

Лабораторная работа №4. Варианты

Вариант 1:

Необходимо решить проблему "Санта Клаус" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- Санта все время спит, пока его не будет либо все его олени (число оленей — N), либо K из его M эльфов.
- Если его будят олени, он впрягает их в сани, развозит игрушки в течение какого-то случайного времени, после чего распрягает их и отправляет погулять.
- Каждый олень гуляет случайное время.
- Если его будят эльфы, он проводит их по одному в свой кабинет, обсуждает с ними новые игрушки, а потом по одному отпускает.
- Время обсуждения равно T. Каждый эльф после обсуждения с Сантой занимается своими делами случайное время.
- Санта должен сначала поехать с оленями, если и олени и эльфы ждут его у дверей.

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

```
...
12:01:10.353 Санта спит.
12:01:11.412 Эльф 5 подошел к двери. Ожидающих эльфов: 1.
    Ожидающих оленей: 0.
12:01:11.700 Эльф 1 подошел к двери. Ожидающих эльфов: 2.
    Ожидающих оленей: 0.
12:01:11.701 Олень 7 подошел к двери. Ожидающих эльфов:
2. Ожидающих оленей: 1.
12:01:11.910 Олень 8 подошел к двери. Ожидающих эльфов:
2. Ожидающих оленей: 2.
12:01:11.815 Олень 9 подошел к двери. Ожидающих эльфов:
2. Ожидающих оленей: 3.
12:01:14.810 Эльф 3 подошел к двери. 3 эльфа будят Санту.
12:01:14.951 Санта пропускает в кабинет эльфа 1.
12:01:15.115 Санта пропускает в кабинет эльфа 5.
12:01:15.222 Санта пропускает в кабинет эльфа 3.
12:01:17.815 Олень 3 подошел к двери. Ожидающих эльфов:
```

0. Ожидающих оленей: 4.
12:01:17.900 Санта с эльфами начинают совещание.
...

Параметры N, M, K и T задаются пользователем.

Вариант 2:

Необходимо решить проблему "Переправа конкурентов" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- На реке есть паром, который может перевозить $2*N$ человек.
- Паром используют для переправы представители двух конкурирующих фирм. Если на пароме будут только сотрудники одной фирмы или равное число сотрудников с обеих сторон, то всё будет хорошо. Если же на нем будут представители обеих фирм в неравных количествах, то не миновать драки, поэтому такая ситуация недопустима.
- Люди приходят на переправу через случайные промежутки времени.
- На берегу представители всех фирм ведут себя культурно и переправляются в порядке очереди (первый пришел - первый переправился).
- Когда паром полностью заполнен, кто-то один из пассажиров должен нажать на кнопку, чтобы началась переправа. Переправа занимает случайное время.

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

...

12:01:10.353 Сотрудник Microsoft №1 пришел на переправу.
В пароме: 0 сотрудников Apple, 1 сотрудник Microsoft.
Очередь сотрудников Apple: 0. Очередь сотрудников Microsoft: 0.

12:01:11.412 Сотрудник Microsoft №2 пришел на переправу.
В пароме: 0 сотрудников Apple, 2 сотрудника Microsoft.
Очередь сотрудников Apple: 0. Очередь сотрудников Microsoft: 0.

12:01:11.700 Сотрудник Apple №1 пришел на переправу. В пароме: 1 сотрудник Apple, 2 сотрудника Microsoft.
Очередь сотрудников Apple: 0. Очередь сотрудников Microsoft: 0.

12:01:11.701 Сотрудник Microsoft №3 пришел на переправу.
В пароме: 0 сотрудников Apple, 2 сотрудника Microsoft.
Очередь сотрудников Apple: 0. Очередь сотрудников Microsoft: 1.

12:01:11.910 Сотрудник Microsoft №4 пришел на переправу.
В пароме: 0 сотрудников Apple, 2 сотрудника Microsoft.
Очередь сотрудников Apple: 0. Очередь сотрудников Microsoft: 2.

12:01:11.815 Сотрудник Apple №2 пришел на переправу. В пароме: 2 сотрудник Apple, 2 сотрудника Microsoft.
Очередь сотрудников Apple: 0. Очередь сотрудников Microsoft: 2.

12:01:14.810 Сотрудник Apple №1 скомандовал "Поехали!"

12:01:14.951 Сотрудник Microsoft №5 пришел на переправу.
Очередь сотрудников Apple: 0. Очередь сотрудников Microsoft: 3.

12:01:15.118 Паром приплыл обратно. В пароме: 0 сотрудников Apple, 3 сотрудника Microsoft. Очередь сотрудников Apple: 0. Очередь сотрудников Microsoft: 0.

