

Домашно задание №3

Вариант 1

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Студент:

Ф№: Група:

Студент:

Ф№: Група:

Задание: „Черният петък“ е популярният ден след „Деня на благодарността“, в който се осъществява традиционното за САЩ „лудо пазаруване“. В този ден тълпи от хора се стичат по магазините, за да се възползват от промоциите, с които се открива началото на коледните празници.

Във вашият скромнен магазин се продават два вида стоки:

- Обувки – характеризират се с модел (число от 1 до 20), номер (byte число от 33 до 50), доставна цена и крайна цена (цяло число от 20 до 200 долара), като крайната цена трябва да е винаги по-висока от доставната;
- Фланелки – характеризират се с модел (число от 1 до 10), размер (S, M, L, XL, XXL), доставна цена и крайна цена (цяло число от 5 до 40 долара), като крайната цена трябва да е винаги по-висока от доставната.

Дейностите, които се извършват са:

- Доставка на стока – имате два основни доставчика, единият доставя обувките, а другия фланелките;
- Продажба на стоката – обикаляйки магазина, вашите клиенти взимат различни стоки, след което отиват на една от двете каси и плащат.

Първият път, когато ви се е наложило да работите по време на „черния петък“ станал пълен хаос. Затова вие сте решили да направите напълно автоматизирана електронна система, която организира процеса „от-до“. Основните дейности на системата са:

- Да се води пълен отчет за това от коя стока има наличност и ако дадена свърши (например установявате, че вече нямате обувки от модел 5 с размер 42) трябва да направите поръчка до съответния доставчик;
- Да организирате продажбата на касите.

Първоначално сте заредили магазина си с всички възможни модели и размери от всичко, като имате по 3 броя от всеки вид (и за това сте изхарчили всичките си пари). Организирайте ги в подходящи списъци. След това организирайте:

- Нишката за вход на новите клиенти – на всеки 3 секунди влизат произволен брой (използвайте `Math.random()`) клиенти от 1 до 3ма на брой. Вашата охрана не позволява в даден момент в магазина да има повече от 50 клиента;
- За всеки клиент стартирайте нишка за извършване на избор на продукти – всеки клиент си избира произволно количество стока (от 1 до 5 продукта) от произволен вид, модел и размер, като избора на продукт му отнема от 1 до 3 секунди. Ако клиента си е избрал стока, която е свършила, той започва нервно да чака докато му я доставите;
- След като клиентът е свършил с избора на продукти, той се нарежда на една от двете каси – естествено той избира тази каса, на която опашката е по-малка;
- Обслужването на един клиент на дадена каса отнема 1 секунда (ако е избрал по-малко от 3 продукта) или 2 секунди (ако е избрал 3 или повече продукта). След извършване на плащането, този клиент изчезва завинаги. Не забравяйте да му вземете парите (сумата от крайните цени на избраните от него стоки)!;
- Ако установите, че дадена стока е свършила (например някой клиент е взел последната фланелка от тип 6 размер S), вие незабавно уведомявате доставчика да донесе 3 бройки от нея от склада. На доставчика му се отнемат от 5 секунди, за да ви достави тази стока. Той си води отчет какви междинни доставки ви е правил.

В края на работния ден сте установили, че сте успели да обслужите точно 200 клиента (използвайте това число като лимит за спиране на достъп на нови клиенти, което ще бъде и индикация, че приключвате изпълнението на програмата)! Вече е време е да се разплатите с доставчиците за допълнителните междинни доставки, които са направили. Сумирайте парите от касите и извадете от тях нужната сума. Колко е вашата печалба?

