

IF184982 Pengantar Logika dan Pemrograman

Kuis 1

Tanggal Mulai: 06 Oktober 2018
Pengumpulan Terakhir: 13 Oktober 2018 jam 23:59 WIB.
Sifat Ujian: Terbuka
Dikirimkan ke alamat email: MM Irfan Subakti <yifana@gmail.com>
Jenis dan format file: File .java yang sudah di-*compress* dalam 1 file .ZIP
Format nama file: IF184982_PLP_K1_NRP_Nama.ZIP

Kerjakan soal di bawah ini sebaik-baiknya.

1. Buatlah program dengan nama "Ganjil.java" yang mencetak bilangan ganjil dari 1 sampai bilangan yang dimasukkan pengguna (*user*). Gunakan fungsi modulus (yaitu operator "%") untuk mendapatkan nilai sisa dari suatu bilangan.

Contoh hasil:

Masukkan bilangan: 20

```
1
3
5
7
9
11
13
15
17
19
```

Petunjuk:

```
5 % 1 = 0
5 % 2 = 1
5 % 3 = 2
5 % 4 = 1
5 % 5 = 0
```

2. Buatlah program dengan nama "TerkecilTerbesar.java" yang menerima input/masukan bilangan bulat dari pengguna. Bila pengguna memasukkan bilangan negatif, tampilkan bilangan terkecil dan terbesar dari semua bilangan yang dimasukkan pengguna. Tampilkan juga hasil perkalian bilangan terkecil dan terbesar tersebut.

Contoh hasil:

Masukkan bilangan: 10
Masukkan bilangan: 15
Masukkan bilangan: 3
Masukkan bilangan: -7
Bilangan terkecil: 3
Bilangan terbesar: 15
Perkalian bilangan terkecil dan terbesar: 45

Petunjuk:

Buatlah variabel `int terkecil` untuk menyimpan nilai input/masukan terkecil dari pengguna. Buatlah juga variabel `int terbesar` untuk menyimpan nilai input/masukan terbesar dari pengguna. Gunakan *function/method* `nextInt()` untuk mengambil masukan/input (*scanning*) bilangan bulat dari pengguna. Beri nilai pertama `terkecil` dari input/masukan pertama dari pengguna. Buat juga variabel `int bilBerikutnya` untuk menyimpan input pengguna berikutnya. Beri nilai pertama `bilBerikutnya` dengan nilai `terkecil`. Beri juga nilai pertama `terbesar` dengan `bilBerikutnya`. Buat perulangan sampai pengguna memasukkan bilangan negatif. Beri nilai variabel `bilBerikutnya` dari input selanjutnya/berikutnya pengguna dari *function/method* `nextInt()`. Selama perulangan evaluasi dan *update* bilangan manakah yang terkecil dan terbesar. Tampilkan bilangan yang terkecil dan terbesar ini beserta hasil perkalian mereka.

3. Buatlah program dengan nama "Prima.java" yang mencetak bilangan prima dari bilangan prima terkecil sampai bilangan yang dimasukkan pengguna.

Contoh hasil:

Masukkan bilangan: 20
2
3
5
7
11
13
17
19

Petunjuk:

Buatlah *function/method* adalah `Prima(int bilangan)` seperti di bawah ini. Gunakan fungsi `Math.sqrt()` untuk mendapatkan nilai akar dari suatu bilangan, juga modulus (yaitu operator "%") untuk mendapatkan nilai sisa dari suatu bilangan. Hasil dari *function* ini adalah `true` → jika prima, dan `false` → bukan prima

```
public static boolean adalahPrima(int bilangan) {
    ...
}
```

Kemudian, cetaklah bilangan prima dari bilangan prima terkecil sampai bilangan yang dimasukkan pengguna, dengan menggunakan *function* `adalahPrima(int bilangan)` di atas.

4. Buatlah program dengan nama “`Bintang.java`” yang menerima masukan bilangan bulat n dari pengguna. Tampilkan karakter `*` (bintang) sebanyak 2 (dua) kali bilangan yang dimasukkan pengguna ($2n$), dengan susunan seperti contoh di bawah ini.

```
Baris ke-1 : * (0 spasi diikuti bintang)
Baris ke-2 : * (spasi diikuti bintang)
Baris ke-3 : * (2 spasi diikuti bintang)
...
Baris ke-n : * (n-1 spasi diikuti bintang)
Baris ke-n+1 : * (n-1 spasi diikuti bintang)
Baris ke-n+2 : * (n-2 spasi diikuti bintang)
Baris ke-n+3 : * (n-3 spasi diikuti bintang)
...
Baris ke-2*n : * (0 spasi diikuti bintang)
```

Contoh hasil:

Masukkan bilangan: 6

```
*
 *
  *
   *
    *
     *
      *
       *
        *
         *
          *
           *
```

Petunjuk:

Gunakan `for ()` dalam `for ()` untuk mendapatkan hasil seperti di atas. `for ()` pertama untuk mengolah baris, `for ()` kedua untuk mengolah kolom.

