

ДОМАЋИ ИЗ ПРЕДМЕТА КОНКУРЕНТНО И ДИСТРИБУИРАНО ПРОГРАМИРАЊЕ ЗА ШКОЛСКУ 2018/2019 У ЈАНУАРСКОМ ИСПИТНОМ РОКУ

Домаћи задатак из предмета 13E113КДП и 13C113КДП се у јануарском и фебруарском испитном року школске 2018/2019 ради самостално.

Домаћи задатак се ради у програмском језику Јава. Домаћи задатак носи не више од 20 поена.

Предуслови за успешну одбрану домаћег су:

- 1: Уписана одговарајућа година ЕТФ одговарајућег смера.
- 2: Благовремена достава писаних материјала и електронске верзије решења (најмање 4 дана пре испита у испитном року).
- 3: Благовремено припремљени услови за неометану проверу рада програма у лабораторији катедре за РТИ ЕТФ-а (барем три дана пре одбране домаћег потребно је инсталирати одговарајуће програме у договору са дежурним лаборантом).
- 4: Успешно обављено усмено одбрана рада.

Усмена одбрана рада се састоји из следећег:

- 1: Кандидат који брани домаћи мора самостално да преведе и инсталира све потребне програме везане за приложено решење.
- 2: Кандидат мора да поседује потребан ниво знања о задатку.
- 3: Кандидат мора да буде свестан недостатака предложеног решења и могућности за њихово превазилажење.
- 4: Кандидат треба да тачно одговори на потребан број питања која се баве тематиком везаном за домаћи задатак.

Израда писаних материјала везаних за домаћи подразумева поштовање одговарајуће форме. Према тој форми сваки домаћи треба да има следеће елементе:

- 1: Насловну страну са јасно израженим обележјима који карактеришу овај факултет и овај предмет. Мора да садржи назив и лого факултета, назив предмета из кога се домаћи брани, назив задатка који се ради, пуно име и презиме аутора, као и број индекса, датум када је начињена прва верзија, датум када је настала текућа верзија и место где је одбрана вршена.
- 2: На првој страници после наслова рада и имена аутора следи садржај на српском језику писан курзивом - *Italic* фонтом Times New Roman 10 pt ћириличним писмом.
- 3: На наредној страни треба да се налази текст задатка који се ради. Уколико текстом задатка нешто није било довољно јасно назначено посебно уоквирити делове који су додате. Уколико предложено решење поседује извештај број недостатака њих назначити на посебан и лако уочљив начин, и предложити алтернативно решење који би отклонило наведене недостатке.
- 4: У наставку је потребно дати детаљан опис предложеног решења и свих његових карактеристика (овде није потребно стављати имплементационе детаље већ функционалне који су од суштинске важности за разумевање пројекта).
- 5: Након функционалне спецификације потребној је дати детаљан опис пакета, класа, интерфејса, функција и параметара, користећи **UML** спецификацију за опис интеракције (**дијаграм интеракције**).
- 6: Упутство за коришћење насталог програмског пакета као целине. Упутство треба да покрива два типа коришћења програмског пакета:
 - а) коришћење у регуларним ситуацијама од стране особе чији је ниво рачунарског знања минималан, а која нема претходно искуство у раду са сличним пакетима.
 - б) коришћење насталог програмског пакета особе чији је ниво познавања потребних вештина на задовољавајућем нивоу у циљу даљег усавршавања система.

7: Листинг програма са потребном количином коментара није потребно предавати у штампаном већ у електронском облику.

8: Примери рада програма у регуларним и ванредним ситуацијама, са потребним објашњењима.

9: Рад писати на српском језику уз чување оригиналних енглеских термина.

Оригинал рада треба да буде откуцан само са једне стране листова А4 формата (210 x 297 mm). Користити маргине: **2.5 cm** горња, **2 cm** доња, лева и десна. Рад треба буде писан ћириличним писмом уз коришћење фонта Times New Roman, величина писма: 10 типографских тачака (10 pt) у две колоне размакнуте **0,5 cm** уз поравнање типа Justify. Рад куцати обичним проредом и двоструким проредом између пасуса. Почетак пасуса куцати од почетка колоне. Поднасловне у раду писати масним словима (**Bold**) великим словима величина писма: 12 типографских тачака. Сва слова у раду треба да буду црна, а позадина бела. Све табеле и слике треба да имају одговарајући наслов и да буду нумерисане. Бројеви и наслови табела налазе се увек изнад табела. Бројеви и наслови слика налазе се увек испод слика. Нумерација страна се пише у доњем десном углу. Насловна страна се не нумерише. Насловна страна треба буде писан ћириличним писмом уз коришћење фонта Arial у једној колони уз поравнање типа Center. Насловна страна садржи: Назив Универзитета (величина писма: 16 типографских тачака); Назив факултета (величина писма: 16 типографских тачака); Име, средње слово, презиме и број индекса студента (величина писма: 16 типографских тачака); Наслов пројекта (величина писма: 22 типографске тачке); Предмет из кода се пројекат ради (величина писма: 16 типографских тачака); Место, година (величина писма: 14 типографских тачака).

