

Задания к лекции 7 часть 2

Задание повышенной трудности: при успешном выполнении начисляются двойные баллы (10 баллов).

Все задания предполагают написание многопоточной программы.

1. Написать программу, позволяющую имитировать обслуживание клиентов парикмахерской. Создать два класса: парикмахер и клиент. Создаётся очередь из нескольких клиентов. Создаётся несколько парикмахеров, с каждым из них связан свой поток команд. Каждый мастер стрижет своего клиента случайное время (это определяется в функции потока), «извлекая» очередного клиента из очереди клиентов, когда мастер освобождается. У одного мастера может одновременно стричься только 1 клиент. Программа должна выводить номер парикмахера, номер клиента и общее количество клиентов, которым оказывается услуга. Работа завершается тогда, когда все клиенты подстрижены.
2. Написать программу, позволяющую имитировать работу процедурного кабинета. Создать два класса: медсестра и пациент. Создаётся очередь из нескольких пациентов. Создаётся несколько медсестёр, с каждой из них связан свой поток команд. Одна медицинская сестра обслуживать только одного пациента. Каждый пациент может занимать некоторое случайное количество времени медсестры. Поток, связанный с медсестрой, «извлекает» пациента из очереди, когда медсестра освобождается. Программа должна выводить на экран фио или номер пациента и данные о времени, проведенном на процедуре. Работа программы заканчивается тогда, когда все пациенты обслужены.
3. Написать программу, позволяющую имитировать работу с игровыми автоматами. Создать два класса: автомат и игрок. Есть несколько автоматов, одновременно на одном автомате может играть только один человек. С каждым автоматом связан свой поток команд. В каждом автомате есть некоторое количество денег. В процессе игры каждый автомат генерирует 4 числа в диапазоне от 1 до 3, если эти числа совпали, то игрок получает выигрыш в размере всей суммы, которая есть в автомате и автомат завершает работу, если совпали 3 числа, то игрок получает половину от суммы, имеющейся в автомате и автомат продолжает работу. Один игрок может сыграть несколько раз на одном автомате. Программа должна выводить на экран номера игроков и данные о результатах их игры. Работа программы заканчивается, когда не остаётся работающих автоматов.
4. Написать программу, позволяющую имитировать работу грузового терминала. Создать два класса: бокс и грузовик. Есть несколько боксов терминала, с каждым связан свой поток команд. В одном боксе терминала может обслуживаться только один грузовик, задержавшись там на случайное время. В каждом боксе есть определенное количество грузов, которые необходимо вывезти. Каждый грузовик за один заход может вывезти некоторое случайное количество груза, он может совершать несколько рейсов. Грузовики выстраиваются в очередь, а потоки, связанные с боксами «извлекают» элементы из очереди. Уехавший из терминала грузовик ставится в конец очереди. Программа должна выводить номер грузовика и количество единиц груза, вывезенных за каждый рейс. Работа завершается тогда, когда из всех боксов терминала вывезены все грузы.

5. Написать программу, позволяющую имитировать «идеальный экзамен». В экзамене может принимать участие несколько экзаменаторов, с каждым связан свой поток команд. Один экзаменатор одновременно может принимать экзамен только у одного студента некоторое случайное время. Студенты организованы в очередь, откуда потоки, связанные с экзаменаторами «извлекают» студентов). Оценка генерируется случайным образом в диапазоне от 2 – 5. При этом у студента есть неограниченное количество попыток сдать экзамен экзаменатору, если он получил 2 (то есть не сдавший студент ставится в конец очереди). Студент получивший оценку 3 и выше считается успешно сдавшим экзамен. Программа должна выводить на экран фιο или номер студента и данные об оценке за каждую попытку. Работа программы заканчивается тогда, когда все студенты успешно сдали экзамен.

6. Написать программу, позволяющую имитировать подачу статей в номер в издательстве. Есть несколько редакторов. Один редактор может одновременно принимать статьи только у одного автора, кроме того, он (редактор) может принимать только определённое количество статей (будем считать, что он работает над конкретным журналом, где может быть опубликовано определенной конечное количество статей). Автор может предложить за раз несколько статей (случайное) для размещения и делать это несколько раз. То есть авторы организованы в очередь, откуда их «извлекают» потоки, связанные с редакторами и потом ставят в конец очереди. Программа должна выводить на экран фιο или номер автора и данные о количестве поданных статей за раз и всего. Работа программы заканчивается тогда, когда у редакторов заполнены все журналы.

7. Написать программу, позволяющую имитировать работу булочной. Создайте классы Клиент и Касса. В булочной есть несколько касс, с каждой связан свой поток команд. Ежедневно к продаже на каждой кассе предлагается некоторое количество изделий. На одной кассе может одновременно обслуживаться только один клиент, который может купить некоторое количество изделий несколько раз в день. Клиенты организованы в очередь, откуда они извлекаются потоком, связанным с кассой, после чего ставятся в конец очереди. Программа должна выводить на экран номер или фιο покупателя и данные о количестве изделий, приобретённых за раз и за день. Работа программы заканчивается тогда, когда заканчиваются хлебобулочные изделия.

8. Написать программу, позволяющую имитировать работу банкоматов (работающих только на выдачу денег) отделения банка. Есть несколько банкоматов, с каждым связан свой поток команд. На одном банкомате может обслуживаться одновременно только один клиент, у каждого банкомата есть резерв денежных средств и лимит на единовременную выдачу. У каждого клиента есть счёт с определенным балансом, клиент может снимать деньги в количестве не превышающим лимит банкомата и размер остатка на счету с банкомата несколько раз. Клиенты организованы в очередь, откуда их извлекают потоки банкоматов, после чего помещают в конец очереди. После каждого снятия деньги списываются со счета клиента. Программа должна выводить на экран фιο или номер клиента и данные о количестве снимаемых денег. Работа программы заканчивается тогда, когда заканчиваются деньги во всех банкоматах.

9. Написать программу, позволяющую имитировать работу регистратуры. В регистратуре есть несколько окон, с каждым связан свой поток команд. В одном окне регистратуры могут одновременно обслуживать только одного пациента, у каждого окна

есть определенное количество рабочего времени. Каждый пациент может занимать некоторое случайное количество времени в окошке. Клиенты организованы в очередь, откуда их извлекают потоки, связанные с окнами. Программа должна выводить на экран ф.и.о. или номер пациента и данные о времени, проведенном в окошке. Работа программы заканчивается тогда, когда заканчиваются клиенты.

10. Написать программу, позволяющую имитировать гонки (тараканьи бега). Есть 3 дорожки, с каждой связан свой поток команд. На одной дорожке может бежать один таракан. Тараканы организованы в очередь, откуда их извлекают потоки, связанные с дорожками. После того, как все дорожки освободились объявляется победитель забега и может начинаться новый забег. Программа должна выводить результаты гонок для всех забегов (например, номер таракана и время прохождения дорожки, номер победителя забегов по итогам гонок). Программа заканчивается, когда все тараканы приняли участие в состязании.