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 2

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Да се моделира „K списък“ със следните качества – той притежава същите качества както стандартния едносвързан списък, но за всеки елемент освен връзка към следващ елемент се пази и връзка с K-ти елемент напред. По този начин може да се осъществи евентуално по-бързо търсене. Например при 10 ако търсите 62ри елемент стандартно ще направите 62 операции със стандартната функция `get(62)`:

1 -> 2 -> 3 -> ... -> 61 -> 62

Докато използвайки модифицираната функция `get(62, true)` търсенето ще бъде с 9 операции:

1 -> 10 -> 20 -> 30 -> 40 -> 50 -> 60 -> 61 -> 62

Направете `main` метод, в който създайте списък с няколко хиляди елемента. След това пуснете две нишки, които стотици пъти търсят последователно различни елементи от списъка. Проверете с колко по-бърза ще бъде “K” нишката.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 3

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Напишете имплементация на структура от данни „Чанта“ (Bag) в клас на Java. Чантата е много подобна на обикновено множество, но позволява повторения. Например {'to', 'be', 'or', 'not', 'to', 'be'} е чанта пълна с думи. Реализирайте стандартните методи за колекции (добавяне, премахване, търсене, променяне на елементи и т.н.). Направете класа синхронизиран така, че да може да бъде използван в многонишково програмиране.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 4

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Да се реализира програма, която търси прости числа, като се възползва от всички ядра на процесора (т.е. пуска толкова нишки, колкото ядра има). За да вземете броя на ядрата използвайте

```
int cores = Runtime.getRuntime().availableProcessors();
```

По зададено число "N" от тип long да се намерят всички прости числа от 2 до N. Да се сравни отнетото време между многозадачната система и еднонишкова.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 5

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Студент:

Ф№: Група:

Задание: Създайте приложение, което моделира асансьор, като се отчитат следните параметри:

- Сградата ви има 12 на етаж. Асансьора може да се движи между всеки един от тях.
- Товароносимостта на асансьора е максимум 8 човека.
- Времето за преместване на асансьора от един етаж на друг е 3 секунди.
- Времето за качване на хора в асансьора на всеки един етаж е 5 секунди.
- На всеки етаж (без първи и последен) има по два бутона за извикване – нагоре и надолу. Асансьора няма да спре на етажа ако вие сте натиснали надолу, а той се движи нагоре.
- Ако капацитета е запълнен, хората, които не могат да се качат ще изчакат на етажа, като извикват асансьора пак.

Идеята на програмата е пред определен регулярен интервал от време да създавате нови нишки с хора, които извикват асансьора от произволен етаж и желаят да се преместят на друг произволен етаж (използвайте `Math.Random()`), за да генерирате произволни числа и ги привеждайте към `int` от 1 до 12). В конзолата да се отпечатват различните събития, например:

- Асансьорът е извикан на 5 етаж за движение нагоре;
- Асансьорът спря на 6 етаж, качват се 2ма човека. Асансьорът продължава да се движи надолу;
- Асансьорът спря на 4 етаж, слизат 3 човека, качват се двама нови. Няма чакащи на етажи 1-3, затова асансьорът продължава да се движи нагоре;
- Асансьорът спря на 5 етаж...

И т.н.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 6

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: „Лакомниците“ се характеризират с това, че ядат обяда си с по две вилници едновременно – една в лявата и една в дясната си ръка. Етикета на лакомника не му позволява да яде само с една вилица.

Така веднъж N на брой лакомника се наредили на една кръгла маса. Между всеки двама от тях стояла по една вилица, а пред всеки имало обяд. С други думи до всеки един лакомник стояла по една вилица от ляво и една от дясно. В точно определено време един от тях казал „старт“ и всички се втурнали едновременно да грабват вилници. Естествено когато някой вземе вилица, съседа му от тази страна няма да има вилица за едната си ръка и съответно той няма да може да яде – трябва да чака докато съседа му се наяде и остави вилицата обратно.

С течение на времето лакомниците установили, че тази подредба предразполага до създаване на „мъртва хватка“ (dead lock). Веднъж се случило така, че всеки държал по една вилица в ръка и чакал някой да остави друга. Всички останали гладни и се скарали помежду си.

Моделирайте задачата на Java.

