

КОНКУРЕНТНО И ДИСТРИБУИРАНО ПРОГРАМИРАЊЕ

Лабораторијска вежба 2

Задатак: Multiple Atomic Broadcast Problem и дистрибуирано програмирање

Потребно је решити проблем недељивог емитовања у дистрибуираном окружењу. Произвођачи и потрошачи представљају клијентске програме који користећи централни сервер комуницирају и приступају дистрибуираним објектима, текстуалним датотекама, на серверу.

- Ради приступања текстуалним датотекама које захтевају произвођачи и потрошачи треба имплементирати интерфејс **Goods** који има методе за дохватање и постављање имена датотеке, као и за постављање и дохватање садржаја саме књиге. Интерфејс је дат у прилогу.
- Класа за произвођаче (**Producer**), чији је код дат у прилогу, има **main** методу и која се повезује са сервером ради уписа. Серверски порт преко кога се комуницира и IP адреса сервера се уносе као аргументи командне линије. Остали аргументи су имена производа које је потребно направити и убацити. Комуникација са сервером се обавља користећи мрежно програмирање у програмском језику Јава (**java.net** пакет). Користити објектне токове података. Клијенти и сервер треба све време да држе отворену прикључницу кроз коју комуницирају. Уколико на клијентској или серверској страни дође до прекида везе та нит се прекида и њени ресурси се ослобађају.
- Класа за потрошаче (**Consumer**) чији је код дат у прилогу, има **main** методу и која се повезује са сервером ради читања. Серверски порт преко кога се комуницира и IP адреса сервера се уносе као аргументи командне линије. Остали аргументи су имена производа које је потребно дохватити и сачувати те производе када се дохвате. Комуникација са сервером се обавља користећи удаљене позиве процедура у програмском језику Јава (**java.rmi** пакет).
- Произвођачи и потрошачи могу да приступају различитим производима у исто време. На серверу треба водити рачуна о томе да када произвођач ставља неки производ он га ставља у неки бафер који прима само производе са тим именом. У једном баферу се налази више производа који сви имају исто име. Када потрошач чита неки производ он чита производ из бафера који чува производе са датим именом, и исто важи и за произвођача.
- Синхронизација између нити на серверу се обавља користећи мониторинску класу **AtomicBroadcast**, коју је такође потребно имплементирати.
- Потребно имплементирати интерфејсе **Goods** и **AB**. Интерфејси који су дати у прилогу морају се користити као такви, без измена.
- Дате **main** методе у класама произвођача и потрошача такође није дозвољено мењати, уз изузетак замене **null** стварањем конкретног објекта на означеним местима.