

Задача о Винни-Пухе или правильные пчёлы
Пояснительная записка

Исполнитель:

Студент группы БПИ193-1 ФКН НИУ ВШЭ

Гриценко Егор Андреевич

«16» декабря 2020 г.

Москва 2020

1. Введение

Программа выполнена в рамках первого микропроекта дисциплины “Архитектура вычислительных систем” второго курса бакалавриата программной инженерии ФКН НИУ ВШЭ.

2. Исходное условие 2-ого варианта задания микропроекта

В одном лесу живут n пчел и один медведь, которые используют один горшок меда, вместимостью N глотков. Сначала горшок пустой. Пока горшок не наполнится, медведь спит. Как только горшок заполняется, медведь просыпается и съедает весь мед, после чего снова засыпает. Каждая пчела многократно собирает по одному глотку меда и кладет его в горшок. Пчела, которая приносит последнюю порцию меда, будит медведя. Создать многопоточное приложение, моделирующее поведение пчел и медведя.

3. Описание допустимой среды выполнения программы и средств разработки

Программа может выполняться на ОС, основанных на GNU/Linux при наличии библиотеки pthread.

Программа написана на языке C; сборка программы осуществляется посредством GNU C Compiler (gcc).

4. Описание решения

Для медведя и для каждой пчелы было решено завести по отдельному потоку. Для медведя и для пчелы заведены 2 функции, зацикленно выполняющие присущие им действия; на основании этих функций производится вызов упомянутых потоков.

В код пчелы было решено, помимо прочего, добавить возможность задержки перед доставкой мёда, посредством функции sleep: иначе один из потоков, как показали тэсты, мог полностью вытеснить остальные, и симуляция получилась не очень достоверной: предполагается, что мёд приносит много разных пчёл, а не одна какая-то.

Операции над счётчиком, моделирующим горшок мёда, защищены spinlock’ом. Код пчёл, защищённый spinlock’ом, также служит для их остановки на момент пробуждения медведя: иначе медведь мог очень долго не просыпаться, если пчёл было достаточно много (скажем, 512). Помимо этого оказалось достаточно удобно встроить в оговариваемый участок кода и печать сообщений о действиях пчёл: таким образом, их вывод производится в порядке возрастания времени в них указываемого.

Изначально для моделирования сна медведя предполагалось использовать mutex и ожидание до сигнала (pthread_cond_wait) в его потоке о том, что горшок наполнился. Но отправка сигнала иногда осуществлялась быстрее, чем поток медведя успевал очистить переменную сигнала и уснуть: возникал stunlock. Поэтому был использован семафор. Соответственно, инкремент его счётчика, означающий «готовность» мёда, осуществляется в коде пчелы, попытка декремента — в коде медведя.

5. Описание интерфейса

Программа может принимать до 3-х опциональных аргументов. Первый аргумент интерпретируется как количество пчёл в интервале от 0 до 4096 (значение по умолчанию — 5) или просьба о выводе информации об использовании в случае несоответствии формата. Второй — как ёмкость горшка (по умолчанию — 200). Третий — как максимальная величина определяемого случайно времени (задержки в миллисекундах), которое пчела тратит на добавление порции мёда (по умолчанию — 0).

Окончание симуляции осуществляется посредством отправки программе SIGINT (также, как ping).

6. Компиляция и примеры команд запуска

Компиляция была произведена как:

```
gcc task.c -pthread -o task
```

Пример команд запуска:

```
./task          # симуляция с 5-ю пчёлами, горшком с вместимостью в 200 порций, без задержки
```

```
./task -h
```

```
./task 4095 1 0 # симуляция с 3095-ю пчёлами, горшком с вместимостью в порцию, без задержки
```

```
./task 1 1 0    # симуляция с одной пчелой, горшком в порцию, без задержки
```

```
./task 5 20 4000 # симуляция с 5-ю пчёлами, горшком со вместимостью 20 и случайной задержкой до 4-х секунд между «вылетами» одной пчелы
```

7. Список некоторых источников, использованных при разработке

1. pthreads(7) — Linux manual page // <https://man7.org/linux/man-pages/man7/pthreads.7.html>
2. pthread_spin_lock(3) — Linux manual page // https://man7.org/linux/man-pages/man3/pthread_spin_lock.3.html
3. pthread_cond_wait - man pages section 3: Basic Library Functions // https://docs.oracle.com/cd/E36784_01/html/E36784/pthread-cond-wait-3c.html
4. pthread_mutex_lock(3) - Linux man page // https://linux.die.net/man/3/pthread_mutex_lock
5. sem_overview(7) - Linux man page // https://linux.die.net/man/7/sem_overview