

Tutorial 0
Mencoba Pengumpulan Pekerjaan ke Server Aren
dan *Review* Pemrograman Java

Struktur Data dan Algoritma
IKI10400
Semester Genap 2010/2011

Fakultas Ilmu Komputer
Universitas Indonesia

Batas waktu pengumpulan kode sumber:
Rabu, 16 Februari 2011 pukul 21.00 Waktu Server Aren

Kode sumber yang dinilai hanya yang dikumpulkan melalui Aren.
Kode sumber yang dikumpulkan melalui mekanisme selain itu
akan diabaikan dan dianggap tidak mengumpulkan.

**Peringatan: jangan mengumpulkan pekerjaan beberapa menit
menjelang batas waktu pengumpulan karena ada kemungkinan
pengumpulan gagal dilakukan atau koneksi internet terputus!**

Jika tidak dapat mengumpulkan tutorial sebelum batas waktu
karena suatu atau beberapa hal khusus, mahasiswa yang
bersangkutan harus melakukan langkah-langkah dalam SOP
Perpanjangan Batas Waktu Pengumpulan Pekerjaan.

Tutorial 0A

I/O 1

Nama berkas kode sumber : SDA11100A.java
Batas waktu eksekusi program : 0,3 detik / kasus uji
Batas memori program : 16 MiB / kasus uji

Input/Output (I/O) yang dalam bahasa Indonesianya adalah masukan dan keluaran, merupakan sesuatu atau sekumpulan sesuatu yang dibaca dan ditulis oleh program – tentunya istilah tersebut didefinisikan di dalam dunia pemrograman. Sesuatu yang dimaksud dapat berupa karakter yang dikenal manusia (misal huruf dalam abjad latin), atau karakter-karakter lainnya yang telah didefinisikan sebelumnya dan dikenal komputer.

Masukan dapat dibaca dengan berbagai cara, misalnya mengetikkan langsung dengan *keyboard*, membaca dari berkas, dan sebagainya. Begitu juga dengan keluaran dapat ditulis dengan berbagai cara, misalnya menampilkan ke layar, menulis ke berkas, dan sebagainya. Adapun dikenal juga istilah masukan standar dan keluaran standar. Masukan standar merupakan sumber tempat membaca masukan yang berasal dari *keyboard*. Keluaran standar merupakan sumber tempat menulis keluaran ke layar. Selain itu, dikenal pula istilah *pipe* (*redirection*), yaitu suatu cara untuk mengalihkan sumber masukan/keluaran dari/ke tempat lainnya. Perhatikan perintah yang diketik di dalam Terminal (Linux), DOS prompt, atau Command (Windows) berikut.

<code>./program1 < in > out</code>	Linux
<code>program1 < in > out</code>	DOS atau Windows

`program1` merupakan nama berkas yang dapat dieksekusi (*executable file*), `in` merupakan nama berkas yang sudah ada dan bisa jadi sudah berisi sesuatu, dan `out` merupakan nama berkas. Karakter '`<`' merupakan karakter yang menandakan pengalihan masukan dari masukan standar menjadi dari berkas yang ada (dalam perintah di atas, dari berkas yang bernama `in`). Karakter '`>`' merupakan karakter yang menandakan pengalihan keluaran dari keluaran standar menjadi ke berkas (dalam perintah di atas, ke berkas yang bernama `out`). Jika sebelumnya berkas bernama `out` sudah ada dan berisi, maka isinya akan ditimpa dengan hasil keluaran dari `program1`. Tentu saja, nama berkas `in` dan `out` boleh diganti dengan nama lainnya, misal `SemogaProgramSayaBenarDenganInputIni.txt`.

Untuk Tutorial 0A ini Anda diminta untuk membuat sebuah program Java yang dapat membaca masukan per baris dari masukan standar dan menampilkan keluaran ke keluaran standar per baris.

Untuk Tutorial 0A, berkas kode sumber diberi nama "`SDA11100A.java`" (tanda kutip hanya untuk kejelasan). Semua kode sumber termasuk dokumentasi harus diletakkan dalam satu berkas. Batas waktu eksekusi program adalah 0,3 detik artinya program Anda harus berjalan paling lama 0,3 detik di *server* Aren (*server* di mana penilai otomatis berada). Batas memori program adalah 16 MiB (16 MebiByte = 16777216 Byte) artinya

paling banyak 16 MiB memori yang dapat digunakan program Anda selama berjalan. Untuk pekerjaan berikutnya, mahasiswa **wajib** memperhatikan nama berkas kode sumber, batas waktu eksekusi program, dan batas memori program dari masalah yang diminta untuk diselesaikan.

