

Praktikum 2

Ajalise keerukuse empiiriline hindamine (II)

JAVA ELEMENTE

- Järjendi esitusi Javas
- Funktsioonitüüpi parameeter
- Ridade lugemine tekstifailist

Harjutusülesanded

Sorteerimisülesanne

- AKE_4. Aeglane sorteerimismeetod
 - AKE_5. Kiire sorteerimismeetod
- Juhiseid programmi *Ajagraafikud.java* käitlemiseks

Iseseisev töö nr 1

Järjendi esitusi Javas

Fikseeritud pikkusega järjend: massiiv.

```
int[] a;  
int[] b = new int[]{2, -3, 1, 0, 12}; //ehk: int[] b = {2, -3, 1, 0, 12};  
int[] c = new int[5];                //ehk: int[] c = {0, 0, 0, 0, 0};
```

Massiivi *d* pikkus (elementide arv): `d.length`

Massiivi *e* *i*-s element ($i = 0 .. e.length-1$): `e[i]`

```
import java.util.Arrays;
```

Massiivi *a* sorteerimine mittekahanevalt:

```
Arrays.sort(a);
```

Massiivi *a* koopia => *b*:

```
int[] b = a.clone();  
või  
int[] b = Arrays.copyOf(a, a.length);
```

Massiiv *b* sõnena (nt väljastamisel):

```
Arrays.toString(b);
```

Muutuva pikkusega järjend: **list**.

Klassi *ArrayList* (või ka klassi *Vector*) isend. Listi elemendi väärtus on viidatüüpi.

ArrayList realiseerib liidese *List*.

```
import java.util.List;  
import java.util.ArrayList;
```

```
ArrayList<Integer> a;  
List<Integer> b = new ArrayList<Integer>(); // listis b on 0 elementi  
List<String> c = new ArrayList<String>(); // listis c on 0 elementi
```

Lisada *elem* listi *d* (viimaseks): `d.add(elem)`

Listi *e* pikkus (elementide arv): `e.size()`

Listi *f* *i*-s element ($i = 0 .. f.size()-1$): `f.get(i)`

```
import java.util.Collections;
```

Listi *lst* sorteerimine :

```
Collections.sort(lst); -- mittekahanevalt (lst elemendid on nt tüüpi Integer või String)  
Collections.sort(lst, elementide võrdlemise lambda-avaldis);
```

Listi *lst* sorteerimine mittekasvavalt:

```
Collections.sort(lst, Collections.reverseOrder());
```

Listi *lst* transponeerimine:

```
Collections.reverse(lst);
```

Listi *b* segamine:

```
Collections.shuffle(b);
```

Juhuvalim listist *b*, *n* elementi:

```
List<Integer> tmp = new ArrayList<Integer>(b); -- tmp := b koopia  
Collections.shuffle(tmp);  
List<Integer> valim = tmp.subList(0, n);
```

Funktsioonitüüpi parameeter

Java8, ühemuutuja funktsiooni edastamine

Formaalsete parameetrite loetelus, meetodi või konstruktori päises:

```
(..., ..., Function<argumenditüüp, tulemusetüüp>) par_nimi, ...)
```

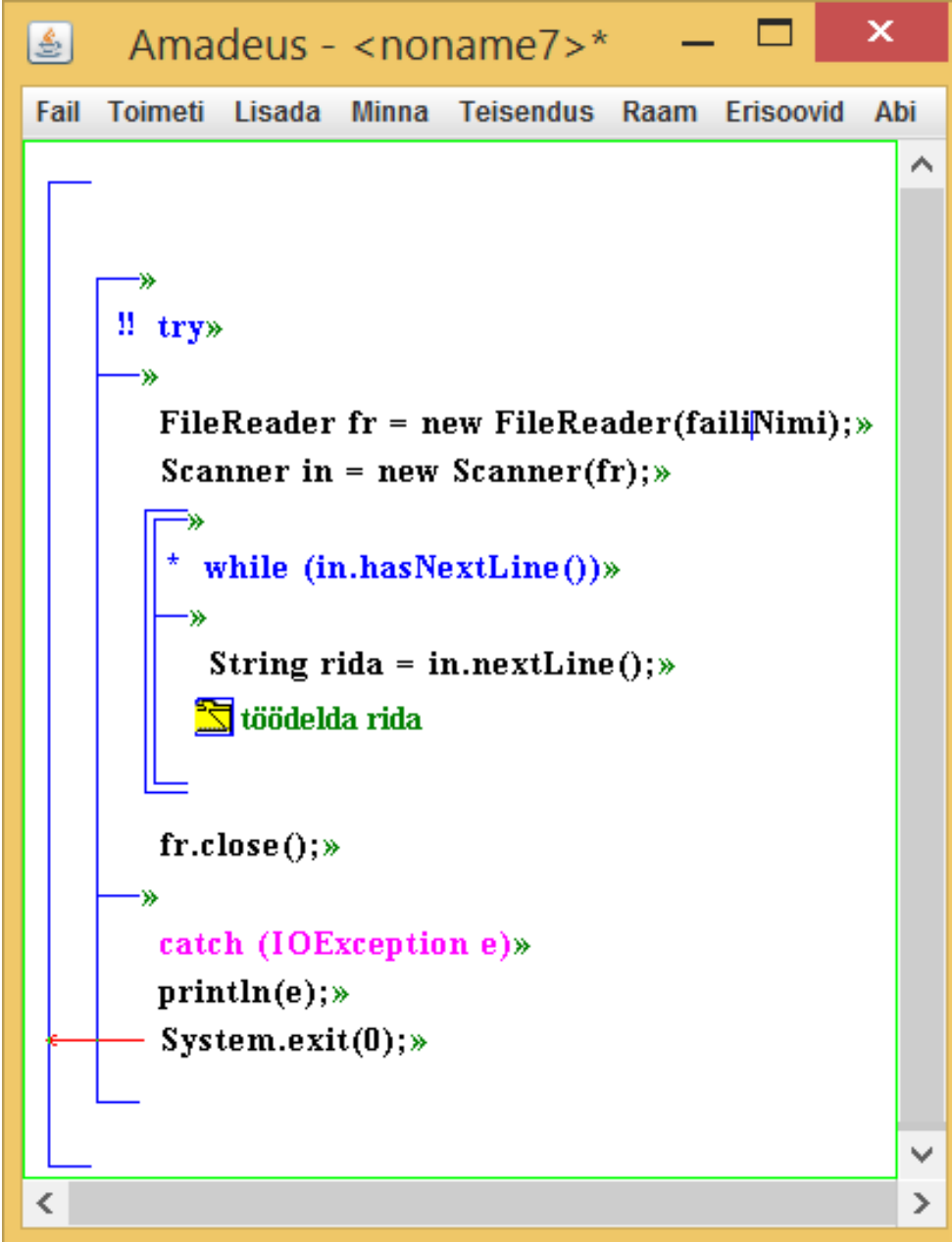
Rakendamine, selle meetodi või konstruktori kehas, avaldisena:

```
par_nimi.apply(x) --- x on argumenditüüpi, avadise väärtus on tulemusetüüpi
```

