

Задание 01. Windows

1. Напишите ассемблерный код с применением команд BTS или BTR, демонстрирующий реализацию механизма синхронизации двух потоков одного процесса и поясните его работу.

Задание 02. Windows

2. Разработайте приложение **OS07_02**, запускающее два дочерних потока **A** и **B**.
3. Все потоки выполняют циклы в 90 итераций, выводящие имена потоков и номера итерации с задержкой в 0.1 сек.
4. Приложение **OS07_02** синхронизирует выполнение потоков **main**, **A** и **B** с помощью механизма **critical section**.
5. Синхронизация должна обеспечивать поочередное выполнение итераций цикла с 30 по 60 в каждом потоке.

Задание 03. Windows

6. Разработайте приложение **OS07_03**, запускающее два дочерних процесса **OS07_03A** и **OS07_03B** с выводом в отдельные консоли.
7. Все процессы выполняют циклы в 90 итераций, выводящие имена процессов и номера итерации с задержкой в 0.1 сек.
8. Приложение **OS07_03** синхронизирует выполнение процессов **OS07_03**, **OS07_03A** и **OS07_03B** с помощью механизма **mutex**.
9. Синхронизация должна обеспечивать поочередное выполнение итераций цикла с 30 по 60.

Задание 04. Windows

10. Разработайте приложение **OS07_04**, запускающее два дочерних процесса **OS07_04A** и **OS07_04B** с выводом в отдельные консоли.

11. Все процессы выполняют циклы в 90 итераций, выводящие имена процессов и номера итерации с задержкой в 0.1 сек.
12. Приложение **OS07_04** синхронизирует выполнение процессов **OS07_04**, **OS07_04A** и **OS07_04B** с помощью механизма **semaphore**.
13. Синхронизация должна обеспечивать поочередное выполнение итераций цикла с 30 по 60 одного (любого) процесса и двух других процессов. Другими словами, итерации с 30 по 60 должны одновременно выполняться только в двух из трех процессов.

Задание 05. Windows

14. Разработайте приложение **OS07_05**, запускающее два дочерних процесса **OS07_05A** и **OS07_05B** с выводом в отдельные консоли.
15. Все процессы выполняют циклы в 90 итераций, выводящие имена процессов и номера итерации с задержкой в 0.1 сек.
16. Приложение **OS07_05** синхронизирует выполнение процессов **OS07_05**, **OS07_05A** и **OS07_05B** с помощью механизма **event**.
17. Синхронизация должна обеспечивать выполнение приложения в два этапа:
 - 1) выполнение итераций с 1 по 15 процесса **OS07_05**;
 - 2) одновременное выполнение всех трех процессов:
OS07_05 – продолжает выполнение итераций; процессы **OS07_05A** и **OS07_05B** выполняются начиная с первой итерации.

Задание 06. Linux

18. Разработайте приложение **OS07_06**, запускающее два дочерних потока **A** и **B**.
19. Все потоки выполняют циклы в 90 итераций, выводящие имена потоков и номера итерации с задержкой в 0.1 сек.
20. Приложение **OS07_06** синхронизирует выполнение потоков **main**, **A** и **B** с помощью механизма **mutex**.

21. Синхронизация должна обеспечивать поочередное выполнение итераций цикла с 30 по 60 в каждом потоке.

Задание 07. Ответьте на следующие вопросы

22. Дайте определение понятию «синхронизация потоков».

23. Объясните понятие «взаимная блокировка».

24. Перечислите механизмы авторизации OS.

25. Поясните в чем разница между механизмом **mutex** и **semaphore**.

26. Почему **mutex**, **semaphore**, **event** создают объект ядра OS, а **critical section** нет.