

## Домашнее задание №8 Плюскова Н.А.

### 1. Что есть «состояние гонки» и к чему оно может привести?

Ошибка программирования многозадачной системы, при которой работа системы зависит от того, в каком порядке выполняются части кода. Состояние гонки возникает тогда, когда несколько потоков многопоточного приложения пытаются одновременно получить доступ к данным, причем хотя бы один поток выполняет запись. Состояния гонки могут давать непредсказуемые результаты, и зачастую их сложно выявить. Иногда последствия состояния гонки проявляются только через большой промежуток времени и в совсем другой части приложения. Основными последствиями могут быть:

- утечки памяти
- ошибки сегментирования
- порча данных
- уязвимости
- взаимные блокировки
- утечки других ресурсов, например, файловых дескрипторов

### 2. Как используются мьютексы и условные переменные?

Взаимные исключения - mutual exclusion или mutex - и условные переменные - conditional variables - являются основными инструментами синхронизации. Они применяются как правило для потоков в рамках одного родительского процесса. Mutex используется для блокировки, а условная переменная - для ожидания. Условные переменные позволяют блокировать один или более потоков, пока либо не будет получено уведомление от другого потока, либо не произойдет spurious wakeup («ложное/случайное пробуждение»). Mutex предназначен для организации взаимоисключающего доступа к общим данным для нескольких потоков с использованием барьеров памяти.

### 3. На что влияет выбор гранулярности блокировки?

Гранулярность блокировки влияет на то, доступ к какому количеству данных поток получает эксклюзивно. Например, можно заблокировать узел в стеке - это мелкогранулярная блокировка, которая обеспечивает истинный параллелизм, а можно заблокировать весь стек одним мьютексом так, что только один поток может выполнять операция над стеком в данный момент времени.

### 4. Когда возникает взаимоблокировка и как ее предотвратить?

Взаимоблокировка — ситуация, при которой несколько процессов находятся в состоянии ожидания ресурсов, занятых друг другом, и ни один из них не может продолжать свое выполнение.

Если взаимная блокировка возникает из-за того, что два процесса обращаются к одним и тем же ресурсам, но в разном порядке (как это описано в начале статьи), то достаточно поменять порядок блокировки ресурсов. Основной способ её предотвращения— разработка иерархии блокировок: между блокировками устанавливается отношение сравнения и вводится правило о запрете захвата «большой» блокировки в состоянии, когда уже захвачена «меньшая».

### 5. Что есть атомарная операция и атомарный тип данных?

Атомарная операция - неделимая операция типа R, W RMW, которая либо выполнена, либо не выполнена. Атомарные типы данных - такие типы, что любые операции над ними атомарны.