Tegelike parameetrite loetelus, meetodi või konstruktori väljakutses:

```
(..., ..., lambda-avaldis, ...)
```

Ridade lugemine teksifailist



```
Amadeus - <noname7>*
Fail Toimeti Lisada Minna Teisendus Raam Erisoovid Abi
!! try»
  FileReader fr = new FileReader(failiNimi);»
  Scanner in = new Scanner(fr);»
  * while (in.hasNextLine())»
    String rida = in.nextLine();»
    töödelda rida
  fr.close();»
  catch (IOException e)»
    println(e);»
    System.exit(0);»
```

Harjutusülesanded

Sorteerimisülesanne:

Antud: täisarvude järjend (massiivina).

Tulemus: antud järjendi elementide väärtused ümber paigutatud mittekahanevasse järjestusse.

AKE_4. Aeglane sorteerimismeetod

AKE_4.1. Kirjutada ja testida mullimeetod järjendi sorteerimiseks mittekahanevalt * .

AKE_4.2. Integreerida mullimeetod programmi *Ajagraafikud.java*, asendamaks selles pistemeetodit.

AKE_5. Kiire sorteerimismeetod

AKE_5.1. Kirjutada ja testida kiirmeetod järjendi sorteerimiseks mittekahanevalt, milles lahkemeks võetakse järjendi esimese elemendi väärtus [*Õpik*, lk 64] * .

AKE_5.2. Integreerida kiirmeetod programmi *Ajagraafikud.java*, asendamaks selles süsteemset sorteerimismeetodit *Arrays.sort*.

* Ülesande lahendus esitada programmina, mis sisaldab nõutud meetodit ja demonstreerib selle tööd juhujärjenditel pikkustega 1, 2, 3, 11, ning 14. Meetodi juhujärjendi loomiseks võib võtta programmist *Ajagraafikud.java*.

Juhiseid programmi *Ajagraafikud.java* käitlemiseks

Failide hankimine (esimene kord)

1. Salvestada (*Salvestada link nimega ...*) *Ajagraafikud.java* kodulehelt <http://kodu.ut.ee/~kiho/ads/fall17/> oma töökausta (nt *TK*).
2. Salvestada (*Salvestada link nimega ...*) fail *ee_ut_kiho_aa.zip* kodulehelt oma töökausta (*TK*).

Käitlemine

1. *File Explorer* : avada töökaust *TK*; avada käsurea aken (*File>ava käsuviip...*).
2. Töökaustas (*TK*) avada *Ajagraafikud.java* toimetiga *Notepad++*.
3. Kui vaja: Programmi *Ajagraafikud.java* toimetamine ja muudetu salvestamine.
4. Kompileerimine: käsurea aknas käsk (kopeerida *Ajagraafikud.java* koodist)
`javac -cp .\;ee_ut_kiho_aa.zip Ajagraafikud.java.`
Kui oli vigu siis jätkata punktist 3.
5. Kävitamine: käsurea aknas käsk
`java -cp .\;ee_ut_kiho_aa.zip Ajagraafikud`
6. Jätkata punktist 3 (või lõpetada).

Käsurea aknas: eelmist/järgmist käsku saab kätte alla/üles noolega;
programmi töö katkestamiseks **Ctrl+c**.

Iseseisev töö nr 1

Esitamise tähtaeg rühmal J. Liivi 2-202, E 12 -14: **23. september 2017, kell 12.15**

Esitamise tähtaeg rühmal J. Liivi 2-202, T 14 -16: **24. september 2017, kell 14.15**