Bagaimana cara mengukur batas waktu eksekusi dan memori program? Silakan diskusi di forum.

Format Masukan

Masukan dibaca dari masukan standar. Masukan terdiri dari beberapa baris yang berisi untaian karakter *S*. Untaian karakter *S* hanya akan berisi huruf-huruf kapital A..Z, huruf-huruf kecil a..z, angka-angka 0..9, dan spasi. Panjang *S* tidak akan melebihi 1000 karakter (perhatikan bahwa mungkin saja terdiri dari 0 karakter) dan banyak baris pada masukan antara 1 s.d. 700.

Format Keluaran

Keluaran ditulis ke keluaran standar. Keluaran berisi baris-baris yang isinya sama persis seperti pada masukan.

Contoh Masukan

```
Tutorial 0A  
SDA Genap 2011
```

Contoh Keluaran

```
Tutorial 0A  
SDA Genap 2011
```

Panduan Singkat

Untuk membaca masukan dari masukan standar, kelas `Scanner` dari *package* `java.util` atau kelas `BufferedReader` dan `InputStreamReader` dari *package* `java.io` dapat digunakan. Kelas-kelas yang berada di dalam *package* harus di-*import* terlebih dahulu sebelum digunakan. Salah satu cara meng-*import* *package* `java.io` adalah sebagai berikut.

```
import java.io.BufferedReader;  
import java.io.InputStreamReader;
```

Contoh kode sumber untuk membuat obyek yang dapat digunakan untuk membaca masukan dari standar masukan adalah sebagai berikut (obyek yang dimaksud di bawah adalah `buffReader`).

```
BufferedReader buffReader = new BufferedReader(  
    new InputStreamReader(System.in));
```

Kompilasi kode sumber dapat dilakukan dengan cara berikut.

```
javac SDA11100A.java
```

Menjalankan program dapat dilakukan dengan cara berikut (dengan asumsi berkas bernama `in` sudah ada, misalnya berisi tulisan pada contoh masukan).

```
java SDA11100A < in > out
```

Jika program dijalankan tanpa *redirection*, sebagai akhir dari masukan gunakan Ctrl+D lalu enter pada Linux, atau Ctrl+Z lalu enter pada DOS atau Windows.

Khusus untuk Tutorial 0A, semua kasus uji (masukan dan jawaban yang diharapkan) disediakan sebelum batas waktu pengumpulan Tutorial 0 dan bisa diunduh di SCeLE SDA.

Contoh Kode Sumber

Berikut adalah contoh kode sumber yang dapat dipelajari untuk Tutorial 0A, tetapi belum tentu bisa dijadikan solusi Tutorial 0A.

```
import java.util.Scanner;  
  
public class SDA11100A {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        while (scanner.hasNextLine()) {  
            System.out.println(scanner.nextLine());  
        }  
    }  
}
```

Coba cari tahu mengapa bisa demikian! Jawaban atas pertanyaan ini cukup didiskusikan di forum saja.

Tutorial 0B

I/O 2

Nama berkas kode sumber : SDA11100B.java
Batas waktu eksekusi program : 1 detik / kasus uji
Batas memori program : 16 MiB / kasus uji

Sangat disarankan untuk menyelesaikan Tutorial 0A terlebih dahulu sebelum mengerjakan Tutorial 0B ini.

Buatlah sebuah program Java yang dapat membaca masukan sebuah bilangan bulat per baris dari masukan standar sebanyak yang diminta dan menampilkan keluaran bilangan bulat yang lebih besar tiga kali lipatnya ke keluaran standar per baris!

Format Masukan

Masukan dibaca dari masukan standar. Masukan terdiri dari tepat $(A+1)$ buah baris. Baris pertama berisi sebuah bilangan bulat A ($1 \leq A \leq 1000$). Baris ke-2 s.d. baris ke- $(A+1)$, masing-masing baris berisi sebuah bilangan bulat yang nilai mutlaknya tidak lebih dari dua miliar.