Поради тази причина на следващото хранене лакомниците си поканили келнер. Идеята на келнера е да контролира раздаването на вилниците по начин, по който да се предотврати deadlock. Измислете алгоритъм за това и го реализирайте - моделирайте модифицираната задача на Java.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 7

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Студент:

Ф№: Група:

Задание: В студентския стол служителките правят ястия във фурната в задната стая и ги редят на щанда. Студентите от своя страна минават и грабят от щанда – всеки студент по един. Щандът е масив с капацитет от N ястия. В случая трябва да моделирате два типа нишки:

1. В първата група от P на брой нишки имате P на брой служителя, които се стремят щанда да е винаги запълнен и съответно непрекъснато всеки един от тях слага по едно ястие на последното свободно място;
2. Във втората група имате Q студента, които са се наредили на импровизирана опашка и с помощта на M на брой нишки грабят с по M ($M < N$) на брой ястия наведнъж.

Ако щандът се запълни докрай с ястия, служителите ще изчакат докато някое ястие бъде взето. Обратно – ако щандът в даден момент е празен, студентите ще изчакат някой служител да сложи такова. Програмата продължава дотогава, докато всичките Q студента бъдат обслужени. Приемаме, че броят на ястията в кухнята е точно Q на брой (ако ястията в кухнята свършат, служителите спират своята работа и отиват в почивка).

В един момент се оказало, че в тази динамична среда може да се получи неприятна „мъртва хватка“ (deadlock). Причината била наличие на т.нар. “race condition”:

http://en.wikipedia.org/wiki/Race_condition#Example

Прочетете повече за проблема в статията в Уикипедия за т.нар. „Producer-Consumer problem”:

http://en.wikipedia.org/wiki/Producers-consumers_problem

В тази статия има дадено готово решение организирано със стек и $P=1$, $N=10$, $Q=\text{inf}$ (безкрайна опашка от консуматори), $M=3$. От вас се изисква, освен да го прочетете и разберете, да го реализирате с опашка и да го обобщите.

Примерно изпълнение ще бъде 3 служителя ($P=3$) да разпределят по 10 ястия ($N=3$) за 400 студента ($Q=400$), които грабят по 5 ястия наведнъж ($M=5$).

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 8

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: „Лакомниците“ се характеризират с това, че ядат обяда си с по две вилници едновременно – една в лявата и една в дясната си ръка. Етикета на лакомника не му позволява да яде само с една вилица.

Така веднъж N на брой лакомника се наредили на една кръгла маса. Между всеки двама от тях стояла по една вилица, а пред всеки имало обяд. С други думи до всеки един лакомник стояла по една вилица от ляво и една от дясно. В точно определено време един от тях казал „старт“ и всички се втурнали едновременно да грабват вилници. Естествено когато някой вземе вилица, съседа му от тази страна няма да има вилица за едната си ръка и съответно той няма да може да яде – трябва да чака докато съседа му се наяде и остави вилицата обратно.

С течение на времето лакомниците установили, че тази подредба предразполага до създаване на „мъртва хватка“ (dead lock). Веднъж се случило така, че всеки държал по една вилица в ръка и чакал някой да остави друга. Всички останали гладни и се скарали помежду си.

Моделирайте задачата на Java.

Поради тази причина на следващото хранене лакомниците си изградиха система. Идеята била вилиците да бъдат номерирани и подредени във възходящ ред между тях. Всеки един от лакомниците трябва да взема вилиците последователно – първо вилицата с по-малкия номер и после вилицата с по-големия номер и съответно да ги връща в обратен ред – първо поставя вилицата с по-голям номер и после тази с по-малък. Теоретично се оказало, че това ще разреши проблема с deadlock. Ако N-1 лакомници са грабнали по една вилица, то те ще са взели всички вилници с номера от 1 до N-1, т.е. ще е останала само вилицата с най-голям номер. Така най-бавният лакомник няма да може да я вземе, защото конвенцията казва, че той трябва да вземе първо вилица с по-малък номер от останалата. Така ще се окаже, че един от лакомниците винаги ще може да грабне две лъжици!

