

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Департамент программной инженерии

**МИКРОПРОЕКТ №2
ЗАДАЧА О ПРОГРАММИСТАХ**

Пояснительная записка

Исполнитель
Студент группы БПИ 196
Судаков Дмитрий Юрьевич

Москва, 2020

Содержание

1	Постановка задачи	2
2	Описание алгоритма	3
2.1	Описание структуры данных	3
2.1.1	enum class StateE	3
2.1.2	enum class TaskE	3
2.1.3	struct Task	3
2.1.4	struct Info	3
2.1.5	метод proger	4
2.2	Формат входных данных	4
2.3	Формат выходных данных	4
3	Список литературы	5
4	Приложение 1. Логика программы	6
5	Приложение 2. Тестирование программы	7

1 Постановка задачи

В отделе работают три программиста. Каждый программист пишет свою программу и отдает ее на проверку другому программисту. Программист проверяет чужую программу, когда его собственная уже написана. По завершении проверки, программист дает ответ: программа написана правильно или написана неправильно. Программист спит, если не пишет свою программу и не проверяет чужую программу. Программист просыпается, когда получает заключение от другого программиста. Если программа признана правильной, программист пишет другую программу, если программа признана неправильной, программист исправляет ее и отправляет на проверку тому же программисту, который ее проверял. Создать многопоточное приложение, моделирующее работу программистов.

2 Описание алгоритма

Поведение программиста реализовано при помощи конечного автомата. Внутри цикла, который исполняется до взведения флага, которое происходит в состоянии 9. В цикле есть две switch-case секции обе по текущему состоянию, первая выводит сообщения в лог, а вторая обрабатывает логику.

Для вывода сообщений в первой switch-case секции используется mutex. Так же второй mutex используется для взаимодействия с переменной отвечающей за количество выполненных заданий, так как к ней могут обращаться сразу несколько человек.

Для общения между программистами используются семафоры. У каждого программиста есть очередь его заданий, и перед получением задания он ожидает свой семафор, а когда какой-то программист отдаёт другому задание он взводит его семафор и помещает задание в его очередь.

Схема конечного автомата приведена в разделе Приложении 1

2.1 Описание структуры данных

2.1.1 enum class StateE

Состояние, которое может принимать программист

Номер на схеме:	Состояние:	Описание:
0	START	Начальное состояние программиста;
1	WRITING	Написание очередного задания;
2	DEBUGING	Отладка задания;
3	REVIEWING	Проверка задания от другого программиста;
4	PUSHING_TASK	Передача задания другому программисту;
5	WAIT_TASK	Ожидание нового задания;
6	CHOOSE_TASK	Выбор нового задания;
7	END_TASKS	Все задания программиста выполнены;
8	END_WORK	Закончена работа на сегодня;
9	NULL_STATE	Состояние, которое не должно обрабатываться.

Если в какой-то момент в цикле обрабатывается состояние StateE::NULL_STATE, значит логика программы нарушена.

2.1.2 enum class TaskE

Задания, которыми обмениваются программисты.

Состояние:	Описание:
REVIEW_ACCEPT	Сигнал программисту что его предыдущее задание принято проверяющим И он может начинать новое;
DEBUG	Задание на отладку программы;
REVIEW	Задание на ревью.

2.1.3 struct Task

Структура описывающая задание, котрыми обмениваются программисты.

Тип:	Поле:	Описание:
TaskE	task	Тип передаваемого задания;
int	requester	Номер программиста, который запрашивает выполнение;
int	executer	Номер программиста, который будет выполнять задание.

2.1.4 struct Info

Структура с информацией, которая передаются каждому потоку.

Тип:	Поле:	Описание:
int	taskAmt	Количество заданий, которое надо выполнить программисту за день;
int	myNumber	Номер программиста;
pthread_mutex_t*	wrMutex	Мьютекс для вывода информации в консоль/файл;
pthread_mutex_t*	workMutex	Мьютекс для ограничения работы потоков с общей переменной;

2.1.5 метод proger

Метод, описывающий работу программиста.

