



Université de La Rochelle
Universiteti i Prishtinës




Java biblioteka dhe konteinerët

K. Bertet, F. Berisha

Plani



- 1) Java biblioteka
 - 2) Java konteinerët
 - 3) Strukturat e të dhënave
 - 4) Mbi algoritmat dhe kompleksitetin
- 

Java biblioteka

API specifikimi (<http://java.sun.com>)

Pakot (grupime klasësh) kryesore:

<code>java.applet:</code>	krijimi i apleteve në Java
<code>java.awt:</code>	krijimi i interfejseve
<code>java.io:</code>	për input/output
<code>java.lang:</code>	klasat fundamentale (String, Math, ...)
<code>java.net:</code>	për rrjete
<code>java.util:</code>	klasat konteinerë (Set, List, ...) strukturat e të dhënave
<code>java.sql:</code>	për baza të dhënash
<code>java.swing:</code>	krijimi i objekteve grafike

.....

Java biblioteka



API Specifikimi (<http://java.sun.com>)

Klasa `java.lang.Math`:

Klasa `java.lang.Integer`:

Klasa `javax.swing.JButton`:

Klasa `java.util.LinkedList`:

Diagramet e klasave



Sajti: <http://www.falkhausen.de/en/diagram/html/>

Legjendë: klasat / relacionet e inheritimit: kaltër e mbyllur
interfejsat / relacionet e implementimit: kaltër e çelë

Diagrame klasash:

- e kornizës që implementon `java.util.Collection`
- e kornizës që implementon `java.util.Map`
- e kornizës (së vjetëruar) `java.util.Stack`

Klasat konteinerë

java.util.*

Collection: menagjon grupe objektesh:

- Set: bashkësi objektesh pa qasje direkte, pa duplikim
 - HashSet: elementet jo të renditura, **hash tabelë**
 - TreeSet: elementet e renditura, **pemë binare**
- List: listë objektesh, me qasje përmes indeksi, me duplikim
 - LinkedList: **listë dyfish e lidhur**
 - ArrayList: listë e mbështetur në **varg**
 - Stack: **stek** (mullar)

Map: menagjon çifte unike (çelës, vlerë):

- HashMap: çifte jo të renditura, **hash tabelë**
- TreeMap: çifte të renditura sipas çelësit, **pemë binare**

Klasat konteinerë dhe struktura e të dhënave

Disa struktura të dhënash, të renditura ose jo të renditura:

- **Listë dyfish e lidhur** (qasje përmes indeksi)
- **Varg** (qasje përmes indeksi)
- **Stek** (qasje majës së stekut)
- **Hash tabelë** (elementet jo të renditura, të vështira për t'u parametrizuar)
- **Pemë binare** (elementet e renditur sipas çelësit)

Cilat janë ndryshimet?
Cila të zgjedhet?

Rikujtoni algoritmat dhe kompleksitetin...

Strukturat e të dhënave

	Shtimi	Kërkimi	Largimi
Listë e lidhur: (shtimi te koka)	$O(1)$	$O(n)$	$O(n)$
Varg i sortuar: (kërkimi binar)	$O(n)$	$O(\log n)$	$O(n)$
Varg: (shtimi në fund)	$O(1)$	$O(n)$	$O(n)$
Stack/Queue: (shtimi/largimi te koka/maja)	$O(1)$	$O(1)$	$O(1)$
Pemë binare: (elementet e renditur)	$O(\log n)$	$O(\log n)$	$O(\log n)$
Hash tabelë: (kompleksiteti varet nga parametra të caktuar)	$O(1)$	$O(1)$	$O(1)$

Përfundim

Cili konteiner të zgjedhet? HashSet, TreeSet, LinkedList, ArrayList, Stack, HashMap, TreeMap?

Por poashtu: cila strukture e të dhënave të zgjedhet?

- **Listë dyfish e lidhur** (qasje përmes indeksi)
- **Varg** (qasje përmes indeksi)
- **Stek** (qasje përmes majës së stekut)
- **Hash tabelë** (elementet jo të renditur, vështirësi parametrizimi)
- **Pemë binare** (elementet e renditur sipas një çelësi)

Varet nga:

- të dhënat:

të renditura? të indeksuara? çift (çelës, vlerë)?

- kompleksiteti i operacioneve më frekvente:

qasje efektive? insertim? të dyja?