НАПОМЕНА: Непоштовање горе наведених правила вуче умањење освојеног броја поена на усменој одбрани домаћег или у потпуности забрањује исту.

Домаћи задатак јануар 2019

Лицитације

Пројектовати дистрибуирани рачунарски систем који треба да омогући интернет лицитације. Програм треба да ради у систему који се састоји од више рачунара повезаних у LAN (Local Area Network) или WAN (Wide Area Network).

У систему постоји три типа програма:

1. Централни сервер који служи за пријем, праћење цене и власништва над предметима лицитације, као и за објављивање тренутног стања.
2. Подсервери на којима се обављају конкретне трансакције, односно куповина, продаја и израчунавање нове цене предмета лицитације.
3. Клијентски програм преко кога клијент прати тренутне цене предмета лицитације, издаје захтеве за куповином и продајом појединих.

Централни сервер својим претплаћеним корисницима нуди услуге праћења цене предмета лицитације. Ова услуга је таква да сервер периодично шаље свим претплаћеним клијентима тренутну цену свих предмета за које су они заинтересовани. Сервер шаље податке један по један, тако да назначавача име предмета, тренутну цену и време преостало за лицитацију. Ово је поступак који се периодично понавља, а та периода износи x секунди. Централни сервер податке о цени добија од подсервера који воде рачуна о појединим предметима лицитације. Централни сервер податке периодично прикупља са подсервера, при чему тај период износи y секунди. За сваки предмет постоји тачно један подсервер који води рачуна о њему. Уколико подсервер није доступан, покушава се поново за $y/2$ секунди. Ако три пута за редом податак не успе да се прибави, подсервер се проглашава недоступним и све нове трансакције се пребацују на неки други подсервер коме се додељује да води евиденцију о том предмету, док се подаци о предмету задржавају на последњим прочитаним вредностима.

Клијент има могућност да се претплати код сервера тако што пошаље корисничко име и шифру и периодично добија податке о ценама предмета лицитације за које је заинтересован. Може добити списак свих тренутно доступних предмета који се лицитирају. Осим прегледа свих цена, корисник може да обавља купопродајне трансакције над предметима лицитације. Трансакције које корисник може да обави су објава новог предмета за лицитацију, са називом предмета, кратким описом, почетном ценом и трајањем лицитације, и давање нове понуде за одабрани предмет, при чему треба да специфицира нову цену. Када централни сервер прими захтев за трансакцијом, прослеђује га подсерверу који је задужен за тај предмет. Уколико се ставља нови предмет на лицитацију, треба одабрати онај подсервер са најмањим бројем тренутних предмета на лицитацији који од тог тренутка постаје надлежан за тај предмет. Уколико у тренутку приспећа захтева за куповином већ постоји захтев за куповином са већом предложеном ценом, куповина се одбија.

Корисник има уз себе везан рачун, са одређеном свотом новца на њему. Приликом стварања рачуна, треба уписати иницијално стање на рачуну. Приликом продаје предмета, кориснику који продаје предмет се пребацује новац са рачуна корисника који тај предмет купује. Уколико корисник нема довољно новца на рачуну, куповина се привремено суспендује, на t минута. Ако и након истека тих t минута корисник нема довољно средстава на рачуну, трансакција се не обавља, већ се покушава продаја кориснику који је имао другу

најбољу понуду. Тог другог корисника треба обавестити да може да обави куповину и он треба да одговори у року од t минута да ли је и даље заинтересован. Уколико јесте, и има довољно новца на рачуну, он постаје власник предмета. У супротном, предмет поново иде на лицитацију, са истим почетним параметрима као и први пут.

Уколико се корисник предомисли, може да одустане од затражене трансакције, али понуда мора да буде активна барем z секунди ($z > 2 * x$, $x > y$). Када се обави трансакција или отказивање, корисник добија повратну информацију о томе.

Уколико листа подсервера која се води на централном серверу остане празна, централни сервер јавља грешку и обуставља све трансакције док се не појави барем један нови подсервер. Да би се подсервер пријавио у листу активних подсервера на централном серверу мора да пошаље централном серверу захтев за комуникацију. Том приликом централни сервер датом подсерверу додељује за које је предмете задужен. Да би балансирао оптерећење централни сервер може са једног подсервера да на други пребаци операције над неким предметима, али тако да у било ком тренутку максимално један води рачуна о једном предмету. Ни корисници ни подсервери не морају да буду у сталној вези са централним сервером. Веза може бити раскинута гашењем програма или затварањем комуникационог канала. Када се кориснички програм повеже може да тражи од сервера статусе свих трансакција. Треба обезбедити да сваки сервер може да у паралели да прима захтеве за већим бројем предмета.

Проблем решити користећи **Java RMI** мрежну комуникацију. За сваки од ова три типа рачунара треба да постоји одговарајући графички кориснички интерфејс (GUI треба да буде развијен користећи Java **SWING** компоненте). Сервери треба да има могућност покретања и без корисничког интерфејса.