Моделирайте модифицираната задача на Java.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 9

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Представете си, че имате бръснарница с един бръснар, един стол за бръснене и чакалня с три стола за изчакване. Бръснаря може да бръсне клиент (ако има такъв), а може и да си почива (ако няма никой нито на стола, нито в чакалнята). Когато бръснарят обръсне даден клиент, той отива и проверява в чакалнята дали има клиенти – ако има, той го извиква най-дълго чакалия клиент и започва да го бръсне, а ако няма, той отива да спи. Клиентите следват сходна последователност – когато клиент влезе в магазина, той поглежда дали бръснаря е зает. Ако бръснаря спи, клиента го събужда и сяда на стола да бъде обръснат. Ако бръснаря е зает и бръсне друг човек, новият клиент отива в чакалнята и проверява дали има свободни места. Ако има свободни места, той сяда, а ако няма свободни места, си тръгва.

Моделирането на тази задача е класически проблем, в компютърните науки, защото има вариант за мъртва хватка (deadlock). Ако например влезе нов клиент и видш, че бръснаря постригва човек, т.е. е зает, клиентът се насочва към чакалнята. Точно в този момент обаче бръснаря свършва с пострижката и също отива към чакалнята. Оказва се, че бръснаря е по-бърз и вижда празна чакалня. Така той отива да спи, а новия клиент сяда да чака да бъде повикан (което няма да се случи).

По-подробно задачата е обяснена и решена с псевдокод на следния адрес:

http://en.wikipedia.org/wiki/Sleeping_barber_problem

Вашата задача е да решите задачата на Java, като имате N на брой бръснаря и чакалня с M на брой места. Когато влезе нов клиент, той проверява дали има спящ бръснар. Ако има, сяда. Ако няма, проверява дали има свободно място. Ако има сяда на него, ако няма си тръгва. Бръснарите пък действат както в класическата задача – след като свършат да бръснат, проверяват чакалнята. Ако има клиент, го взимат. Ако няма, лягат да спят.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 10

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Студент:

Ф№: Група:

Задание: Настъпило времето за втората контролна работа по ПИКЗ. В аулата, която събира 200 души, се насочили всички 200 студенти – 8 групи по 25 човека. Скупчили се пред вратата и започнали да чакат да бъдат пуснати. Асистентите – 4 на брой - стояли вътре в залата и кроели хитър план, с който да се намалят подсказванията и преписванията. Идеята била да се подредят студентите по такъв начин, че да не е възможно в околност на един студент да има друг, който да е от същата академична група. Едно примерно решение е следното подреждане на 25 реда по 8 групи с циклично разместване на групите през две места:

1	2	3	4	5	6	7	8
3	4	5	6	7	8	1	2
5	6	7	8	1	2	3	4
7	8	1	2	3	4	5	6
1	2	3	4	5	6	7	8
...

Станала обаче беля – под натиска упражнен върху вратата, пантите подали и тя се изкъртила. Така започнали да нахълтват студенти и да седат “кой където му скимне”. Вривата била неопикуема и асистентите се видели в чудо. Понеже не успели да проведат инструктаж, започнали да хващат вече седнали студенти и да ги преразпределят на правилни за тях позиции. Трябва да моделирате задачата ако имате следните параметри:

- Капацитета на вратата позволява да влизат по максимум четирима студенти наведнъж;
- Първоначално влизат четирима и си избират напълно произволни мяста, на които да седнат (използвайте `Math.Random()` за да генерирате произволни числа за ред и стълб);
- Сядането на тези четирима студента им отнема от 1 до 3 секунди на всеки, в зависимост от това дали е избран от първи до девети ред (1 секунда), от десети до осемнадесети ред (2 сек.) или от деветнадесети до двадесет и пети ред (3 сек.).
- В момента, в който някой четиримата студенти седне, друг нов влиза през вратата;
- Всеки от четиримата асистенти започва да дебне за неправилно седнали студенти. Ако даден асистент намери студент, който не е седнал на правилното място, той моментално му се скарва и го премества на първото свободно за неговата група. Това действие отнема фиксирано време от 2 секунди.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 11

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: В сайта на любителите на алкохола решили да организират онлайн анкета, в която били зададени пет въпроса:

Обичате да пиете:

1. Бира
2. Водка
3. Ракия
4. Уиски
5. Аз не пия
6. Няма да отговоря

Идеята на собствениците била да запишат потребителските имена на отговорилите с “5” потребители и да ги баннат от форума. Колкото до парадоксалния въпрос №6, в който хем човекът е отговорил, хем не иска да отговаря – тези потребителски имена също били запазвани в списък с „наистина пияни“, който впоследствие щял да бъде използван, за да бъдат наградени с форумна значка.