5. Buatlah program dengan nama "Rotasi.java" yang menerima masukan 2 buah string dari pengguna. Tampilkan hasil dari pengecekan bila dua buah string tersebut adalah merupakan rotasi satu lainnya.

Contoh hasil 1:

Masukkan string ke-1: ALAMI
 Masukkan string ke-2: MIALA
 Gabungan string ke-1 sebanyak 2 kali adalah: ALAMIALAMI
 String ke-2 (MIALA) ada di gabungan string tersebut.
 Kedua string merupakan rotasi satu sama lainnya.

Contoh hasil 2:

Masukkan string ke-1: TEMPE
 Masukkan string ke-2: PELEM
 Gabungan string ke-1 sebanyak 2 kali adalah: TEMPETEMPE
 String ke-2 (PELEM) tidak ada di gabungan string tersebut.
 Kedua string bukan merupakan rotasi satu sama lainnya.

Petunjuk:

Buatlah *function/method* cekRotasi () sebagai berikut:

```
private static boolean cekRotasi(String s1, String s2) {
    return ...
}
```

Di dalam *function* ini cek apakah string pertama sama panjangnya dengan string kedua dan juga apakah gabungan string pertama sebanyak dua kali mengandung string kedua. Gunakan fungsi `length()` dan `contains()` untuk operasi `String` dalam *function* ini.

6. Buatlah program dengan nama "NilaiASCII.java" yang menerima masukan/input sebuah kata dari pengguna. Tampilkan total nilai ASCII dari kata tersebut.

Contoh hasil:

Masukkan kata: AKU
 Total nilai ASCII dari "AKU" adalah 225

Petunjuk:

Nilai ASCII karakter A = 65, B = 66, ..., Z = 90, a = 97, b = 98, ..., z = 122

Nilai ASCII dari suatu karakter adalah nilai integer dari karakter tersebut. *Casting* (penyesuaian format) suatu tipe data tertentu ke tipe data yang diinginkan dalam Java bisa dilakukan dengan format sebagai berikut:

```
char c = 'A';
```

```
int nilaiInteger = (int) c; → casting nilai karakter 'A' (yang mempunyai nilai ASCII = 65) ke nilai integer, sehingga nilaiInteger sekarang bernilai 65.
```

Function/method `System.out.print()` atau `System.out.println()` mencetak suatu string ke layar/screen. Misal: `System.out.println("Milenia")` → mencetak Milenia. Namun bila diinginkan untuk mencetak karakter " itu sendiri, tidak bisa langsung dilakukan seperti ini: `System.out.println("Milenia")` untuk mendapatkan

hasil "Milenia". Karena karakter " itu sendiri sudah digunakan Java sebagai penanda string → string adalah apa pun yang diapit oleh karakter " (contoh: "Arek Milenia"). Untuk menampilkan karakter " itu sendiri, digunakan *escape sequence* \". Di Java dikenal istilah *escape sequence* yaitu karakter yang memiliki arti khusus bagi *compiler*. Contoh *escape sequence* adalah seperti di bawah ini.

- \t Menambahkan tab pada string
- \b Menambahkan backspace pada string
- \n Menambahkan baris baru pada string
- \r Mengarahkan kursor pada awal baris
- \f Menambahkan halaman baru pada string
- \' Menambahkan tanda petik tunggal pada string
- \" Menambahkan tanda petik ganda pada string
- \\ Menambahkan karakter garis miring terbalik – *backslash* (\) pada string

Contoh: `System.out.println("\\Milenia\\")` → output: `\Milenia\`

Kumpulkan semua file .java di atas dalam satu folder. *Compress* dalam 1 file .ZIP, beri nama file IF184982_PLP_K1_NRP_Nama.ZIP. Kirimkan ke email: MM Irfan Subakti <yifana@gmail.com> sebelum tanggal 13 Oktober 2018, jam 23:59. Semoga sukses! 😊