Аргументы : void* Ссылка на структуру Info с информацией;

2.2 Формат входных данных

В начале работы программы пользователи просят ввести количество выполняемых заданий для каждого программиста - целое число от 1 до 10.

В аргументах командной строки можно передать путь к файлу (абсолютный или относительный) если он не был указан, то вся информация выводится в консоль.

2.3 Формат выходных данных

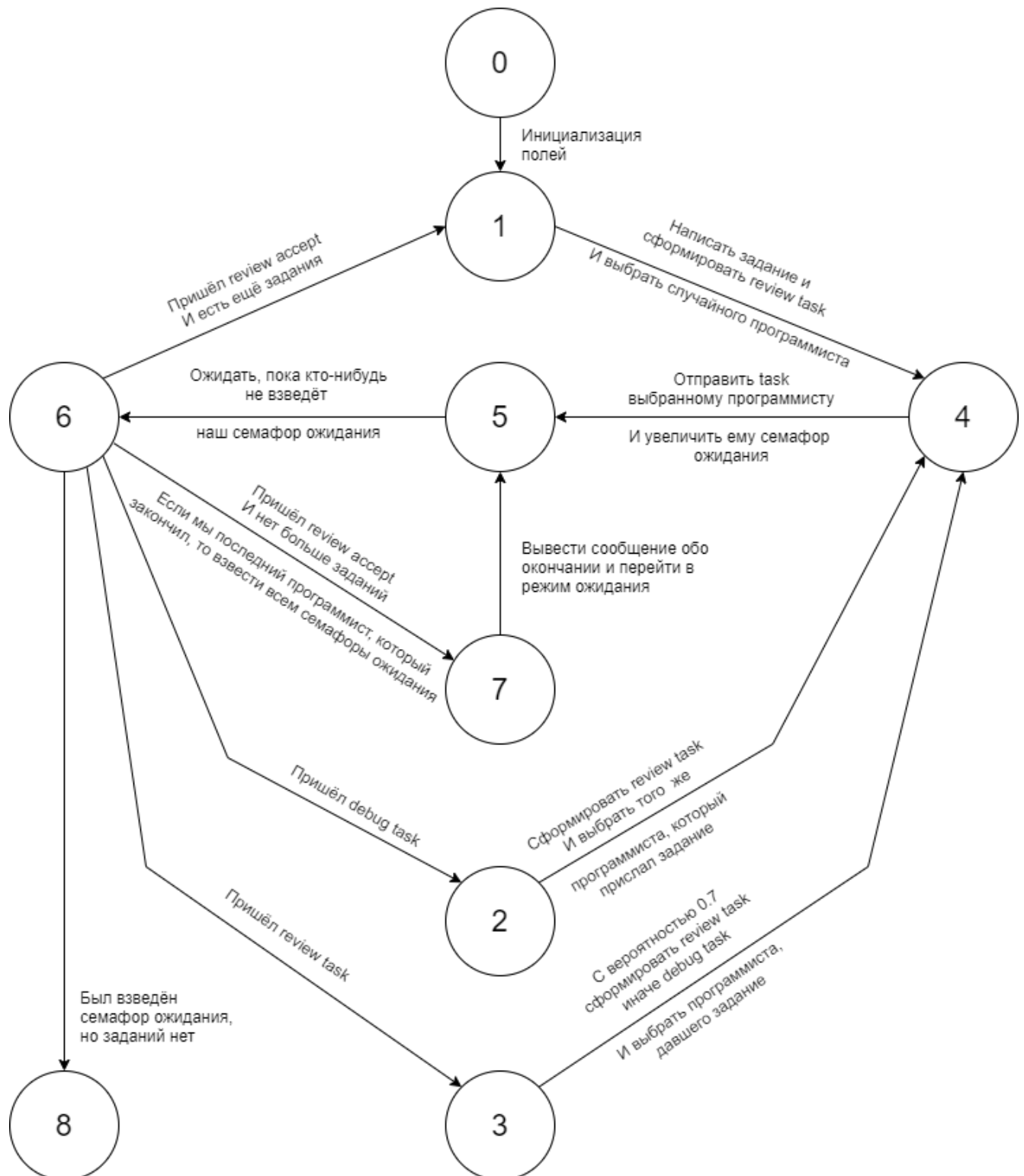
Сообщения от всех потоков выводятся в консоль или указанный файл.