Съвсем кратко след пускането на анкетата се видяло, че много потребители, явно членове на някоя от групите „любител на...“, решили да атакуват анкетата и да я компрометират. Така те започнали да гласуват по множество пъти. Анкетата била спряна и авторите започнали да мислят за решение. След известно време решили, че ще записват имената на вече гласувалите в хеш таблица и ще забраняват повторното гласуване.

Организирайте системата и направете пример като започнете да генерирате потребители с произволни **трибуквени** имена (използвайте `Math.Random()`, за да генерирате произволно число и след това на базата на него избирайте малка буква на латиница a-z). За всеки генериран потребител пуснете отделна нишка, в която генерирайте произволно число от 1 до 6 и се опитайте да гласувате в анкетата.

Този процес трябва да се повтаря до достигане на максималния възможен брой потребители – 3 букви по 26 възможни в азбуката = 78 възможни гласувания. Накрая изведете резултати, в които напишете кое питие колко любителя има, кои са пияните и кои са баннатите потребители, както и колко са били повторните опити за гласуване.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 12

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Две матрици - $A=[a_{ik}]$ и $B=[b_{kj}]$ с размерности $m \times r$ и $r \times n$, са записани в два двумерни масива, член променливи за клас с `main` метод. Матрицата $C=[c_{ij}] = A \cdot B$ се получава от умножението на тези две матрици. Създайте отделна нишка за изчисляването на всеки отделен елемент c_{ij} . Ясно е, че ще бъдат създадени $m \times n$ на брой нишки. Изведете резултата на екрана.

Проверете дали няма да се подобри бързодействието, ако нишките не се пускат накуп, а се пускат „на порции“ (по брой – броят на ядрата на процесора на вашия компютър) и съответно се изчакват.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 13

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: На Дунав мост станала поледица в комбинация с авария в системата за обслужване на гранична полиция. Образували се големи опашки от по над 200 камиона и автомобили и от двете страни на границата. Гранична полиция позволявала на не повече от 20 превозни средства в една посока да се намират на моста по едно и също време, както и не повече от 30 превозни средства общо за двете посоки. Напишете програма, която организира опашката от превозни средства и от двете страни и следи за спазване на правилото за преминаване. Погрижете се преминаването да бъде осъществено така, че да бъде честно и за двете страни – не е хубаво опашката от едната страна да намалява много по-бързо спрямо другата.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 14

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Дадена е матрица с размерност $N \times N$. Матрицата е инициализирана с произволни цели числа от 0 до 9. Задачата ви е да изчислите броя на срещанията на всяка една цифра, като работата трябва да се се раздели между K на брой нишки.

Идеята е да разделите матрицата на K на брой области – редове на матрицата. Удачно е да се подбере такова число K , което да е делител на N , но не е задължително. Всяка нишка пресмята броя на срещанията на всяка цифра в своята област и обновява глобалните за всички нишки статистики.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 15

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Студент:

Ф№: Група:

Задание: В парка Дисниленд са дошла голяма група от N на брой деца. Трябва да се създаде програма моделираща „vlakche на ужасите“. Всяко vlakche има капацитет от максимум C деца. C променливата $numTurns$ се определя по колко пъти се върти едно vlakche по трасето преди да спре. C $turnTime$ се определя броя секунди, които изисква всяка обиколка на трасето.

Всяко дете иска да се вози на vlakчето, то е основна атракция, но в парка има и много други забавления. Затова след влизането си в парка, детето първо ще се занимава с други забавления. Тези други забавления ще отнемат произволен брой секунди – между 0 и 10 (използвайте `Math.Random()`, за да генерирате произволно число). В момента, в който свършат забавленията, детето отиде да се качи на vlakчето.