Format Keluaran

Keluaran ditulis ke keluaran standar. Keluaran berisi tepat A buah baris. Masing-masing baris berisi sebuah bilangan bulat yang besarnya tiga kali lipat dari bilangan bulat pada masukan dengan bilangan di baris ke- i pada keluaran berkorespondensi dengan bilangan di baris ke- $(i+1)$ pada masukan. Misalnya, bilangan di baris pertama pada keluaran besarnya tiga kali lipat dari bilangan di baris kedua pada masukan. Jika bilangan pada masukan merupakan bilangan negatif, tanda tersebut juga harus tetap tertulis pada keluaran.

Contoh Masukan

```
3
123
-9
0
```

Contoh Keluaran

```
369
-27
0
```

Tutorial 0C

I/O 3

Nama berkas kode sumber : SDA11100C.java

Batas waktu eksekusi program : 1 detik / kasus uji

Batas memori program : 16 MiB / kasus uji

Sangat disarankan untuk menyelesaikan Tutorial 0A terlebih dahulu sebelum mengerjakan Tutorial 0C ini.

Buatlah sebuah program yang dapat menulis isi masukan sama persis ke keluaran!

Format Masukan

Masukan dibaca dari masukan standar. Masukan terdiri dari X buah karakter ASCII 8-bit ($1 \leq X \leq 1000$).

Format Keluaran

Keluaran ditulis ke keluaran standar. Isi keluaran sama persis dengan isi masukan.

Contoh Masukan

Tutorial 0C SDA ini sangat menarik.

Contoh Keluaran

Tutorial 0C SDA ini sangat menarik.

Petunjuk

Kelas `BufferedInputStream` dan `BufferedOutputStream` dapat digunakan untuk menyelesaikan Tutorial 0C ini.

Tutorial 0D

Pola 1

Nama berkas kode sumber : SDA11100D.java

Batas waktu eksekusi program : 1 detik / kasus uji

Batas memori program : 16 MiB / kasus uji

Sangat disarankan untuk menyelesaikan Tutorial 0A dan Tutorial 0B terlebih dahulu sebelum mengerjakan Tutorial 0D ini.

Buatlah sebuah program yang dapat menampilkan pola bintang dengan syarat-syarat yang ditentukan!

Format Masukan

Masukan dibaca dari masukan standar. Masukan terdiri dari sebuah baris yang berisi sebuah bilangan bulat A ($1 \leq A \leq 20$).

Format Keluaran

Keluaran ditulis ke keluaran standar. Keluaran berisi A buah baris. Baris pertama berisi tepat A buah bintang (karakter '*'), baris kedua berisi tepat $(A-1)$ buah bintang, baris ketiga berisi tepat $(A-2)$ buah bintang, dan seterusnya hingga baris ke- A berisi tepat 1 buah bintang.

Contoh Masukan

5

Contoh Keluaran

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

Tutorial 0E

Pola 2

Nama berkas kode sumber : SDA11100E.java
Batas waktu eksekusi program : 1 detik / kasus uji
Batas memori program : 16 MiB / kasus uji

Sangat disarankan untuk menyelesaikan Tutorial 0A, 0B, dan 0D terlebih dahulu sebelum mengerjakan Tutorial 0E ini.

Buatlah sebuah program yang dapat menampilkan pola untaian karakter dengan syarat-syarat yang ditentukan!

Masukan

Masukan dibaca dari masukan standar. Masukan terdiri dari sebuah baris yang berisi sebuah bilangan bulat A ($1 \leq A \leq 20$).

Keluaran

Keluaran ditulis ke keluaran standar. Keluaran berisi A buah baris. Baris pertama berisi tepat sebuah karakter (karakter 'A'), baris kedua berisi tepat dua buah karakter (karakter 'B' dan 'C'), baris ketiga berisi tepat tiga buah karakter (karakter 'D', 'E', dan 'F'), dan seterusnya hingga baris ke- A berisi tepat A buah karakter. Jika karakter sudah melebihi 'Z', ulangi lagi dari 'A'. Perhatikan bahwa huruf-huruf yang ditampilkan adalah huruf-huruf kapital.

Contoh Masukan

8

Contoh Keluaran

A BC DEF GHIJ KLMNO PQRSTU VWXYZAB CDEFGHIJ
--