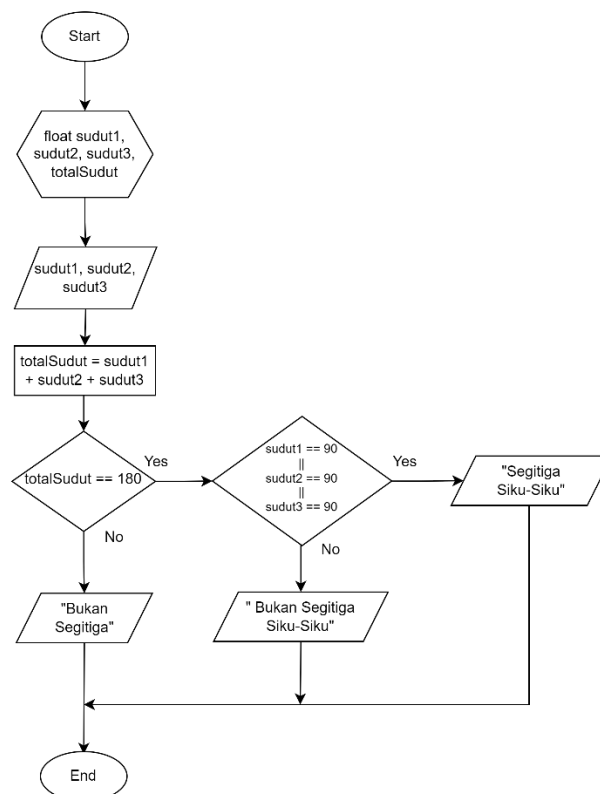
```
public static void main(String[] args) {  
    Scanner sc = new Scanner(System.in);  
  
    System.out.print("Masukkan tahun: ");  
    int tahun = sc.nextInt();  
  
    if (tahun % 4 == 0){  
        if (tahun % 100 != 0){  
            System.out.println(tahun + " adalah tahun kabisat");  
        }else{  
            if (tahun % 400 == 0){  
                System.out.println(tahun + " adalah tahun kabisat");  
            }else{  
                System.out.println(tahun + " adalah bukan tahun kabisat");  
            }  
        }  
    } else {  
        System.out.println(tahun + " adalah bukan tahun kabisat");  
    }  
}
```

5. Push dan commit hasil modifikasi anda ke repository !

2.2 Percobaan 2

Waktu Percobaan 50 Menit

1. Menentukan jenis bidang datar segitiga dari masukan ketiga sudutnya.
2. Perhatikan flowchart berikut ini!



3. Selanjutnya kita akan membuat programnya. Buka text editor kemudian simpan dengan nama **Pemilihan2Percobaan2NoAbsen.java**
4. Buatlah struktur dasar java (membuat class dan program main).
5. Tambahkan import library Scanner.

6. Deklarasikan Scanner : beri nama Scanner dengan identitas Absen. Format input **Absen**
7. Tuliskan perintah untuk memasukkan inputan dari keyboard, untuk 3 sudut segitiga.
8. Tambahkan kode program berikut untuk menghitung jumlah ketidak sudut segitiga

```
totalSudut = sudut1 + sudut2 + sudut3;
```

9. Buatlah struktur kondisi seperti dibawah ini :

```
if (totalSudut == 180) {  
    if ((sudut1 == 90) || (sudut2 == 90) || (sudut3 == 90))  
        System.out.println(x:"Segitiga tersebut adalah segitiga siku-siku");  
    else  
        System.out.println(x:"Segitiga tersebut adalah bukan segitiga siku-siku");  
} else  
    System.out.println(x:"Bukan Segitiga");
```

10. Jalankan program tersebut, maka outputnya adalah sebagai berikut :

```
Masukkan Sudut 1 : 90  
Masukkan Sudut 2 : 30  
Masukkan Sudut 3 : 60  
Segitiga tersebut adalah segitiga siku-siku
```