Vлакчето ще потегля тогава, когато се запълни капацитета му, или по изключение с определен брой празни места, но в рамките на 5 секунди е установено, че не се е качил никой. Ако на дадено дете му е дошъл реда да се вози, но vlakчето вече е тръгнало, то ще се нареди на опашка и ще изчака.

Програмата трябва да дава статистика какво се случва. Очаквания изход ще бъде нещо от сорта на:

- Vлакчето се запълни, започва първо возене, първа обиколка, останаха да чакат ... деца;
- Vлакчето направи първа обиколка, на опашка чакат ... деца;
- Vлакчето направи втора обиколка, на опашка чакат ... деца;
- ...
- Vлакчето направи C обиколка, завърши първо возене;
- Vлакчето се запълни, започва второ возене, първа обиколка, останаха да чакат ... деца;
- ...

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 16

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Трябва да моделирате играта „камък-ножица-хартия“ с помощта на три нишки. Всяка нишка стартира и си избира произволно „камък“, „ножица“ или „хартия“. Условието е, че „камъка бие ножицата“, „ножицата бие хартията“, а „хартията бие камъка“. Ако една нишка победи другите две, тя получава 2 точки. Ако две нишки са равни, но бият трета, те си делят по 1 точка. Ако и трите нишки са равни, не получават точки.

Съвсем нормално е една нишка да направи избора си първа. Тогава тя трябва да изчака другите две да свършат и чак тогава да се направи съпоставка за резултат. След като резултата бъде изчислен, нишките трябва да продължат и да изиграят следваща игра. Процесът се повтаря N на брой пъти, т.е. трябва да моделирате N на брой игри. Извеждайте в конзолата междинните резултати, както и в края изведете общия резултат.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 17

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Трима души решили да залагат пари на „кой ще хвърли най-много пъти ези с монета“. Всеки извадил по една монета и започнал да я хвърля. Ако някой хвърли тура, той спира и брой колко пъти му се е паднало ези. Този, който е натрупал най-много пъти ези, печели.

Разбира се може да се случи така, че двама, та дори е възможно и тримата, да са завършили с равен брой ези-та. В такъв случай те продължават да играят до намиране на победител.

След като свършили първата игра, решили да изиграят втора. После трета и т.н. Накрая пресметнали общия си резултат от всички игри.

Моделирайте играта за N игри.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 18

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Студент:

Ф№: Група:

Задание: Искаме да направим някакъв елементарен прототип на торент. Намерете някой файл с големина 1-2 мегабайта. Заредете го целия в паметта (например го прочетете и заредете в масив). Към вашата програма се присъединяват N на брой нишки, всяка от които желае да получи този файл. Вие обаче не желаете да го пращате на всички, а искате да им дадете по една частичка от файла на всяка, а те самите да си ги изтеглят помежду си. Затова процедурите по следния начин:

1. Първата нишка извиква вашата функция "giveMeData()" и вие и връщате първата 1/N част от файла;
2. Втората нишка извиква "giveMeData()" и вие връщате втората 1/N част от файла;
3. ...

Всяка нишка знае коя част от файла е взела, затова си я подрежда в нейния си масив където ѝ е мястото. Така първоначално всяка една от N-те нишки ще има по 1/N част от файла.

След като са получили първите си частички, нишките започват да си ги разменят помежду си. Самите нишки обаче не са стиснати както главната програма – те ще изпратят всичката информация, която имат до този, който я е поискал.

Програмата завършва тогава, когато всяка нишка докаже, че е получила всичко, при това коректно. Сравнението дали е коректно или не се прави като се сравнява получения масив с оригиналния масив от главната програма. Ключови моменти, които биха предизвикали затруднение (общи съвети):

1. Коя нишка, коя ще вика. Нишките се познават помежду си, но не знаят коя каква информация притежава. Възможно е да се получат десетки прехвърляния на ненужна информация. Затова е добре да помислите за някаква синхронизация. Например може да реализирате функция "getParts()", с която нишката ще върне информация какви са текущите парчета, които притежава. След това „getPart(int)“ може да връща конкретно парче. Това е само препоръка, за да се минимизира прехвърлянето на ненужни данни;
2. Помислете за централен регистър на „кой какво притежава“, за по-голяма ефективност.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 19

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: В среда от идеален вакуум, от една тръба пускате множество водородни атоми "H", а от друга множество кислородни атоми "O", и двете наследници на абстрактния клас "atom". Както знаем, при сблъсъка на тези две множества атоми, очакваме да се образуват водни молекули "H₂O". За всеки един от атомите пуснете по една нишка.

