

## Лабораторијска вежба број 2 из Објектно оријентисаног програмирања II

У сваком задатку:

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

1) 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.
- Активна апстрактна **операција** је израз који у посебној програмској нити извршава одговарајућу бинарну операцију над операндима (изразима).
- **Збир** је операција која садржи два израза ( $a$  и  $b$ ). Вредност је  $a+b$ . Извод је  $a'+b'$ . Текстуални опис је облика  $(a+b)$ .
- **Разлика** је операција која садржи два израза ( $a$  и  $b$ ). Вредност је  $a-b$ . Извод је  $a'-b'$ . Текстуални опис је облика  $(a-b)$ .
- **Производ** је операција која садржи два израза ( $a$  и  $b$ ). Вредност је  $ab$ . Извод је  $a'b+ab'$ . Текстуални опис је облика  $(a*b)$ .
- **Количник** је операција која садржи два израза ( $a$  и  $b$ ). Вредност је  $a/b$ . Покушај дељења нулом је грешка. Извод је  $(a'b-ab')/b^2$ . Текстуални опис је облика  $(a/b)$ .

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.

2) Написати на језику *Java* следеће типове:

- **Производ** има јединствен, аутоматски генерисан идентификациони број и задату масу (g) која може да се дохвати. Може да се састави текстуални опис у облику *идБрој (маса)*.
- Активан **радник** има име и производи предмете задатом продуктивношћу (g/s). Може да му се зада захтев да произведе један производ задате масе који по завршеној производњи чува код себе док се не преузме од њега. Покушај задавања новог захтева док производ по претходном захтеву није завршен и преузет, или покушај преузимања производа док још није готов зауставља нит која поставља захтев док се не испуне услови за наставак рада. Може да се испита да ли постоји готов производ за преузимање и да се састави текстуални опис једног од облика: *име/сека* – ако чека на захтев за производом, *име/radi* – ако је производња у току, односно *име/производ* – ако постоји готов производ (*производ* је текстуални опис готовог производа).

Написати на језику *Java* интерактиван програм (с менијем) који може да извршава следеће операције:

- направи радника задатог имена и продуктивности,
- захтевај од радника да направи предмет задате масе,
- испитај да ли радник има готов производ,
- преузми производ од радника и испиши производ на главном излазу,
- испиши радника на главном излазу,
- заврши програм.

### 3) Написати на језику *Java* следеће типове:

- Апстрактан **предмет** има јединствен, аутоматски генерисан идентификациони број и једнословну ознаку врсте предмета. Може да се састави текстуални опис појма који садржи ознаку врсте и идентификациони број појма.
- **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 једна по једна. 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.

### 4) Написати на језику *Java* следеће типове:

- Апстрактна **пошиљка** има јединствен, аутоматски генерисан идентификациони број 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 целобројни 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 500g. Текстуални опис је **Pprio[id(tez)]**.
- **Пакет** је приоритетна пошиљка задатог приоритета. Грешка је ако тежина прелази 50kg. 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 нерастућем 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 20g до 550g. У осталим случајевима пошиљка је пакет случајне тежине од 1kg до 55kg и случајног приоритета од ниског до највишег.
- Активан **прималац** у случајним временским интервалима од 0,8s до 1,2s 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, пошиљаоца и примаоца и покрене симулацију.

---

#### 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 фотокопија).
- г) Није дозвољено коришћење унапред припремљених решења у било којем облику. Студент који користи унапред припремљена решења, биће удаљен и **губи право на полагање колоквијума**.
- 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) Резултат рада мора бити у \*.java датотекама у фасцикли Т: \ и њеним потфасциклама.
- е) Оцене радова биће објављене на *Web*-у на адреси: [galeb.etf.bg.ac.yu/~kraus/](http://galeb.etf.bg.ac.yu/~kraus/) (одреднице: *настава* | <име предмета> | *оцене* | *колоквијуми*).