11. Push dan commit hasil praktikum anda ke repository

Pertanyaan

1. Jenis bangun datar segitiga selain segitiga siku-siku terdapat segitiga sama sisi, segitiga sama kaki dan segitiga sembarang. Modifikasi program tersebut untuk dapat mengeluarkan output jenis segitiga yang lain (selain segitiga siku-siku). Silahkan menggunakan operator logika untuk menghubungkan multi kondisi.

```
import java.util.Scanner;

public class Pemilihan2Percobaan225 {
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan Sudut 1: ");
        int sudut1 = sc.nextInt();
        System.out.print("Masukkan Sudut 2: ");
        int sudut2 = sc.nextInt();
        System.out.print("Masukkan Sudut 3: ");
        int sudut3 = sc.nextInt();

        int totalSudut = sudut1 + sudut2 + sudut3;

        if (totalSudut == 180){

            if (sudut1 == sudut2 && sudut2 == sudut3){
                System.out.println("Segitiga Sama Sisi");
            }else if ((sudut1 == sudut2 || sudut2 == sudut3 || sudut1 == sudut3) && sudut1 != 90 && sudut2 !=
90 && sudut3 != 90){
                System.out.println("Segitiga Sama Kaki");
            }else if (sudut1 == 90 || sudut2 == 90 || sudut3 == 90){
                System.out.println("Segitiga Siku-Siku");
            }else{
                System.out.println("Segitiga Sembarang");
            }
        }else{
            System.out.println("Bukan Segitiga");
        }
    }
}
```

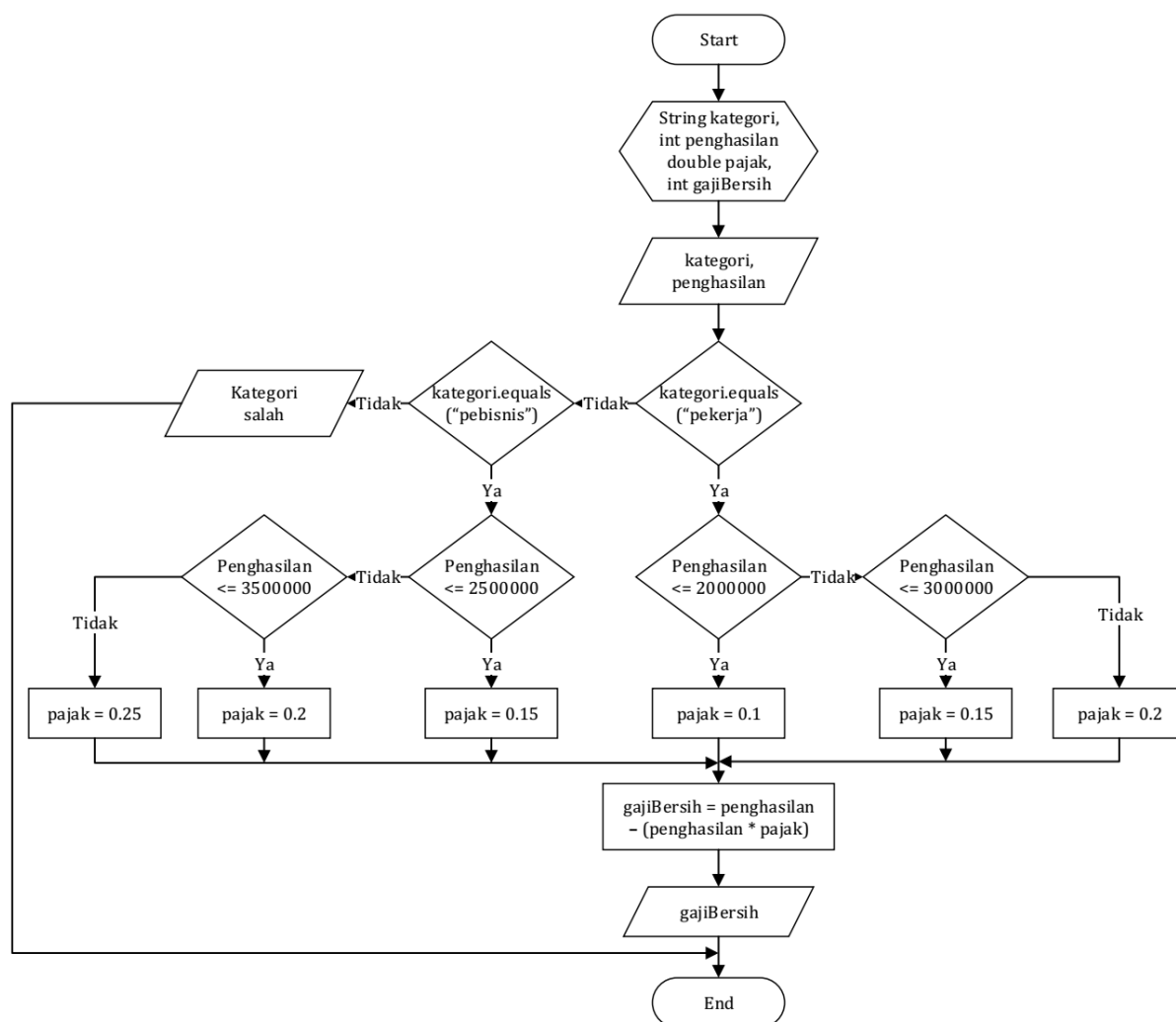
2. Push dan commit hasil modifikasi anda ke repository !

