Билет №13

Дайте определение понятию “синхронизация потоков” и опишите как взаимодействуют программы, процессы и потоки?

Синхронизация потоков – это настройка взаимодействия потоков между собой.

Делается это с помощью мьютекса.

Задача мьютекса — обеспечить такой механизм, чтобы доступ к объекту в определенное время был только у одного потока. Если Поток-1 захватил мьютекс объекта А, остальные потоки не получат к нему доступ, чтобы что-то в нем менять. До тех пор, пока мьютекс объекта А не освободится, остальные потоки будут вынуждены ждать.

Пример из жизни: представь, что ты и еще 10 незнакомых людей участвуете в тренинге. Вам нужно поочередно высказывать идеи и что-то обсуждать. Но, поскольку друг друга вы видите впервые, чтобы постоянно не перебивать друг друга и не скатываться в гвалт, вы используете правило c «говорящим мячиком»: говорить может только один человек — тот, у кого в руках мячик.

Оператор synchronized.

Им помечается определенный кусок нашего кода. Если блок кода помечен ключевым словом synchronized, это значит, что блок может выполняться только одним потоком одновременно.

Синхронизацию можно реализовать по-разному. Например, создать целый синхронизированный метод:

|  |
| --- |
| public synchronized void doSomething() {  //...логика метода  } |

Или же написать блок кода, где синхронизация осуществляется по какому-то объекту:

|  |
| --- |
| public class Main {  private Object obj = new Object();  public void doSomething() {  //...какая-то логика, доступная для всех потоков  synchronized (obj) {  //логика, которая одновременно доступна только для одного потока  }  }  } |

Смысл прост. Если один поток зашел внутрь блока кода, который помечен словом synchronized, он моментально захватывает мьютекс объекта, и все другие потоки, которые попытаются зайти в этот же блок или метод вынуждены ждать, пока предыдущий поток не завершит свою работу и не освободит монитор.

Взаимодействие между процессами

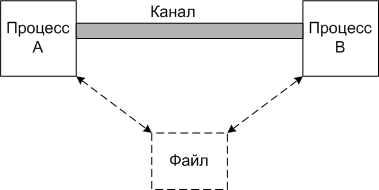
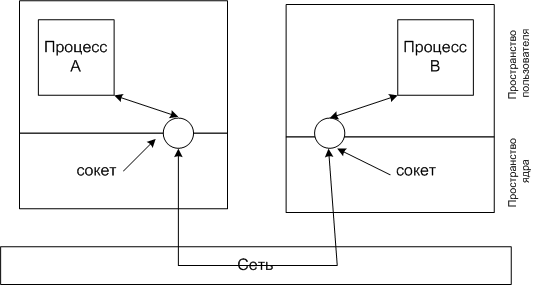
Ситуации, когда приходится процессам взаимодействовать:

Передача информации от одного процесса другому

Контроль над деятельностью процессов (например: когда они борются за один ресурс)

Согласование действий процессов (Например: когда один процесс поставляет данные, а другой их выводит на печать. Если согласованности не будет, то второй процесс может начать печать раньше, чем поступят данные).

Два вторых случая относятся и к потокам. В первом случае у потоков нет проблем, т.к. они используют общее адресное пространство.

Потоки