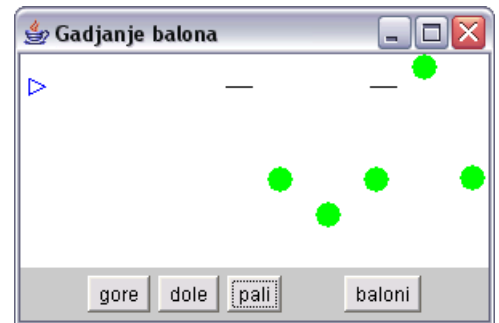


## Лабораторијска вежба број 3 из Објектно оријентисаног програмирања II

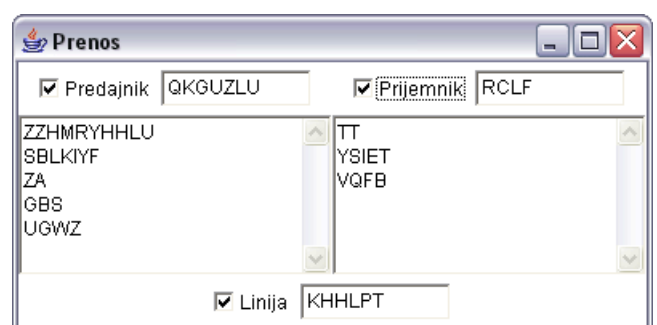
1) Написати на језику *Java* следећи пакет типова:

- Апстрактна **фигура** у равни предвиђа дохватање тренутних координата и исцртавање фигуре на графичкој сцени.
- Активна **сцена** је графичка компонента која може да садржи произвољан број фигура које исцртава сваких 40 ms. Ствара се празна, после чега се фигуре могу појединачно додавати. Из сцене може да се избаци одабрана фигура. Фигуре у сцени могу да се дохватају једна по једна, по редоследу додавања.
- **Стрела** је фигура која се исцртава као водоравна дуж задате боје и дужине. Ствара се са задатим координатама врха после чега се креће задатом реалном брзином удесно. Стрела се избацује из сцене када напусти видљиви део сцене.
- **Лук** је фигура која се исцртава задатом бојом унутар квадрата задате дужине ивица тако да једно теме показује удесно ( $\triangleright$ ). Ствара се са задатим координатама средине леве ивице после чега може да се помера за унапред задати корак нагоре или надоле, али само до ивица сцене. Може да испали стрелу задате дужине задатом реалном брзином.
- **Балон** је фигура која се исцртава као попуњен круг задате боје и пречника. Ствара се са задатим координатама центра после чега се креће задатом реалном брзином нагоре. Кад центар балона пређе горњу ивицу сцене, поново се појављује на доњој ивици на истој усправној линији као и пре. Кад балон буде пробушен неком стрелом, промени боју у другу унапред задату боју после чега се избацује из сцене. Сматра се да је стрела пробушила балон кад врх стреле доспе у круг балона.
- **Игра** на графичкој корисничкој површи према слици поставља један лук плаве боје у тачки (5, 20) у односу на горњи леви угао сцене унутар квадрата дужине ивица 10 p (*pixel*). Лук може да се помера нагоре и надоле с кораком 10 p. Стреле црне боје дужине 16 p испаљују се брзином 0,2 p/ms. На захтев, одједном се додаје 5 зелених балона пречника 16 p на доњој ивици сцене од десне ивице улево на међусобном размаку од по 30 p. Балони се крећу брзинама између 0,06 p/ms и 0,08 p/ms. Кад буду пробушени, боја им се мења у црвено.



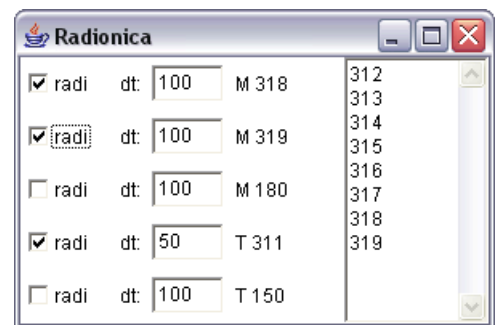
2) Написати на језику *Java* следећи пакет типова:

