Задания к ЛР№5\_потоки

|  |  |
| --- | --- |
| 1 | **Крестики – нолики**. Вариант графический. Сообщение о выигравшем.  **Гонки** – счет. 5 участников ведут счет до 10. По окончании должно выдаваться консольное сообщение каждым об окончании счета и имя потока-победителя. |
| 2 | **Гонки**. 3 участника наперегонки наращивают каждый свой ProgressBar на разное случайное число единиц прогресса. По окончании должно появиться сообщение о том, кто прибыл первым, кто последним. Вариант графический.  **Крестики – нолики**. Упрощенный вариант – консольный. Должны выдаваться сообщения: 1-й поток окончил! 2-й поток окончил! И имя выигравшего потока, если была выигрышная комбинация. |
| 3 | **Спорщики.** Три участника путешествия наперебой выбирают маршрут (суша, море, воздух). По окончании объявляется победитель и выбранный способ перемещения.  **Слова.** Пять потоков в строгой очередности считывают слова из одного и того же файла (Поток1: слово1, слово5, слово9; Поток2: слово2, слово6, слово10 и т.д. )в свой массив сло, затем выводят в консоль почитанное: сначала все слова потока1, затем потока 2, и т.д. |
| 4 | **Лото.** Четыре игрока строго по очереди выставляют по одной свей фишке (у каждого свой цвет фишки) на общую доску, раскрашенную в разные цвета, о чем каждый выдает сообщение, например: «Красный, фишка 3». По окончании (когда все клетки заполнены) сообщается, кто закончил первым. Фишек по 10 у каждого, мест на доске 30. Расположение цветов на доске случайное. (Графический вариант).  **Лото1.** Предыдущая задача, но порядок игроков и размещение фишек случайное, у каждого из 4-х игроков фишки одного цвета. Есть графическое поле, на которое все игроки должны попытаться выставить свои 10 фишек в случайные места. Но мест всего 30. Побеждает тот, кто выставит больше |
| 5 | **Гонка.** Несколько потоков (машинок) старутют одновременно. Но скорости возрастают случайным образом. На финише должен быть объявлен победитель. Можно консольно.  **Автостоянка.** Доступно несколько машиномест. На одном месте может находиться только один автомобиль. Если все места заняты, то автомобиль не станет ждать больше определенного времени и уедет на другую стоянку, грфический вариант лучше. |
| 6 | Графический **тренажер клавиатуры**. Буквы движутся по горизонтали слева-направо в 3-х потоках на разных уровнях горизонта. Пользователь нажимает соотвветсвующую букву, если успевает за время пересечения буквой поля, счет игрока нарасчивается.  **Map-Reduce.** Главный поток рассылает 3-м другим строки, задача каждого потока составить hashmap – слово-сколько раз оно встерачется в строке. Затем главный поток должнен собрать и агрегировать результат. |
| 7 | **Produser-Consumer.** Добавьте еще два объекта Consumer, которые будут запускаться тоже отдельным потоком.  Выводите на консоль какой из объектов Consumer обработал объект из очереди.  Создать **массив**, заполненный случайными числами, на 500 элементов. На графическом интерфейсе создать textBox1 и textBox2, в один текст бокс вывести первую часть массива а во второй вторую часть массива. Выполнить с помощью потоков. |
| 8 | Создать **3 функции**. Главную и 2 второстепенных (побочных). В определенный момент из главной во второстепенные передается число n и i. Один из побочных потоков считает n! - чисилтель, другой - Σ(n+i), i=[0,n]-знаменатель. Главный поток должен по окончании их работы вычислить результат деления. Дальше 2 функции продолжают свою работу одновременно и выводят по очереди символы А и В n раз.  **Самокаты.** На стоянке 10 мест, которые пополняются каждые 4 часа. Но каждые 10 минут подходит новый пользователь и берет самокат. Смоделировать задачу и определить, как часто надо пополнять стоянку, чтобы она не пустовала. |
| 9 | **Заводской цех.** В цеху есть 5 токарных станков. Каждые 10 минут к каждому из них подходит новый рабочий и обрабатывает свою деталь случайное количество времени. Смоделировать задачу с помощью потоков и подсчитать главным потоком сколько деталей было обработано за сутки на станках.  **Кубики.** У вас есть три поля, в которых находятся по 9 разноцветных кубиков, расположенных произвольно в вертикальной цепочке. Каждому полю соответствует поток для упорядочения кубиков по цветам (красный, желтый, зеленый). Каждому потоку выдается время для выполнения одного шага перестановки. В конце объявляется победитель – кто первым выстроил кубики. |
| 10 | **Коршеринг.** В службе коршеринга 30 автомобилей. Каждый час приходит клиент - новый поток - и берет машину на срок от 4 до 48 часов (время определяется случайно и время проката всегда четное число). Через оговоренный срок машину возвращают. Подсчитать сколько клиентов вопспользовались машинами за сутки. Вывести статистику сколько машин брались на каждый инервал времени: 4, 6, 8… 48 часов.  Создать главный поток и два рабочих, где: главный ждет первого рабочего с join(), принимает число (некторорый результат расчета) от первого и передает его второму. Далее второй и первый выполняют на «кто быстрее» одинаковые задания с этим числом, причем Второй рабочий использует yield(), а Первый рабочий использует sleep(). |

***Требования к отчету к ЛР №5***

1. Постановка задачи. Описание решения.
2. UML-диаграмма классов (не забудьте о вложенных классах, если они есть).
3. Временная диаграмма работы потоков (фрагмент).
4. Текст программы с комментариями. Классы должны находиться в разных файлах.
5. Скриншоты исполнения программы.
6. Все задачи выполняются с только на основе изученных методов (sleep, yield, join, wait, notify, synchronized).