

Выбранный подход

Для взятия двух локов был использован *timed_mutex* и его метод *try_lock_for*. Теперь философ, который захотел поест, будет в цикле сначала пытаться в течение некоторого времени взять левую вилку, и в случае успеха попытается в течение некоторого времени взять уже правую вилку. Если правую вилку за это время взять удастся, вызовется метод *eat()*, затем опускаются правая и левая вилка и цикл разрывается, в противном случае отпускается левая вилка и цикл продолжается.

Код в проекте разбит на несколько файлов. Вся общая логика для философов вынесена в класс *BasePhilosopher*, логика, описанная выше, реализована в классе *Philosopher*, логика из файла-заготовки перенесена в класс *BlockingPhilosopher*.

Тестирование производительности

При тестировании производительности сравнивался алгоритм без защиты от взаимоблокировок (тот, который был в файле заготовке), алгоритм с использованием *try_lock_for* с максимальным временем ожидания лока 100 и 10 миллисекунд.

Все результаты и выводы программ лежат в папке *results*. В папке *blocking* - алгоритм без защиты от взаимоблокировок, в папке *10ms* - алгоритм с максимальным временем ожидания блокировки 10 миллисекунд, в папке *100ms* - со временем ожидания 100 миллисекунд.

Смотря на результаты, можно увидеть, что меньше всего философы ожидают, когда максимальное время ожидания - 10 миллисекунд. Также видно, что большее количество философов не сильно влияет на время их ожидания и суммарное количество приемов пищи.

В итоге получились следующие средние значения при пяти философях для алгоритма с максимальным временем ожидания 10ms, 100ms и алгоритма из заготовки соответственно:

1. 390 приемов пищи и 22000 ms суммарного ожидания
2. 350 приемов пищи и 25000 ms суммарного ожидания
3. 320 приемов пищи и 28000 ms суммарного ожидания

И такие результаты при ста философях:

1. 410 приемов пищи и 20000 ms суммарного ожидания
2. 325 приемов пищи и 27000 ms суммарного ожидания
3. 270 приемов пищи и 33000 ms суммарного ожидания

Также были произведены запуски с 10, 20 и 50 философами, результаты этих запусков находятся в папке *results*.