...

Параметр N задается пользователем.

Вариант 3:

Необходимо решить проблему "Поиск-вставка-удаление" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- 3 типа нитей имеют доступ к общему связному списку: поисковики (N штук), вставщики (M штук) и удалятели (K штук).
- Поисковики могут просматривать список одновременно.
- Вставщики вставляют элементы в конец списка, поэтому только 1 из них может делать это в один момент времени. Его работа не мешает поисковикам осуществлять поиск в списке.
- Удалятели могут удалить элемент из любого места в списке, поэтому во время работы 1 удалятора никто другой не может иметь доступ к списку.
- Все нити ожидают завершения работы других нитей.
- Доступ нитей к списку по возможности должен осуществляться в порядке очереди.
- Время выполнения всех операций случайно.

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е.

последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

```
...
12:01:10.353 Поисковик №2 хочет начать поиск в списке.
12:01:11.412 Поисковик №2 начал поиск в списке.
12:01:11.700 Поисковик №2 закончил поиск в списке.
12:01:11.701 Удалятель №1 хочет начать удаление из
списка.
12:01:11.910 Удалятель №1 начал удаление из списка.
12:01:11.815 Поисковик №2 хочет начать поиск в списке.
12:01:14.810 Удалятель №3 хочет начать удаление из
списка.
12:01:14.951 Удалятель №1 закончил удаление из списка.
12:01:15.118 Поисковик №2 начал поиск в списке.
...
```

Параметр N, M и K задаются пользователем.

Вариант 4:

Необходимо решить проблему "Иммиграционная служба" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- В зале иммиграционной службы иммигранты принимают гражданство. Граждане могут присутствовать в помещении в качестве зрителей.
- Присяга принимается в присутствии судьи. Когда судья заходит в помещение, все иммигранты, которые находятся там, должны принять присягу. После этого судья выходит из помещения, после чего иммигранты могут получить свои справки у одной из N стоек, а затем идти. Пока иммигранты получают справки, судья не возвращается в помещение. Когда судья в помещении, никто не может входить туда, а иммигранты не могут выходить (граждане могут уходить).
- Зал вмещает M иммигрантов. Иммигранты заходят в зал в порядке очереди. Если иммигранты приходят в момент, пока судья принимает присягу, они должны ждать за дверью.

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

```
...
12:01:10.353 Иммигрант №10 зашел в здание. В зале: 5
```

иммигрантов, 3 зрителя.
12:01:11.412 Зритель №3 покинул зал. В зале: 5
иммигрантов, 2 зрителя.
12:01:11.700 Иммигрант №10 зашел в зал. В зале: 6
иммигрантов, 2 зрителя.
12:01:11.701 Судья зашел в зал и начал принимать присягу.
В зале: судья, 6 иммигрантов, 2 зрителя.
12:01:11.910 Иммигрант №11 зашел в здание. В зале: судья,
6 иммигрантов, 2 зрителя. В очереди: 1 иммигрант.
12:01:11.815 Зритель №13 зашел в здание. В зале: судья, 6
иммигрантов, 2 зрителя. В очереди: 1 иммигрант, 1
зритель.
12:01:14.810 Судья закончил принимать присягу и покинул
помещение. В зале: судья, 6 иммигрантов, 2 зрителя. В
очереди: 1 иммигрант, 1 зритель.
12:01:14.951 Иммигрант №11 зашел в зал. В зале: 7
иммигрантов, 2 зрителя. В очереди: 1 зритель.
12:01:15.118 Иммигрант №8 получает справку. В зале: 7
иммигрантов, 2 зрителя. В очереди: 1 зритель.
12:01:17.222 Зритель №13 зашел зал. В зале: 7
иммигрантов, 2 зрителя. В очереди: 1 зритель.
12:01:17.223 Иммигрант №6 получает справку. В зале: 7
иммигрантов, 3 зрителя.
12:01:17.523 Иммигрант №8 получил справку и покинул
помещение. В зале: 6 иммигрантов, 3 зрителя.
...

Параметр N задается пользователем.

Вариант 5:

Необходимо решить проблему "Детская комната" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- Родители приводят детей в комнату для игр: по одному ребенку на родителя. После привода ребенка родитель может уйти на какое-то время, после чего вернуться и забрать его.
- По требованиям безопасности в комнате всегда должен находиться минимум 1 родитель на каждых N детей.

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

...

12:01:10.353 Родитель №1 привел ребенка. В комнате: ребенок №1, родитель №1.

