ЕЛЕКТРОТЕХНИЧКИ ФАКУЛТЕТ
Одсек за СИ Београд 23. и 24.04.2012.

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 `Citaj` у пакету `usluge`.

1) Написати на језику *Java* следећи пакет типова:
- За ***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 операнада 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 $-2 \le x \le 3$ с кораком 0,25.

2) Написати на језику *Java* следећи пакет типова:
- ***Клијент*** има јединствен, аутоматски генерисан целобројан идентификатор и целобројан кôд врсте тражене услуге у опсегу од 1 до 3. Може да се дохвати идентификатор и врста тражене услуге клијента. Текстуални опис је облика *ид***(***кôд***)**.
- ***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 тејстуални 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).

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

- ***Аутомобил*** има јединствен, аутоматски генерисан целобројан идентификатор, задат капацитет резервоара и тренутну количину горива. Сви подаци могу да се дохвате. Почетна количина горива је случајна вредност између 10% и 30% капацитета резервоара. У аутомобил може да се сипа задата количина горива. Грешка је ако се резервоар препуни (тада се резервоар напуни и пријави грешка). Текстуални опис је облика *ид***(***гориво***/***капацитет***)**.
- Активан ***аутопут*** има задату бензинску станицу која може да се дохвати. У случајним временским интервалима од 0,5 *s* до 1 *s* ствара по један аутомобил капацитета резервоара 50 *l* који додаје тој станици. Може да се прекине рад аутопута када се прекида и рад његове станице.
- Активна ***пумпа*** се ствара за задату бензинску станицу. Пумпа циклички дохвата по један аутомобил из реда своје станице и сипа му потребну количину горива до пуног резервоара, брзином од 1 *l* на сваких 100 *ms*. Завршетак сипања дојави бензинској станици. Текстуални опис садржи текстуални опис аутомобила којег управо опслужује.
- Бензинска ***станица*** има четири пумпе и ред за чекање за највише 20 аутомобила. Станица може да се отвори и затвори, може да се прекине њен рад, да јој се дода задати аутомобил на крај реда, да се извади први аутомобил из њеног реда и да јој се дојави завршетак пуњења једног аутомобила. Ако је станица затворена или је ред пун, додавање аутомобила се занемари. Ако је ред празан, при узимању се сачека да се појави неки аутомобил. Приликом затварања, прекида се чекање возила у реду и чека се да се заврши сипање горива које је у току. Приликом прекидања рада станице прекида се и рад свих њених пумпи. Текстуални опис станице садржи текстуалне описе њених пумпи и низ идентификатора аутомобила који чекају на пумпе.

Написати на језику *Java* **програм** који направи једну бензинску станицу, отвори станицу, сваке 3 s испише станицу, после 6 исписа затвори станицу и на крају је уништи. Користити константне податке (не треба ништа учитавати).

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

- ***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** 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.

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 поставља 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/` (одреднице: *настава* | <име предмета> | *оцене* | *колоквијуми*).