2.3 Percobaan 3

Waktu Percobaan 40 Menit

Studi Kasus

1. Perhatikan flowchart di bawah ini, flowchart tersebut digunakan untuk menghitung gaji bersih seseorang setelah dipotong pajak sesuai dengan kategorinya (pekerja dan pebisnis) dan besarnya penghasilan.



- Kita buat program berdasarkan algoritma pada flowchart tersebut. Buka text editor kemudian simpan dengan nama **Pemilihan2Percobaan3NoAbsen.java**
- Buatlah struktur dasar java (membuat class dan program main).
- Tambahkan import library Scanner.
- Deklarasikan Scanner : beri nama Scanner dengan identitas Absen. Format input **Absen**
- Deklarasikan variabel **kategori**, **penghasilan**, **gajiBersih**, dan **pajak**;

```
String kategori;
int penghasilan, gajiBersih;
double pajak = 0;
```

- Tambahkan kode berikut ini untuk menerima input dari keyboard

```
System.out.print("Masukkan Kategori : ");
kategori = noAbsen.nextLine();
System.out.print("Masukkan Besarnya Penghasilan : ");
penghasilan = noAbsen.nextInt();
```

- Buatlah struktur pengecekan kondisi bersarang. Pengecekan pertama digunakan untuk mengecek kategori (pekerja atau pebisnis). Selanjutnya dilakukan pengecekan kedua untuk

menentukan besarnya pajak berdasarkan penghasilan yang telah dimasukkan. Kemudian tambahkan kode program untuk menghitung gaji bersih yang diterima setelah dipotong pajak

```
if (kategori.equalsIgnoreCase(anotherString:"pekerja")) {
    if (penghasilan <= 2000000)
        pajak = 0.1;
    else if (penghasilan <= 3000000)
        pajak = 0.15;
    else
        pajak = 0.2;
    gajiBersih = (int) (penghasilan - (pajak * penghasilan));
    System.out.print("Penghasil Bersih : " + gajiBersih);
} else if (kategori.equalsIgnoreCase(anotherString:"pebisnis")) {
    if (penghasilan <= 2500000)
        pajak = 0.15;
    else if (penghasilan <= 3500000)
        pajak = 0.2;
    else
        pajak = 0.25;
    gajiBersih = (int) (penghasilan - (pajak * penghasilan));
    System.out.print("Penghasil Bersih : " + gajiBersih);
} else
    System.out.println(x:"Masukan Kategori Salah ");
```

9. Jalankan program tersebut. Amati apa yang terjadi!

Pertanyaan

1. Jelaskan fungsi dari **(int)** pada sintaks `gajiBersih = (int) (penghasilan - (penghasilan * pajak));`
Digunakan untuk mengubah tipe data gaji bersih ke interger
2. Jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Apa kegunaan dari **equalsIgnoreCase**?
EqualsIgnore case akan mencocokkan apakah kata PEBISNIS cocok dengan pebisnis, tapi tidak memperhatikan besar kecil hurufnya (tidak case sensitive)
3. Ubah **equalsIgnoreCase** menjadi **equals**, kemudian jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Mengapa hasilnya demikian? Apa kegunaan dari **equals**?
Karena equals itu memcocokkan dengan memperhatikan besar kecil nya huruf (case sensitive) sehingga PEBISNIS dan pebisnis itu tidak sama

3. Tugas

Waktu Percobaan : 160 Menit

Buatlah kode program berdasarkan flowchart yang telah dibuat pada Tugas pertemuan 6 Matakuliah Dasar Pemrograman terkait Project! Push dan commit hasil kode program anda ke repository project Anda!

Catatan : tugas hanya boleh menerapkan materi dari pertemuan 1 hingga pertemuan 6.