12:01:11.412 Родитель №2 привел ребенка. В комнате: дети №1,2, родители №1,2.

12:01:11.700 Родитель №2 ушел. В комнате: дети №1,2, родитель №1.

12:01:11.701 Родитель №3 привел ребенка. В комнате: дети №1,2,3, родители №1,3.

12:01:11.910 Родитель №1 ушел. В комнате: дети №1,2,3, родитель №3.

12:01:11.815 Родитель №2 забрал ребенка. В комнате: дети №1,3, родитель №3.

12:01:14.810 Родитель №4 привел ребенка. В комнате: дети №1,2,3,4, родители №3,4.

12:01:14.951 Родитель №1 забрал ребенка. В комнате: дети №2,3,4, родители №3,4.

12:01:15.118 Родитель №4 ушел. В комнате: дети №2,3,4, родитель №3.

...

Параметр N задается пользователем.

Вариант 6:

Необходимо решить проблему "Обыск в общежитии" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- В комнате общежития проходят вечеринки. Комната вмещает сколько угодно студентов.
- Студенты приходят на вечеринку и уходят когда им вздумается.
- В комнату может также зайти комендант, если в ней минимум N студентов, чтобы прекратить слишком шумную вечеринку, или если там никого нет, чтобы совершить обыск.
- Пока комендант находится в комнате, никто не может заходить туда, но кто-угодно может уходить.
- Комендант не может уйти, пока в комнате есть хоть один студент.

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

...

12:01:10.353 Студент зашел в комнату и включил музыку, вечеринка началась. В комнате: 1 студент.

12:01:11.412 Студент зашел в комнату. В комнате: 2 студента.

12:01:11.700 Комендант прошел мимо комнаты и ушел по своим делам.

12:01:11.701 Студент зашел в комнату. В комнате: 3 студента.

12:01:11.910 Студент вышел из комнаты. В комнате: 2 студента.

12:01:11.815 Студент вышел из комнаты. В комнате: 1 студент.

12:01:14.810 Студент вышел из комнаты. В комнате: никого.

12:01:14.951 Комендант зашел в комнаты, чтобы совершить обыск. В комнате: комендант.

12:01:15.118 Студент хотел зайти в комнату, но там был комендант, поэтому он ушел. В комнате: комендант.

...

Параметр N задается пользователем.

Вариант 7:

Необходимо решить проблему "Американские горки" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- На американских горках есть N тележек, каждая из которых вмещает P человек.
- Люди приходят на аттракцион через случайные интервалы времени и ждут в очереди. Если на станции есть свободная тележка, она подается на посадку, когда в очереди набирается не менее P человек. Люди садятся в тележку в порядке очереди. После посадки она отправляется, и на ее место может стать следующая тележка.
- Когда тележка возвращается, из нее по очереди выходят люди, после чего она либо становится на посадку, либо ожидает в очереди, пока погрузятся предыдущие тележки. Если тележка приезжает на станцию, пока другая тележка еще не закончила выгрузку пассажиров, она ожидает, пока выгрузка закончится.
- На пути могут быть одновременно несколько тележек, но они не могут обгонять друг друга.
- Тележки едут по маршруту время T .

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

```
...
12:01:10.353 Тележки 1, 2, 3 стоят на станции.
12:01:11.412 Человек 1 пришел на станцию. В очереди 1
человек.
12:01:11.700 Человек 2 пришел на станцию. В очереди 2
человека.
12:01:11.701 Человек 3 пришел на станцию. В очереди 3
человека.
12:01:11.910 Тележка 1 подается на посадку.
12:01:11.815 Человек 1 садится в тележку 1. В очереди 2
человека.
12:01:14.810 Человек 4 пришел на станцию. В очереди 3
человека.
12:01:14.951 Человек 2 садится в тележку 1. В очереди 2
человека.
12:01:15.118 Человек 3 садится в тележку 1. В очереди 1
человек.
12:01:15.128 Тележка 1 отправляется.
...
```

Параметры N, P и T задаются пользователем.

Вариант 8:

Необходимо решить проблему "Обедающие философы" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- 5 философов сидят за столом. Они могут быть в 2-х взаимоисключающих состояниях: думать и есть. Между каждым философом лежит 1 вилка, а перед каждым — тарелка. Для того чтобы начать есть, философу нужно взять 2 вилки (левую и правую).
- Философ берет и кладет вилки в случайном порядке.
- Если со времени окончания последней еды философа прошло больше, чем T, и он не начал есть, то он умирает.