3 Список литературы

- Пункт 4 из предложенного списка литературы по многопоточности. Грегори Р. Эндрюс. Основы многопоточного, параллельного и распределенного программирования. - М.: Издательский дом "Вильямс 2003.
- Семафоры в pthreads [Электронный ресурс] / learnc.info (дата обновления: 15.01.2018). - Электрон. текстовые дан. - Режим доступа: https://learnc.info/c/pthreads_semaphores.html (дата обращения: 08.12.2020).
- Конечный автомат [Электронный ресурс] / Wikipedia (дата обновления: 07.12.2020). - Электрон. текстовые дан. - Режим доступа: https://ru.wikipedia.org/wiki/\T2A\CYRK\T2A\cyro\T2A\cyrn\T2A\cyre\T2A\cyrch\T2A\cyrn\T2A\cyry\T2A\cyrishrt_\T2A\cyra\T2A\cyrv\T2A\cyrt\T2A\cyro\T2A\cyrm\T2A\cyra\T2A\cyrt (дата обращения: 08.12.2020).

4 Приложение 1. Логика программы

Конечный автомат, описывающий логику программиста.



5 Приложение 2. Тестирование программы

Сначала попробуем ввести некорректные данные :

```
Enter the tasks amount for three programmers (from 1 to 10): 0 1 2

Incorrect tasks amount
dimasya@DESKTOP-OIONLKE:/mnt/s/SoftCraft/sources/CONSOLE/MicroProject_2$
```

Попробуем запустить без аргументов :

```
dimasya@DESKTOP-OIONLKE:/mnt/s/SoftCraft/sources/CONSOLE/MicroProject_2$ ./res
Enter the tasks amount for three programmers (from 1 to 10): 1 1 1

Programmer : 0, start working
Programmer : 0, start writing task : 1
Programmer : 1, start working
Programmer : 1, start writing task : 1
Programmer : 2, start working
Programmer : 2, start writing task : 1
Programmer : 1, send REVIEW task to 0
Programmer : 1, wait new task (sleep)
Programmer : 2, send REVIEW task to 0
Programmer : 2, wait new task (sleep)
Programmer : 0, send REVIEW task to 2
Programmer : 2, get REVIEW task
Programmer : 2, start reviewing task from programmer : 0
Programmer : 0, get REVIEW task
Programmer : 0, start reviewing task from programmer : 1
Programmer : 0, send REVIEW ACCEPT to 1
Programmer : 0, get REVIEW task
Programmer : 0, start reviewing task from programmer : 2
Programmer : 1, get REVIEW ACCEPT
Programmer : 1, dont have any tasks for today and wait for review requests
Programmer : 1, wait new task (sleep)
Programmer : 2, send REVIEW ACCEPT to 0
Programmer : 2, wait new task (sleep)
Programmer : 0, send REVIEW ACCEPT to 2
Programmer : 0, get REVIEW ACCEPT
Programmer : 0, dont have any tasks for today and wait for review requests
Programmer : 2, get REVIEW ACCEPT
Programmer : 2, dont have any tasks for today and wait for review requests
Programmer : 2, ends his work for today
Programmer : 2, go home
Programmer : 1, ends his work for today
Programmer : 1, go home
Programmer : 0, ends his work for today
Programmer : 0, go home
dimasya@DESKTOP-OIONLKE:/mnt/s/SoftCraft/sources/CONSOLE/MicroProject_2$
```

Все последующие тесты будут запускаться с указанием имени файла для вывода в аргументах командной строки. Результаты тестов будут лежать в папке output, в файлах log_ .txt, где вместо символа " _ "будет номер теста.

Тесты :

1. 1 2 3

2. 1 5 8

3. 10 10 10