

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ИНСТИТУТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Мини-проект №2  
по предмету «Архитектуры вычислительных систем»  
Вариант №6 (студент №28 в списке)  
«Задача о курильщиках»

## Пояснительная записка

Выполнил студент группы БПИ199  
факультета компьютерных наук  
Щербаков Дмитрий Александрович  
13 декабря 2020 года

# Текст задания

Есть три процесса-курильщика и один процесс-посредник. Курильщик непрерывно скручивает сигареты и курит их. Чтобы скрутить сигарету, нужны табак, бумага и спички. У одного процесса-курильщика есть табак, у второго – бумага, а у третьего – спички. Посредник кладет на стол по два разных случайных компонента. Тот процесс-курильщик, у которого есть третий компонент, забирает компоненты со стола, скручивает сигарету и курит. Посредник дожидается, пока курильщик закончит, затем процесс повторяется. Создать многопоточное приложение, моделирующее поведение курильщиков и посредника. При решении задачи использовать семафоры.

# Метод решения

Программа написана на языке C99.

С помощью библиотеки `pthread` создадим 4 потока – 3 курильщика и один посредник. Для синхронизации мы используем четыре POSIX-семафора – по одному на каждый из созданных потоков. Семафор посредника инициализируем числом 1 – тогда он сможет начать выкладывать компоненты сигарет на стол. Семафоры курильщиков проинициализированы нулем. Семафоры курильщиков случайным образом разблокируются в потоке-посреднике, затем в соответствующем потоке-курильщике семафор блокируется, совершается процесс «курения» (в нашем случае `sleep(1)`) и разблокируется семафор для посредника, после чего он снова может выкладывать на стол компоненты сигарет.

В данном случае семафоры являются бинарными – принимают лишь значения 1 и 0 и работают подобно мьютексам.

В работе используются именованные семафоры, созданные функцией `sem_open`, так как неименованные семафоры, созданные с помощью `sem_init`, не работают на POSIX-совместимой macOS.

Так как в задании не указаны условия выхода, был установлен лимит на 10 курений для всех курильщиков. Это условие контролируется с помощью переменной `total_smokes`, которая на каждом шаге увеличивается на 1, и все потоки завершают работу по достижении числа 10. Поток-посредник должен разбудить другие потоки для того, чтобы они могли завершить работу.

# Запуск и работа программы

Программа компилируется и линкуется с помощью make и запускается из командной строки без аргументов.

```
d.shcherbakov1@macbook-d project02 % make
gcc project02.c -O3 -o project02 -pthread
d.shcherbakov1@macbook-d project02 % ./project02
Smoker with paper got tobacco and matches and is smoking
Smoker with paper has finished smoking
Smoker with paper got tobacco and matches and is smoking
Smoker with paper has finished smoking
Smoker with matches got paper and tobacco and is smoking
Smoker with matches has finished smoking
Smoker with tobacco got paper and matches and is smoking
Smoker with tobacco has finished smoking
Smoker with tobacco got paper and matches and is smoking
Smoker with tobacco has finished smoking
Smoker with tobacco got paper and matches and is smoking
Smoker with tobacco has finished smoking
Smoker with paper got tobacco and matches and is smoking
Smoker with paper has finished smoking
Smoker with matches got paper and tobacco and is smoking
Smoker with matches has finished smoking
Smoker with tobacco got paper and matches and is smoking
Smoker with tobacco has finished smoking
d.shcherbakov1@macbook-d project02 % █
```

*Рисунок 1. Пример работы программы*

# СПИСОК ИСТОЧНИКОВ

1. man 2 sem\_wait
2. man 2 sem\_post
3. man 2 sem\_open
4. man 2 sem\_unlink
5. man 2 sem\_close
6. man 3 pthread\_create
7. man 3 pthread\_join