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

...
12:01:10.353 Философ 1 думает. Состояние вилок: 01100
12:01:10.389 Философ 5 взял левую вилку (№5). Состояние вилок: 01101
12:01:10.390 Философ 5 взял правую вилку (№1). Состояние вилок: 11101
12:01:10.391 Философ 5 начал есть.
12:01:11.800 Философ 3 закончил есть.
12:01:11.812 Философ 4 взял левую вилку (№4). Состояние вилок: 11111
12:01:11.813 Философ 4 не смог взять правую вилку (№5). Состояние вилок: 11111
12:01:11.813 Философ 4 положил левую вилку. Состояние вилок: 11101
12:01:11.815 Философ 2 положил левую вилку (№2). Состояние вилок: 10101
...

Параметр T задается пользователем.

Вариант 9:

Необходимо решить проблему "Читатели-писатели" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- Есть общий ресурс, из которого можно читать или писать.
- В системе есть N потоков читателей и M потоков писателей. Они работают так: в бесконечном цикле сначала спят случайный интервал времени, потом пытаются получить доступ на чтение/запись, после получения доступа выполняют эту операцию случайный интервал времени.
- Чтение может происходить несколькими нитями одновременно, но не больше K.
- Когда какая-то из нитей производит запись, остальные нити не должны иметь доступа к ресурсу.
- Если писатель пишет, никто другой не может иметь доступа к ресурсу.
- Если писатель хочет записывать, но в это время читатели читают, он должен ждать, пока читатели завершат чтение, но другие читатели уже не могут начинать чтение, пока писатель не начнет запись.

Результаты работы программы должны отображаться в консоли по мере их появления, как показано ниже. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности

операций, выполняемых программой.

```
12:01:10.353 Читатель 1 хочет читать. | Ресурс свободен.
12:01:10.354 Читатель 1 начал чтение. | Читатель 1
читает.
12:01:10.453 Читатель 2 хочет читать. | Читатель 1
читает.
12:01:10.456 Читатель 2 начал чтение. | Читатели 1, 2
читают.
12:01:11.000 Писатель 1 хочет писать. | Читатели 1, 2
читают.
12:01:11.001 Читатели 1 и 2 прерваны. | Ресурс свободен.
12:05:11.005 Писатель 1 начал запись. | Писатель 1 пишет.
12:06:30.300 Читатель 1 хочет читать. | Писатель 1 пишет.
Читатель 1 ожидает.
...
```

Параметры N, M и K задаются пользователем.

Вариант 10:

Необходимо решить проблему "Спящие парикмахеры" с использованием библиотеки PTHREAD с учетом следующих ограничений:

- В парикмахерской работает N парикмахеров, у каждого из которых есть 1 кресло для стрижки. Клиенты приходят к парикмахеру через случайные промежутки времени. Когда клиентов нет, парикмахеры спят в кресле. Когда приходит клиент он будит какого-то из парикмахеров и начинается стрижка.
- Если все парикмахеры стригут кого-то, клиент садится в одно из M кресел для ожидания. Если свободных мест нет, он становится у стены. Всего в парикмахерской может находиться K клиентов.
- Когда парикмахер заканчивает стрижку, он идет в кассу получать оплату. Касса одна, поэтому парикмахеры занимают ее в порядке прихода.
- Когда один из парикмахеров освобождается, он начинает стричь клиента из очереди, который ждет дольше всего или засыпает, если в очереди нет клиентов.
- Время стрижки клиента также является случайной величиной.

Результаты работы программы должны отображаться по мере их появления следующим образом. Вывод должен быть синхронизированным, т.е. последовательность строк в логе должна соответствовать последовательности операций, выполняемых программой.

...
12:01:10.353 Парикмахеры спят.
12:01:11.412 Клиент 1 пришел в парикмахерску. Очередь: 0
12:01:11.700 Клиент 1 будит парикмахера 2.
12:01:11.701 Клиент 1 садится на стрижку к парикмахеру 2.
Очередь: 0
12:01:11.910 Клиент 2 пришел в парикмахерскую. Очередь: 0
12:01:11.815 Клиент 2 будит парикмахера 1.
12:01:14.810 Клиент 3 пришел в парикмахерскую. Очередь: 0
12:01:14.951 Клиент 1 постригся у парикмахера 2 и идет
платить
12:01:15.115 Клиент 3 садится в кресло для ожидания.
Очередь: 1
12:01:15.222 Клиент 1 расплачивается и уходит.
12:01:11.701 Клиент 3 садится на стрижку к парикмахеру 2.
Очередь: 0
...

Параметры N, M и K задаются пользователем.