

Лабораторијска вежба број 2 из Објектно оријентисаног програмирања II

У сваком задатку:

- Грешке пријављивати изузецима типа класа које садрже текст поруке.
- На располагању стоји класа `Citaј` у пакету `usluge`.

1) Написати на језику *Java* следеће типове:

- **Мерљивим** појмовима може да се одреди тежина.
- Мерљива **особа** има име и тежину. Може да се дохвати име и да се састави текстуални опис у облику *име (тежина)*.
- Мерљив теретни **контејнер** има јединствен, аутоматски генерисан регистарски број и тежину када је празан. У контејнер је могуће сместити товар задате тежине и извадити товар задате тежине. Може да се дохвати регистарски број и сопствена тежина и да се састави текстуални опис у облику *регБрој (укупнаТежина)*.
- Апстрактан **авион** има ознаку, максималну тежину, тежину када је празан и садржи низ од задатог броја мерљивих појмова. Може да се стави неки појам на задато место у низу, да се уклони појам са задатог места, да се израчуна тренутна тежина авиона и да се састави текстуални опис у облику *ознака (тренутнаТежина) [појам, ..., појам]*. Грешка је ако се покуша претоварити авион, ставити нешто на попуњено место или уклањати нешто с празног места.
- **Путнички** авион може да превози само путнике, а **теретни** авион може да превози само контејнере. Грешка је ако се покуша додати појам неодговарајуће врсте.
- **Аеродром** има назив и садржи одређен број авиона у сваком тренутку. Ствара се празан, а затим авиони могу долетати и одлетати по редоследу долетања. Може да се састави текстуални опис аеродрома, тако што се у једном реду испише назив аеродрома, а затим у потребном броју редова садржани авиони. Текстуални описи садржаних авиона се састављају конкурентно.

Написати на језику *Java* програм који направи аеродром и дода на њега један путнички авион са три путника и један теретни авион са два контејнера, све са константним параметрима (не треба ништа учитавати) и после испише аеродром на главном излазу.

2) Написати на језику *Java* следеће типове:

- **Странка** банке уплаћује (>0) или подиже (<0) случајан износ новца (у опсегу од -1000 до $+1000$) који може да се дохвати.
- Кроз активан **улаз** банке, кад је отворен, странке улазе у случајним временским интервалима од 1 до 3 s. За сваку странку која уђе у банку, улазак се региструје код банке. Свака странка стане у ред код шалтера испред којег чека најмањи број странака. Улаз може да се отвори, затвори и уништи.
- Испред активног **шалтера** банке може да чека произвољан број странака који се опслужују по редоследу пристизања. Може се добити информација о броју странака у реду испред шалтера. Ако у банци нема довољно новца странка се одбија. Опслуживање странака траје случајно време од 3 до 5 s и састоји се од исписивања броја шалтера, износа новца који се уплаћује/подиже и да ли је странка опслужена или је одбијена. За сваку странку која заврши рад на шалтеру се код банке региструје излазак. Сав новац се налази на једном месту у банци. Нит шалтера се блокира ако испред нема странака.
- **Банка** има један улаз и два шалтера. Може да се региструје улазак и излазак странке, да се дохвати тренутни број странака и количина новца у банци. Приликом отварања банке задаје се и исписује почетна сума новца у банци и отвори се улаз. Приликом затварања банке затвори се улаз у банку, сачека се да све странке напусте банку и испише се преостала сума новца у банци.

Написати на језику *Java* програм (с менијем) који прочита почетну суму новца у банци и дужину радног времена, отвори банку, сачека крај радног времена, затвори банку по истеку радног времена и понавља претходне кораке док за дужину радног времена не прочита негативну вредност.