- Графички **приказивач** садржи графичку компоненту типа `TextComponent` на којој може да прикаже задати текст.
- **Бафер** може да садржи произвољан број једноредних порука типа `String`. Ствара се празан после чега могу да се додају поруке једна по једна, да се поруке узимају једна по једна (по редоследу додавања) и да се састави текстуални опис садржаја бафера. Ако при узимању у баферу нема поруке, нит читаоца се привремено зауставља. Текстуални опис бафера садржи све тренутне поруке, по једну поруку у сваком реду. После сваке промене садржаја бафера исти се приказује на вишередном приказивачу који се задаје приликом стварања бафера.
- Апстрактан **актер** у петљи извршава неку апстрактну акцију. Може да се дохвати назив врсте актера, да се постави једноредни приказивач за приказ стања, да се рад актера започне, привремено заустави, настави даље и заврши.
- **Предајник** је актер који генерише поруке случајне дужине од 2 до 10 знакова састављене само од великих слова. Генерисање сваког слова траје 40 ms. Поруке ставља у предајни бафер и на придруженом приказивачу приказује управо послату поруку.
- **Линија** је актер који поруке из предајног бафера преноси у пријемни бафер. Пренос сваке поруке траје 240 ms. Управо пренета порука се приказује на придруженом приказивачу.
- **Пријемник** је актер који у случајним временским интервалима између 120 и 360 ms из пријемног бафера узима по једну поруку и приказује на придруженом приказивачу.
- Графичка **командна плоча** је компонента типа `Panel` која садржи једног актера, коме придружује приказивач и која може да покреће и зауставља рад актера пољем за потврду.
- **Програм** на графичкој корисничкој површи према слици приказује пренос текстуалних порука од предајника до пријемника уз помоћ преносне линије.



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

- **Производ** има јединствен, аутоматски генерисан идентификациони број. Текстуални облик производа садржи само његов идентификациони број.
- **Складиште** производа може да смести задати број производа. Ствара се празно, после чега може да се додаје и узима по један производ. При покушају стављања у пуно складиште односно узимања из празног складишта нит извршиоца радње се привремено блокира. Текстуални облик складишта састоји се од низа текстуалних облика садржаних производа, један производ по реду.
- Апстрактан **механизам** предвиђа извођење неке радње.
- **Машина** је механизам чија се радња састоји од стварања једног производа и његовог стављања у складиште које је задато приликом стварања машине. Текстуални облик садржи слово **M** и текстуални облик производа који је управо стављен у складиште.
- Покретна **трака** је механизам чија се радња састоји од узимања једног производа из складишта које је задато приликом стварања траке. Текстуални облик садржи слово **T** и текстуални облик производа који је управо узет из складишта.
- Активан **мотор** понавља радњу једног механизма, који се задаје приликом стварања мотора, у случајним временским интервалима  $\Delta t \pm 20\%$  ms (на почетку је  $\Delta t = 100$  ms и може да се промени за време живота мотора). Рад мотора може да се привремено заустави, да се настави и да се дефинитивно прекине.
- **Графичко складиште** је складиште које се иницијализује графичким текстуалним простором (TextArea) на коме приказује сваку промену стања складишта.
- **Графички мотор** је мотор који се иницијализује графичком плочом (Panel) коју попуњава компонентама тако да може зауставити и наставити рад мотора, променити временски интервал  $\Delta t$  (промена има ефекта одмах) и приказати текстуални облик придруженог механизма.
- **Радионица** је програм који на графичкој корисничкој површи према слици приказује рад с једним складиштем капацитета 20, три машине и две покретне траке.



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

- **Упоредивач** предвиђа испитивање да ли се цео број *a* налази испред броја *b*.
- **Мање** и **веће** су упоређивачи код којих је *a* испред *b*, ако је  $a < b$ , односно  $a > b$ , респективно.
- **Уређивач** предвиђа уређивање задатог низа целих бројева коришћењем задатог упоређивача, као и могућност прекидања поступка уређивања.
- **Избор** је уређивач који примењује алгоритам избора за уређивање низа.
- **Програм** на графичкој корисничкој површи према слици уређује једноцифрене низове целих бројева задате дужине по растућем или опадајућем редоследу. Низови се приказују са по 20 елемената у једном реду. Дугме "Napuni" ствара низ задате дужине "duz" напуњен случајним цифрама. Обезбедити да дугме "Prekini" делује за време уређивања.



#### 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) и **Подсетника за АWT**.
- 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) Текст решења распоредити у \*.java датотеке (не стављати цело решење у једну датотеку) 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 предмета> | *оцене* | *колоквијуми*).