

Teme:**Razredni člani (2):**

- Notranji(inner) oz. gnezdeni(nestred) razredi

Avtomatično ovijanje(inboxing) in odvijanje(outboxing) primitivov v objekte/iz objektov

Enorazsežne tabele (utrjevanje):

- izvedba struktur:

- o sklad
- o vrsta(linearna, krožna),

Možna vira: pridana datoteka z realizacijo: 14_lav03_vaja_01c_xRazedniClani2_inboxOutbox.First.java.txt
14_lav03_vaja_01c_wLantern demo.java.txt

Naloga 1

Preglejte in preverite delovanje dane realizacije sklada v pridani datoteki (glej vir zgoraj). Oglejte si tudi posebnosti načina klica metode za vračanje elementov s sklada.

Naloga 2

Dodajte realizaciji sklada iz pridane datoteke metodo peek/0; metoda vrne vrednost z vrha sklada enako kot metoda pop/0 s to razliko, da je nedestruktivna (stanje sklada po izvedbi je enako stanju pred izvedbo)

Naloga 3

Programu (iz predhodne naloge) dodajte realizaciji:

- Linearno vrsto, kot notranji razred;
- Krožno vrsto, kot notranji razred;

velikosti obeh naj bosta enaki velikosti obstoječega sklada. Razširite demonstracijo z demonstracijo delovanja obeh dodanih realizacij.

(nasvet: veliko uporabljate copy&paste, prilagodite imena kazalcev, metod,)

Naloga 4

Izberite eno od realizacij (vseeno katero: sklad, lin. vrsta, krožna vrsta) in zamenjajte tip elementa v strukturi tako, da bo element vrste Integer namesto int (pri Sklad bo potrebno ustrezno 'pokvariti' kontejner, ter metodi push in pop).

Pokažite, da kljub menjavi vrste podatka v kontejnerju demonstracijski program še vedno deluje (dokaz za avtomatičen inboxing/outboxing metod primitivom in njegovo objektno ovojnico oz. v primeru med int in Integer) na enak način in z enakim rezultatom kot pred spremembo.

Naloga 5

Izvedite dinamično demonstracijo delovanja dodajanja in odvzemanja elementa iz izbrane strukture: obstoječa vizualizacija, dana z realizacijo sklada je linijska, kar pa pomeni, da potrebujemo za vsako spremembo stanja nov izris celotne strukture, da pač odrazimo spremembe. V naslednjem primeru bi uporabili paket Lanterna, ki je eden ob boljših javanskih približkov knjižnice 'curses', ki v besedilnem načinu omogoča absolutno pozicioniranje izpisov.

Razvoj paketa je dosegljiv na naslovu <https://github.com/mabe02/lanterna/tree/master>, kjer imate na voljo tudi dokumentacijo, in hitra navodila za namestitev paketa (če vaša okolja podpirajo upravljalca paketov maven ali gradle), direktno v maven : <https://mvnrepository.com/artifact/com.googlecode.lanterna/lanterna/3.1.3>

V primeru, da ne uporabljate upravljalca paketov (recimo v primeru rabe BlueJ), paket z vsemi odvisnostmi pridobite iz <https://jar-download.com> (katerokoli verzijo lanterne 3.x). Slednji po novem zahteva prijavo ...

BlueJ in namestitev paketa (ena od):

Per project:

- Tools->preference->libraries
- Uporabimo gumb Add File (desno) in polje 'user libraries from config' dodamo lanterna-3.1.3.jar oz. verzije, ki jo uporabimo
- Ponovno zaženemo BlueJ (v tools->preferences->libraries mora biti ob knjižnici oznaka loaded)

Globalna namestitev:

- Skopiramo datoteko v mapo\BlueJ\lib\userlib\ in nato zaženemo BlueJ

Maven (prenos paketa z vsemi morebitnimi odvisnostmi za uporabo z BlueJ)

- <https://maven.apache.org/download.cgi>
- prenesemo in razširimo: apache-maven-3.9.9-bin.zip (inačica iz časa nastanka dokumenta) v npr. c:\maven
- v mapo c:\maven dodamo ukazno (prenesi.bat) datoteko z vsebino:

```
set Path=c:\maven\bin;%PATH%
set MAVEN_HOME=c:\maven

mvn dependency:copy -Dartifact=com.googlecode.lanterna:lanterna:3.2.0-alpha1 -DoutputDirectory=.
```
- v mapi c:\maven zaženemo prenesi.bat
 - o rezultat uspešnega prenosa je datoteka c:\maven\lanterna-3.2.0-alpha1.jar
- !!

Pridana demonstracija rabe metod iz knjižnice izrisuje dodajanje naključne vrednosti na sklad. Seveda to niso vrednosti iz vašega sklada