

Описание решения

Мой номер - 8, следовательно при выполнении данного задания мне достался метод `std::this_thread::sleep_until`. Ниже представлена организация проекта, который демонстрирует работу данной конструкции.

```
github.com/demyanov17/thread_task
├── main.cpp ..... Функция, организующая работу потоков
├── print.cpp ..... Вспомогательная функция для печати «*»
├── functions.h ..... Файл с вспомогательной функцией
├── Makefile ..... Файл для компиляции и запуска проекта
└── report.pdf ..... Отчёт по заданию, который Вы читаете в данный момент;
```

Рис. 1: структура github репозитория с проектом

main.cpp

В программе взаимодействуют 2 потока. Первый печатает текущее время с точностью до секунд и засыпает с использованием `std::this_thread::sleep_until` до наступления следующей минуты, после чего выводит сообщение о наступлении следующей минуты. Второй поток заботится о том, чтобы пользователь не скучал во время сна первого потока и каждую секунду печатает звёздочки в количестве, равном номеру секунды ожидания пробуждения (символы «*» появляются в арифметической прогрессии с $a_0 = 1$ и разностью, равной 1).

print.cpp

Вспомогательная функция, которая отвечает за работу второго потока - вывод звездочек в `std::cout`

functions.h

Модуль содержащий заголовок функции `print_()` из `print.cpp`

Makefile

Файл для сборки описанного проекта. Для запуска необходимо выполнить в терминале следующие команды:

1. `make`
2. `./task3`

Также можно чистить директорию от объектников, выполнив `make clean`.

Заключение

В данном отчете приведено описание структуры и модулей проекта. Стоит отметить, что данный проект носит учебный характер и направлен на ознакомление автора с параллельным программированием, очевидно, что вывод программы можно повторить, обходясь и одним потоком.