

Лабораторная работа №4. (8 часов: 27.03.17 – 24.04.17)

Тема: Синхронизация потоков. Разработка классов, безопасных для потоков.

Задача. Написать класс `SyncArray` для управления доступом параллельных потоков к контейнеру элементов тип `int`, который организован как массив.

Требования к реализации:

1. Интерфейс класса `SyncArray` разработать самостоятельно.
2. Для доступа параллельных потоков к элементам массива можно использовать **только** перегруженные операторы индексирования `[]`.
3. Размер массива задается в конструкторе.
4. Элементы массива могут находиться в одном из двух состояний: «свободен» или «занят»:
 - 4.1. Элемент считается свободным, если из него прочитано целое число;
 - 4.2. Элемент массива считается занятым, если в него записано целое число.
5. После создания массива все его элементы свободны.
6. Параллельные потоки могут одновременно читать и записывать целые числа в элементы массива:
 - 6.1. Если поток читает целое число из *пустого* элемента массива, то он зависает до тех пор, пока этот элемент не станет *занятым*;
 - 6.2. Если поток записывает целое число в *занятый* элемент массива, то он зависает до тех пор, пока другой поток не *освободит* этот элемент.
7. Все объекты синхронизации доступа к элементам массива определяются как **атрибуты** класса `SyncArray`.
8. Синхронизация доступа параллельных потоков к элементам массива выполняется **только** внутри методов класса `SyncArray`.
9. Для отладки класса `SyncArray` в каждом его методе должна быть установлена **задержка** в 7 мс. (`Sleep`).

Дополнительные требования:

1. Для тестирования класса `SyncArray` написать программу для консольного процесса, который состоит из потока `main` и нескольких потоков `tester`.
2. Поток `main` должен выполнять следующие действия:
 - создать объект типа `SyncArray`, размер массива вводится пользователем с консоли;
 - ввести с консоли количество потоков `tester`, которые должен создать поток `main`;
 - создать требуемое количество потоков `tester`, каждому из которых передать в качестве аргумента адрес массива типа `SyncArray`;
 - ждать сигнал на готовность к работе от всех потоков `tester`;
 - подать сигнал на начало работы потоков `tester`;
 - завершить свою работу после окончания работы всех потоков `tester`.
3. Поток `tester` должен выполнять следующие действия:
 - ввести с консоли:
 - индекс тестируемого элемента массива;
 - количество итераций с этим элементом;
 - целое число, которое поток будет записывать в данный элемент;
 - подать сигнал на готовность к работе;
 - ждать сигнал на начало работы;
 - для заданного элемента массива выполнить заданное количество следующих итераций:
 - записать в элемент массива заданное целое число;
 - вывести на консоль сообщение: "В элемент [N] записано число M";
 - поспать 7 мс.;
 - прочитать значение элемента массива;
 - вывести на консоль сообщение: "\tИз элемента [N] прочитано число M";
 - поспать 7 мс.;
 - завершить свою работу.