

1. Для каждого из перечисленных утверждений укажите, является ли оно свойством безопасности (safety), или живости (liveness). Объясните.

Safety – ничего плохого не происходит. При разных значениях, два процесса никогда не договорятся между собой.

Liveness – когда-нибудь происходит что-то хорошее

1. Почтальон всегда звонит дважды.
 - Safety – почтальон звонит дважды чтобы кто то точно ответил.
2. Нет худа без добра.
 - Liveness – есть две вещи – желание человека получить чего-то хорошее и плохие случаи, которые происходят в жизни человека. Пословица говорит о том, что в определенных обстоятельствах желание человека и результат плохого случая совпадут.
3. Кто ищет, тот всегда найдет.
 - Liveness – если долго искать, в определенный момент будет найдено успешное решение.
4. Никто не ожидает испанской инквизиции.
 - Safety – длительное время не наступает испанская инквизиция, поэтому ее никто не ожидает. Испанская инквизиция – плохо.
5. Я тебя никогда не увижу && я тебя никогда не забуду.
 - Safety – as long as я тебя не увижу, я тебя не забуду, и наоборот.
6. И все биндюжники вставали, когда в пивную он входил.
 - Safety – как только он входил, все биндюжники вставали (чтобы не случилось чего-то плохого)
7. Я закончу то, что ты начал.
 - Liveness – в какой-то момент в будущем я закончу этот процесс.
8. Прерывание приводит к генерации сообщения в течение одной секунды.
 - Safety – в течении одной секунды будет обязательно сгенерировано сообщение.
9. Прерывание приводит к генерации сообщения.
 - Liveness – к определенному моменту в будущем будет сгенерировано сообщение.
10. Отсутствие дедлоков – два процесса захватили объекты и ждут пока другой даст знак что он освободит объект чтобы поменяться друг с другом.
 - Liveness - если два процесса просто повиснут, ожидая друг друга, ничего плохого в локальном смысле происходить не будет.

2. В решении проблемы Producer-Consumer мы подразумевали, что Боб видит состояние баночки на подоконнике Алис. Предложите протокол, использующий банки и нитки, который бы не требовал видимости банки на чужом подоконнике (так работают прерывания на самом деле).

- Альтернатива видимости банки – если buffer полный, то producer засыпает, если buffer пустой – засыпает consumer. Таким образом потоки зависят не от друг друга, а от еще одного объекта, что позволяет им не попасть в deadlock.

3. Пусть запуск некоторой программы на 2х процессорах дает ускорение S2. Используя закон Амдала, выведите ускорение этой программы Sn при запуске на n процессорах.

- $$S_n = \frac{1}{\alpha + \frac{1-\alpha}{n}} = \frac{n}{\alpha(n-1)+1}$$

4. Выбирая между покупкой одного процессора, выполняющего 5 зиллионов операций в секунду, и покупкой 10 процессоров, каждый из которых выполняет 1 зиллион операций в секунду, что вы выберете для работы конкретного приложения? Используйте закон Амдала.

- Закон Амдала гласит - Если задача разделяется на несколько частей, суммарное время её выполнения на параллельной системе не может быть меньше времени выполнения самого медленного фрагмента.

- 1ый случай – допустим 95% программы является последовательными расчетами, тогда более оптимально выбрать 1 процессор
- 2ой случай – 40% программы является последовательными расчетами, тогда более оптимально выбрать 10 процессоров.