

## 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* следећи пакет типова:

- **Клијент** има јединствен, аутоматски генерисан целобројан идентификатор и целобројан код врсте тражене услуге у опсегу од 1 до 3. Може да се дохвати идентификатор и врста тражене услуге клијента. Текстуални опис је облика *ид (код)*.
- **Ред** клијената садржи највише 10 клијената. Клијент се додаје на крај реда и уклања се с почетка реда. Ствара се празан после чега клијенти могу да се додају и уклањају. Покушај додавања клијента у пун ред или уклањања из празног реда привремено блокира нит извршиоца. Након сваког додавања и уклањања из реда исписује свој текстуални опис на главном излазу у облику *клијент, ..., клијент*.
- **Активан службеник** има јединствен, аутоматски генерисан целобројан идентификатор и једнословну ознаку врсте. Може да обавља одређену понављајућу активност. Резултат активности је обрађивани клијент. Након сваке извршене активности испише свој текстуални опис на главном излазу у облику *врста/ид/клијент* и чека случајно време у задатом опсегу пре него што понови активност. Могуће је покренути, привремено зауставити и дефинитивно обуставити рад службеника.
- **Портир** је службеник који "ствара" клијента који тражи случајну врсту услуге с подједнаким вероватноћама и смешта га у задати улазни ред. **Разводник** је службеник који преузима клијенте из задатог улазног реда и у зависности од врсте тражене услуге ставља их у један од задатих шалтерских редова. **Шалтерски службеник** опслужује и уклања клијенте из задатог шалтерског реда. Одговарајући редови се задају при стварању појединог службеника. Ознаке врсте службеника су, редом: **P**, **R** и **S**.
- Шалтерска **служба** садржи четири реда, један улазни и три шалтерска. Запошљава једног портира, једног разводника и три шалтерска службеника. Трајање чекања код портира је случајан временски интервал између 100 и 300 ms, код разводника између 150 и 250 ms а код шалтерског службеника између 200 и 1000 ms. Шалтерска служба може да се отвори, затвори и да се уништи. Портир пушта клијенте у службу само када је она отворена. При затварању службе, затечени клијенти у редовима се опслужују. Уништавање службе подразумева уништавање свих службеника, не чекајући да се затечени клијенти опслуже. Покретање службеника када раде и њихово заустављање када не раде нема никаквог ефекта.

Написати на језику *Java* **програм** који направи једну шалтерску службу, отвори службу, после 5 s затвори и после још 3 s је уништи. Користити константне податке (не треба ништа учитавати).

2) Написати на језику *Java* следећи пакет типова:

- **Предмет** има јединствен, аутоматски генерисан целобројан идентификатор и једнословну ознаку врсте предмета. Може да се састави текстуални опис појма који садржи ознаку врсте и идентификатор предмета.
- **Мерљивим** стварима може да се одреди реална величина.
- **Правоугаоник** је мерљив предмет задат дужинама ивица. Ознака врсте појма је **P**. Величину представља површина правоугаоника. Текстуални облик је **Pid(a, b)**.
- **Сфера** је мерљив предмет задат полупречником. Ознака врсте појма је **S**. Величину представља запремина сфере. Текстуални облик је **Sid(r)**.
- Уређен **низ** мерљивих ствари ствара се празан задатог капацитета после чега се ствари додају појединачно. Може да се дохвати број ствари у низу, да се извади из низа ствар задатог редног броја, да се одреди укупна величина свих ствари у низу **конкурентним** израчунавањем величина појединачних ствари и да се састави текстуални опис низа који садржи текстуалне описе садржаних ствари, једна ствар по реду. Грешка је ако се низ препуни или ако се покуша извадити непостојећа ствар.

Написати на језику *Java* интерактиван **програм** (с менијем) који може да извршава следеће команде: направи низ задатог капацитета, прочитај предмет и стави у низ, извади предмет из низа, испиши низ, испиши укупну величину предмета у низу и заврши с радом.

3) Написати на језику *Java* следећи пакет типова:

- Апстрактна **пошиљка** има јединствен, аутоматски генерисан целобројан идентификатор и реалну тежину који могу да се дохвате. Може да се састави текстуални опис пошиљке у облику *id(tez)*.
- **Приоритетним** стварима може да се одреди целобројан приоритет, који може имати следеће вредности: *најнижи*, *низак*, *нормалан*, *висок* и *највиши*.
- Апстрактна **приоритетна пошиљка** је пошиљка с приоритетом.
- **Писмо** је приоритетна пошиљка најнижег приоритета. Грешка је ако тежина прелази 500 g. Текстуални опис је **Pprio** [*id(tez)*].
- **Пакет** је приоритетна пошиљка задатог приоритета. Грешка је ако тежина прелази 50 kg. Текстуални опис је **PKprio** [*id(tez)*].
- **Приоритетни ред** приоритетних пошиљки се ствара празан, задатог капацитета, после чега се пошиљке додају и узимају једна по једна по нарастућем приоритету. Може да се дохвати број пошиљки у реду, да се одреди укупна тежина свих пошиљки у реду и да се састави текстуални опис реда који садржи текстуалне описе садржаних пошиљки, једна пошиљка по реду. Ако се ред препуни или ако се покуша извадити из празног реда нит извршиоца операције се привремено блокира.
- **Активан пошиљалац** сваке секунде генерише пошиљку случајне врсте и смешта је у задати приоритетни ред. У 20% случајева пошиљка је писмо случајне тежине од 20 g до 550 g. У осталим случајевима пошиљка је пакет случајне тежине од 1 kg до 55 kg и случајног приоритета од ниског до највишег.
- **Активан прималац** у случајним временским интервалима од 0,8 s до 1,2 s дохвата и исписује по једну пошиљку из задатог приоритетног реда.

Написати на језику *Java* **програм** који с главног улаза учита трајање симулације и капацитет приоритетног реда, направи један приоритетни ред, пошиљача и примача и покрене симулацију.

4) Написати на језику *Java* следећи пакет типова:

- **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 целобројан 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.

---

**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: [home.etf.rs/~kraus/](http://home.etf.rs/~kraus/) (одреднице: *настава* | <име пред-мета> | *оцене* | *колоквијуми*).