Всеки един от пуснатите през тръбите атоми „ще се помотае“ за известен период от време, преди да се свърже в молекула - нека това време е произволно от 1 до 5 секунди (използвайте Math.Random(), за да генерирате произволно число). След това той ще уведоми основната нишка "MoleculeMaker", която се използва за създаване на молекули, като извика метода й "IAmReady(atom a)" и се предаде сама себе си като нейн параметър.

Методът IAmReady(atom a) от своя страна пази списък с водородни и списък с кислородни атоми. Когато установи, че е получил достатъчно водородни (2) и достатъчно кислородни (1) атоми, той ги премахва от списъка и ги изпраща като готова молекула, а в конзолата отпечатва съобщение за успех.

Процесът продължава до прихващане на всички възможни атоми. Накрая програмата трябва да отпечата колко молекули е създала, и колко атоми (и от кой вид) са останали неупотребени.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 20

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: И тази година ще се проведе традиционният национален оффроуд събор. Този път решили да е в малко китно селце в Родопите. Хубаво го решили, но... там няма инфраструктура за толкова много автомобили. И станало голямото задръстване при един малък мост. Едни коли искали да минат в едната посока, други в другата.

Мостчето е малко и доста нестабилно – по него може да мине не повече от един джип. Съответно била създадена организация на движение – минава една кола от юг, после една от север, после пак от юг и пак от север, и т.н. Тоест по условие имате две опашки от автомобили от двете страни на моста. Организирайте ги в две нишки, които се уведомяват една друга, че може да се продължи. Преминаването на един автомобил по моста отнема точно 2 секунди.

Част от шофьорите обаче били с много добре подготвени коли и високо самочувствие. Когато те достигнат първа линия (т.е. готови са да преминат моста), те не желаят да чакат много. Ако моста за тях е свободен, те го преминават. Но ако по моста върви друга кола и те чакат повече от 1 секунда, те ще решат да се направят на мъже и преминават с джипката си направо през реката.

Моделирайте задачата, като отпечатвате в конзолата какво се случва при всяко предприето действие.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 21

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание: Станал масов студентски купон на къмпинг за 8 декември! Всичко хубаво, но се оказало, че банята е само една и е обща. А след големия купон вечерта в дискотеката, всички искали да си вземат по един душ и да се изкъпят преди да си легнат.

Банята е с капацитет N . Мъжете са K на брой, а жените са L на брой. Наредили се на опашка пред банята. Целта ви е да изкъпете всички на порции по N човека, като в един момент може да има само по един вид хора – мъже или жени – вътре.

Отначало мъжете решили да са кавалери и да пуснат жените да се къпят първи, но част от жените се оказали феминитски и наложили строг ред, с който честно ще се спазват всички европейски норми за равнопоставеност между половете. Идеята е следната:

- Ако в даден момент мъжете на опашката са повече от жените, то следващия път ще влязат мъже в банята;
- Обратно, ако в даден момент жените на опашката са повече от мъжете, то следващия път ще влязат жени в банята.

Къпането е абстрактно, затова е бързичко – отнема по 1 секунда на група от N човека.

Моделирайте задачата, като съставите нишки – една нишка за мъжете и една за жените. Те трябва да се синхронизират с централния метод „къпане“ и да се изчакват помежду си.

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 22

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание:

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 23

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание:

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 24

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание:

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 25

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание:

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 26

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание:

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 27

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание:

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 28

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание:

.....
хон. ас. Боян Петров

Домашно задание №3

Вариант 29

Дисциплина: Програмиране и използване на компютри (ПИКЗ)

Студент:

Ф№: Група:

Задание:

.....
хон. ас. Боян Петров