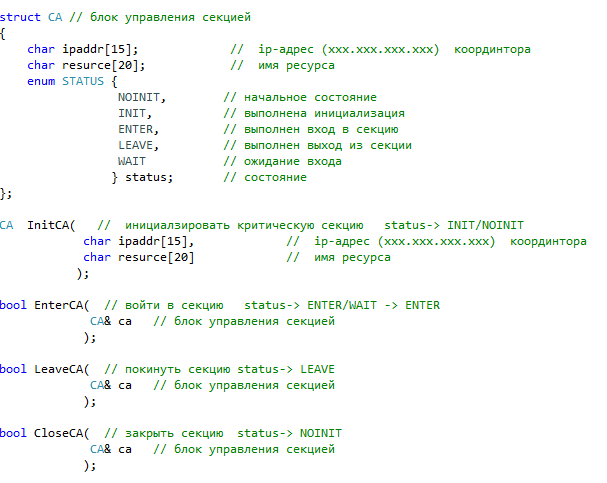
Лабораторная работа 2

РИС, ФИТ, III курс

6 часов

**Взаимное исключение (I)**

1. Используйте материал лекции 04 (/Для\_студентов\_ФИТ\_БГТУ/ПРЕПОДАВАТЕЛИ/Тимонович/РИС/Материалы и книги/РИС Смелов 2017), а также раздел 5.5 из книги (/Для\_студентов\_ФИТ\_БГТУ/ПРЕПОДАВАТЕЛИ/Тимонович/РИС/Материалы и книги/Таненбаум\_распределенные системы.pdf)
2. Используйте параллельный сервер, разработанный в курсе ПСП.
3. Разработайте координатор (UDP-сервер), реализующий централизованный механизм взаимного исключения.
4. Спецификация функций, интерфейса клиента представлена на следующем рисунке

****

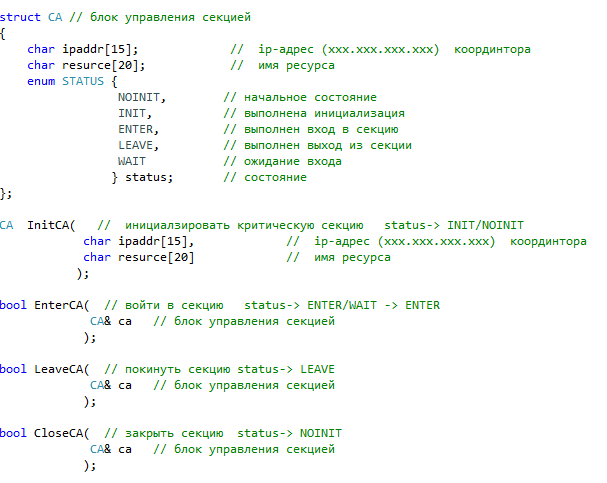
1. Продемонстрируйте работу координатор, обслуживающего запросы двух клиентов. В качестве критического ресурса используйте текстовый файл, расположенный на общем сетевом ресурсе двух клиентов (на каждом клиентском компьютере необходимо добавить общий сетевой диск). Клиенты должны дописывать в файл отметки времени - 5 строк с интервалом в 5 сек.

1. Координатор, должен отображать диагностические сообщения об инициализации и удалении критической секции, о входе в секцию и о выходе из нее.

1. При демонстрации одновременно запускаются два клиента (A и B), которые пытаются дописать строки в общий файл, расположенный на сетевом диске (критический ресурс). Один из клиентов (например, B) должен перейти в режим ожидания (приостановиться на функции EnterCA), а другой клиент (A) должен успешно выполнить запись в файл. После того, как запись в файл клиентом A будет завершена (выполнит функцию LeaveCA), клиент B должен выполнить запись своей порции информации.

**API для работы с централизованным файловым хранилищем (II)**

1. Разработайте API DFS для работы с централизованным файловым хранилищем. API должно базироваться на API, разработанном в 1ой части лабораторной работы.
2. Продемонстрируйте, разработанное API DFS: запись 10 строк в файл, чтение и вывод на консоль 10 записей.
3. Продемонстрируйте совместную работу нескольких процессов, записывающих и считывающих данных.



1. Ответьте на следующие запросы:
2. обеспечивает API DSF строгую непротиворечивость хранилища? если нет, что необходимо сделать, чтобы это выполнялось?
3. обеспечивает API DSF последовательную непротиворечивость хранилища? если нет, что необходимо сделать, чтобы это выполнялось?
4. обеспечивает API DSF причинную непротиворечивость хранилища? если нет, что необходимо сделать, чтобы это выполнялось?