

Правительство Российской Федерации
Национальный исследовательский университет
«Высшая школа экономики»

Факультет компьютерных наук
Департамент программной инженерии

Микропроект №2 по Архитектуре вычислительных систем.
Пояснительная записка.

Исполнитель:
Студент группы БПИ197
Пике Кирилл

13 Декабря 2020г.

Оглавление

1. Текст задачи.....	3
2. Взаимодействие субъектов задания.	4
3. Реализация проекта.	5
4. Работа программы.	6
5. Используемые источники.....	7

1. Текст задачи.

Задача о гостинице - 3 (дамы и джентльмены). В гостинице 10 номеров рассчитаны на одного человека и 15 номеров рассчитаны на двух человек. В гостиницу приходят клиенты дамы и клиенты джентльмены, и конечно они могут провести ночь в номере только с представителем своего пола. Если для клиента не находится подходящего номера, он уходит искать ночлег в другое место. Создать многопоточное приложение, моделирующее работу гостиницы.

2. Взаимодействие субъектов задания.

В данном задании присутствует 2 типа субъектов: управляющий отелем и клиенты.

Можно выделить несколько типов взаимодействия между субъектами:

- Клиенты генерируются и приходят в очередь для заселения, а управляющий проверяет свободны ли какие-либо еще индивидуальные номера или двухместные номера по правилу заселения и заселяет гостя если место есть или же отправляет его дальше, если места нет.
- Когда появляется новый клиент, то появляется сообщение о его возникновении, указывающее на его порядковый номер и пол (М – мужской, F - женский).
- Когда клиента успешно заселяют, то появляется сообщение об этом, указывающее порядковый номер заселенца.
- Если посетитель уже пришел, но для него не было места, то ему приходится уйти, что сопровождается соответствующим сообщением.

3. Реализация проекта.

Для реализации проекта были использованы стандартная библиотека потоков языка программирования C++ и Posix thread.

Тип многопоточной программы, исходя из условия, можно было бы определить как зависимость производитель-потребитель. У нас есть производитель, который генерирует дам и джентльменов и потребитель в виде гостиницы. Критическими секциями являются два момента. Когда происходит проверка наличия свободных мест в гостинице, так как нам нужна обновленная информация о наличии мест и момент, когда мы заселяем персону в номер.

Стандартная библиотека была использована для создания 2-х потоков: потока для генерации клиентов и потока для их размещения в отеле. Библиотека Posix thread была использована для синхронизации потоков и для взаимного исключения исполнения критичных участков кода.

Для симуляции номеров отеля были использованы 1 байтные массивы, которые хранили либо букву 'M', либо 'F' в зависимости от пола жителя номера. Для генерации клиентов была использована функция генерации псевдослучайных чисел, аргумент которой был количеством тиков, так как из-за скорого ее использования, эта функция без аргумента давала одинаковые значения на каждом запуске программы. Для реализации очереди из клиентов был использован двусвязный список.

4. Работа программы.

Программа запускается из командной строки, никаких аргументов при запуске не требуется. Никаких дополнительных файлов не требуется.

```
Microsoft Visual Studio Debug Console
Client number 72F has arrived.
Client number 74F has arrived.
Client number 75F has arrived.
1 Settled
2 Settled
3 Settled
4 Settled
5 Settled
6 Settled
7 Settled
8 Settled
9 Settled
10 Settled
11 Settled
12 Settled
13 Settled
14 Settled
15 Settled
16 Settled
17 Settled
18 Settled
19 Settled
20 Settled
21 Settled
22 Settled
23 Settled
24 Settled
25 Settled
26 Settled
27 Settled
28 Settled
29 Settled
30 Settled
31 Settled
32 Settled
33 Settled
34 Settled
35 Settled
36 Settled
37 Settled
38 Settled
39 Settled
40Failed to settle
41Failed to settle
42Failed to settle
43Failed to settle
44Failed to settle
45Failed to settle
46Failed to settle
47 Settled
```

Создаются клиенты, начиная с какого-то клиента происходит заселение. У всех клиентов в очереди, помимо заселившихся, не получается заселиться и они уходят

```
34 Settled
35 Settled
36 Settled
37 Settled
38 Settled
39 Settled
40Failed to settle
41Failed to settle
42Failed to settle
43Failed to settle
44Failed to settle
45Failed to settle
46Failed to settle
47 Settled
Client number 76F has arrived.
Client number 48F Failed to settle
Client number 49F Failed to settle
Client number 50F Failed to settle
Client number 51F Failed to settle
Client number 52F Failed to settle
Client number 53F Failed to settle
Client number 54F Failed to settle
Client number 55M Failed to settle
Client number 56F Failed to settle
Client number 57F Failed to settle
Client number 58M Failed to settle
Client number 59F Failed to settle
Client number 60F Failed to settle
Client number 61M Failed to settle
Client number 62M Failed to settle
Client number 63F Failed to settle
Client number 64M Failed to settle
Client number 65F Failed to settle
Client number 66F Failed to settle
Client number 67F Failed to settle
Client number 68F Failed to settle
Client number 69M Failed to settle
Client number 70F Failed to settle
Client number 71M Failed to settle
Client number 72M Failed to settle
```

В конце так же присутствует отображение комнат отеля по полу, чтобы убедиться, что заселение прошло верно.

5. Используемые источники.

- Материалы лекций и семинаров учебного курса «Архитектура вычислительных систем» на сайте Softcraft.ru: <http://softcraft.ru/edu/comparch/> от 11.12.2020.
- Справочный материал по использованию Deque в языке программирования C++ <https://metanit.com/cpp/tutorial/7.8.php> от 12.12.2020.
- Справочный материал для ознакомления с методом srand в языке программирования C++ <http://www.cplusplus.com/reference/cstdlib/srand/> от 12.12.2020.
- Справочный материал для ознакомления с методом clock в языке программирования C++ <http://www.cplusplus.com/reference/ctime/clock/> от 12.12.2020.