3) Написати на језику *Java* следеће типове:

- За апстрактан **израз** може да се израчуна вредност реалног типа, да се створи израз који представља његов извод по задатој променљивој и да се састави текстуални опис израза.
- **Константа** је израз који има реалну вредност која не може да се промени после иницијализације. Извод константе је константа 0. Текстуални опис садржи вредност константе.
- **Променљива** је израз који има једнословно име и реалну вредност (подразумевано 0) која може да се промени после иницијализације. Извод променљиве по променљивој са истим именом је константа 1, а по било којој другој променљивој константа 0. Текстуални опис променљиве садржи име променљиве.
- **Бинарни** израз је израз који садржи два израза (a и b). При рачунању вредности израза вредности операнда се израчунавају конкурентно. Текстуални опис је облика $(a\#b)$, где је $\#$ симбол реализоване операције.
- **Збир, разлика, производ и количник** су бинарни изрази чије су вредности $a+b$, $a-b$, ab и a/b , а изводи $a'+b'$, $a'-b'$, $a'b+ab'$ и $(a'b-ab')/b^2$. Покушај дељења нулом је грешка.

Написати на језику *Java* програм који направи објекат за израз $(x+3)/((x-2)(x+1))$, испише на главном излазу алгебарски облик израза и његовог извода и после табелира вредност израза и његовог извода на главном излазу за свако $-2 \leq x \leq 3$ са кораком 0,25.

4) Написати на језику *Java* следеће типове:

- **Вредносно** је нешто чему може да се одреди вредност.
- **Јединица** мере има ознаку која може да се дохвати. Дозвољене ознаке су: "ком", "l", "m" и "кг". Грешка је ако се покуша направити јединица мере с другачијом ознаком. Текстуални опис јединица мере садржи ознаку јединице.
- **Артикал** има назив и јединицу мере који могу да се дохвате. Два артикла се сматрају истим ако имају исти назив. Текстуални опис артикла садржи назив артикла.
- **Млеко и шећер** су артикли који се мере у литрима, односно у килограмима.
- Вредносни **запис** о артиклу има артикал, количину и јединичну цену артикла. Могу да се дохвате поља записа, да се промени количина, да се промени јединична цена и да се израчуна вредност укупне количине артикла у запису. Текстуални опис записа садржи артикал, количину, јединицу мере, јединичну цену и вредност артикла.
- **Низ** записа може да садржи задат број записа. Ствара се празан, задатог капацитета, после чега се записи могу додавати један по један (капацитет низа се по потреби аутоматски повећава за 10%, али најмање за 5 места). Може да се дохвати број елемената у низу, да се дохвати елемент са задатим редним бројем (грешка је ако је индекс изван опсега) и да се састави текстуални опис садржаја низа, по један елемент у сваком реду, при чему се текстуални описи елемената састављају конкурентно.
- **Складиште** има назив и адресу и садржи низ записа о артиклима. Ствара се празно, почетног капацитета од 5 записа, после чега се записи додају један по један. Грешка је ако већ постоји запис за исти артикал. Може да се дохвати запис за артикал задат по називу (грешка је ако такав запис не постоји) и да се израчуна укупна вредност артикала у складишту.

Написати на језику *Java* програм који направи једно складиште, стави у њега неколико артикала и испише на главном излазу укупну вредност артикала у складишту. Користити константне податке (не треба ништа учитавати).

НАПОМЕНЕ:

- а) Потребно је решавати искључиво задатак чији се број добије на почетку вежбе.
- б) За израду лабораторијске вежбе, на располагању је **120** минута.
- в) Дозвољено је коришћење оригиналних књига и збирки задатака (не фотокопије).
- г) Није дозвољено коришћење унапред припремљених решења у било којем облику. Студент који користи унапред припремљена решења, биће удаљен уз анулирање поена на свим лабораторијским вежбама.
- д) У току израде лабораторијске вежбе, дежурни може студентима да постаља питања у вези њихових решења, што може утицати на број освојених поена на лабораторијској вежби.
- ђ) Студент може бити позван на накнадну одбрану рада, која може да утиче на број поена. Непојављивање студента на одбрани или показивање вишег степена неразумевања сопственог решења повлачи анулирање поена на свим лабораторијским вежбама.
- е) Резултат рада мора бити у *.java датотекама на диску L.
- ж) Оцене радова биће објављене на *Web*-у на адреси: kondor.etf.rs/~kraus/ (одреднице: *настава* | <име предмета> | *оцене* | *колоквијуми*).