|  |
| --- |
| **ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**  **НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**  **«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**  Факультет компьютерных наук  Департамент программной инженерии  **МНОГОПОТОЧНОЕ ПРОГРАММИРОВАНИЕ.**  **ВЗАИМВОДЕЙСТВИЕ ПОТОКОВ. МИКРОПРОЕКТ.**  Пояснительная записка  Исполнитель:  Студент группы БПИ191  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Дондик Я.В. /  «12» декабря 2020 г. |

Содержание

[1. Текст задания 3](#_Toc58789062)

[2. Применяемые расчетные методы 4](#_Toc58789063)

[2.1. Теория решения задания 4](#_Toc58789064)

[2.1.1. Общая идея решения задания 4](#_Toc58789065)

[2.1.2. Входные и выходные данные 4](#_Toc58789066)

[2.1.3. Особенности при решении задания 4](#_Toc58789067)

[3. Тестирование программы 5](#_Toc58789068)

[3.1.1. Тест 1 5](#_Toc58789069)

[3.1.2. Тест 2 6](#_Toc58789070)

[3.1.3. Тест 3 7](#_Toc58789071)

[3.1.4. Тест 4 9](#_Toc58789072)

[4. Список источников. 10](#_Toc58789073)

1. Текст задания

**Вариант 14.** Задача о гостинице-2 (умные клиенты).

В гостинице 10 номеров с ценой 200 рублей, 10 номеров с ценой 400 рублей и 5 номеров с ценой 600 руб.

Клиент, зашедший в гостиницу, обладает некоторой суммой и получает номер по своим финансовым возможностям, если тот свободен. Если среди доступных клиенту номеров нет свободных, клиент уходит искать ночлег в другое место.

Создать многопоточное приложение, моделирующее работу гостиницы.

2. Применяемые расчетные методы

2.1. Теория решения задания

2.1.1. Общая идея решения задания

Разделим задачу на три основных вида потоков:

1. Главный поток – создает и контролирует поток «отеля» и потоки «туристов»;
2. Поток «Отель» – приглашает клиентов (туристов) к ресепшену, чтобы те попробовали снять номер;
3. Поток «Турист» – пытается снять свободный номер согласно своим финансовым возможностям.

Пусть все «туристы» ожидают приглашение от «отеля», а после получения такового, подходят к ресепшену и пытаются арендовать комнату по наибольшей стоимости, которая подходит им по бюджету. Если свободных номеров по этой цене не оказалось, «турист» пытается снять номер подешевле.

Если свободных номеров, за которые «турист» смог бы заплатить, не оказалось, то он разворачивается и уходит прочь (т. е. завершает свою работу и ждет присоединения к главному потоку).

2.1.2. Входные и выходные данные

Входные данные: количество «туристов» (потоков) N, где N >= 1

Выходные данные: логгированный вывод информации, с указанием времени и действия «туриста»

2.1.3. Особенности при решении задания

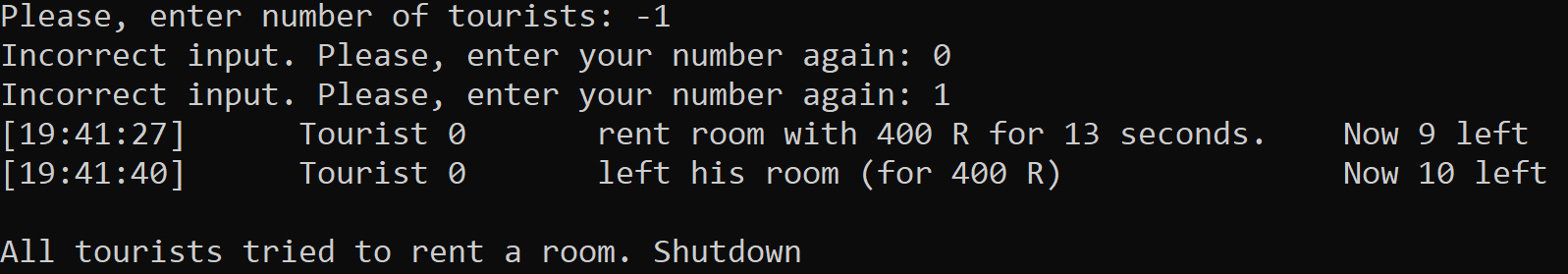
1. Для работы с блокировкой вывода, исключительного доступа к количеству свободных комнат используются семафоры из библиотеки semaphore.h
2. Для работы с приглашениями «туристов» были использованы condition\_variable, с защитой от «ложных пробуждений»
3. Для удобства визуализации работы программы «туристы» приглашаются с частотой раз в 0.3 сек., а аренда комнаты длится от 5 до 19 сек.
4. Вывод логов в консоль сопровождается меткой времени о произошедшем событии и указанием действия «туриста».
5. Поток «отеля» завершает работу (ожидает присоединения к главному потоку) с того момента, как были приглашены все «туристы»2.2. Дополнительный функционал программы

3. Тестирование программы

**3.1. Проверка программы на тестах**

3.1.1. Тест 1

Проверим программу неправильном вводе, а затем попробуем ввести одного туриста:

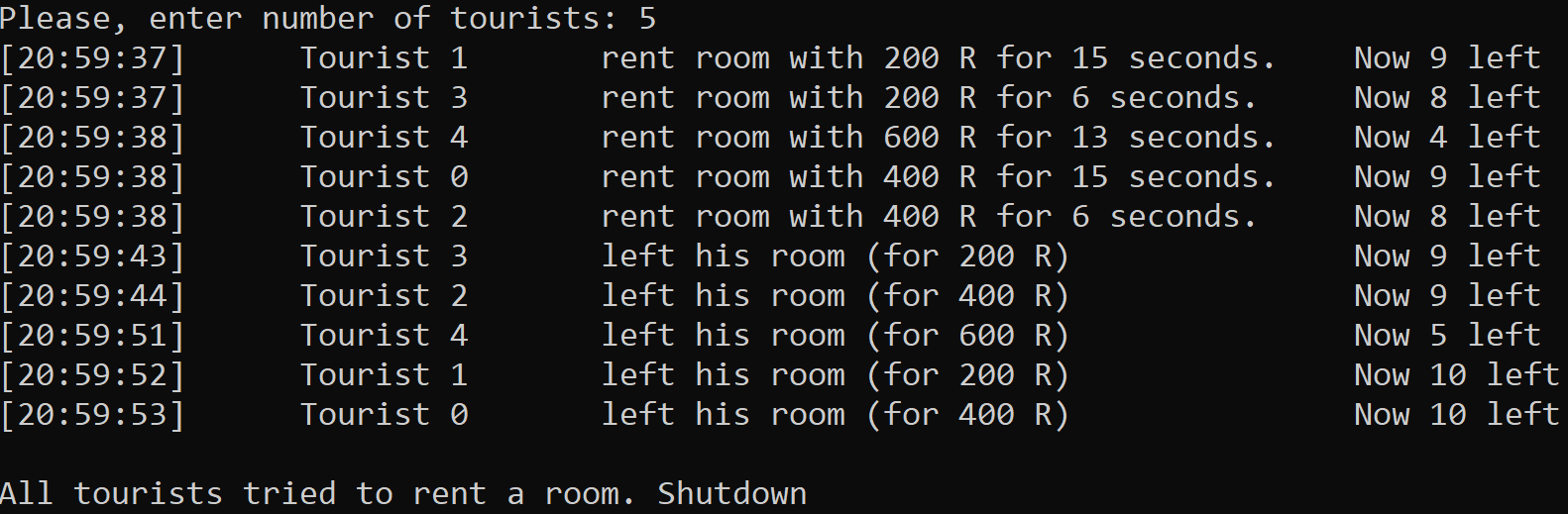


*Рисунок 1.* Тестирование программы на неправильном вводе

Несложно заметить, что программа принимает количество «туристов» от 1 и более. Далее «турист» успешно арендует комнату за 400 R и свободных комнат остается 9. По прошествии 13 секунд, «турист» освобождает комнату и теперь свободных комнат за 400 R становится снова 10.

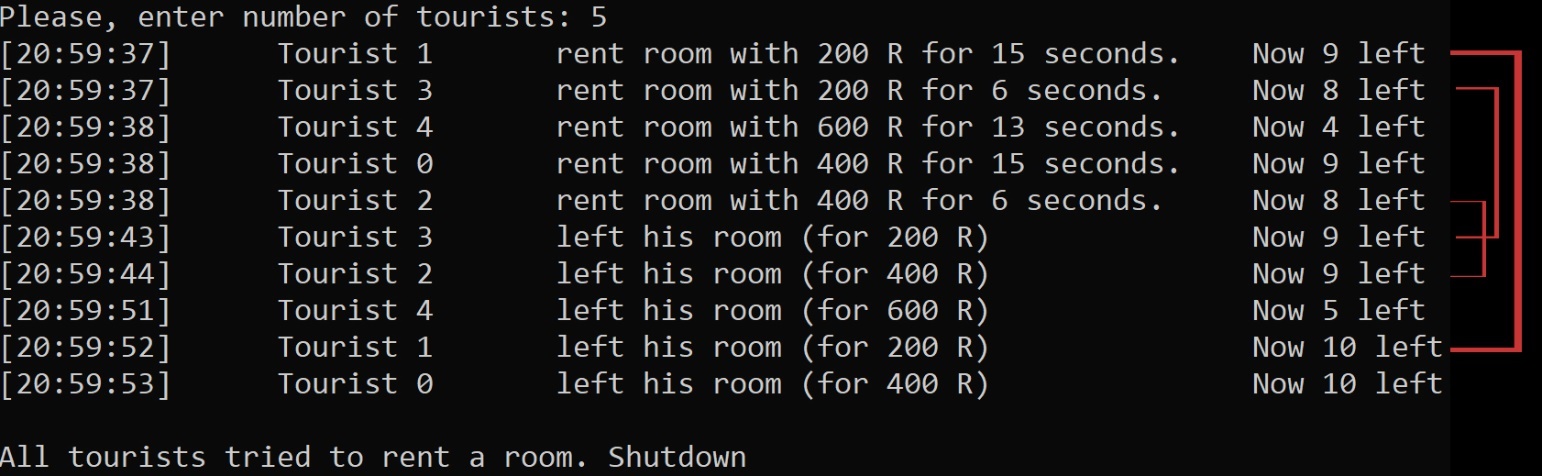
Программа отработала успешно.

3.1.2. Тест 2

Проверим программу на тесте с пятью «туристами» (рис. 2):

*Рисунок 2.* Тестирование программы на пяти туристах

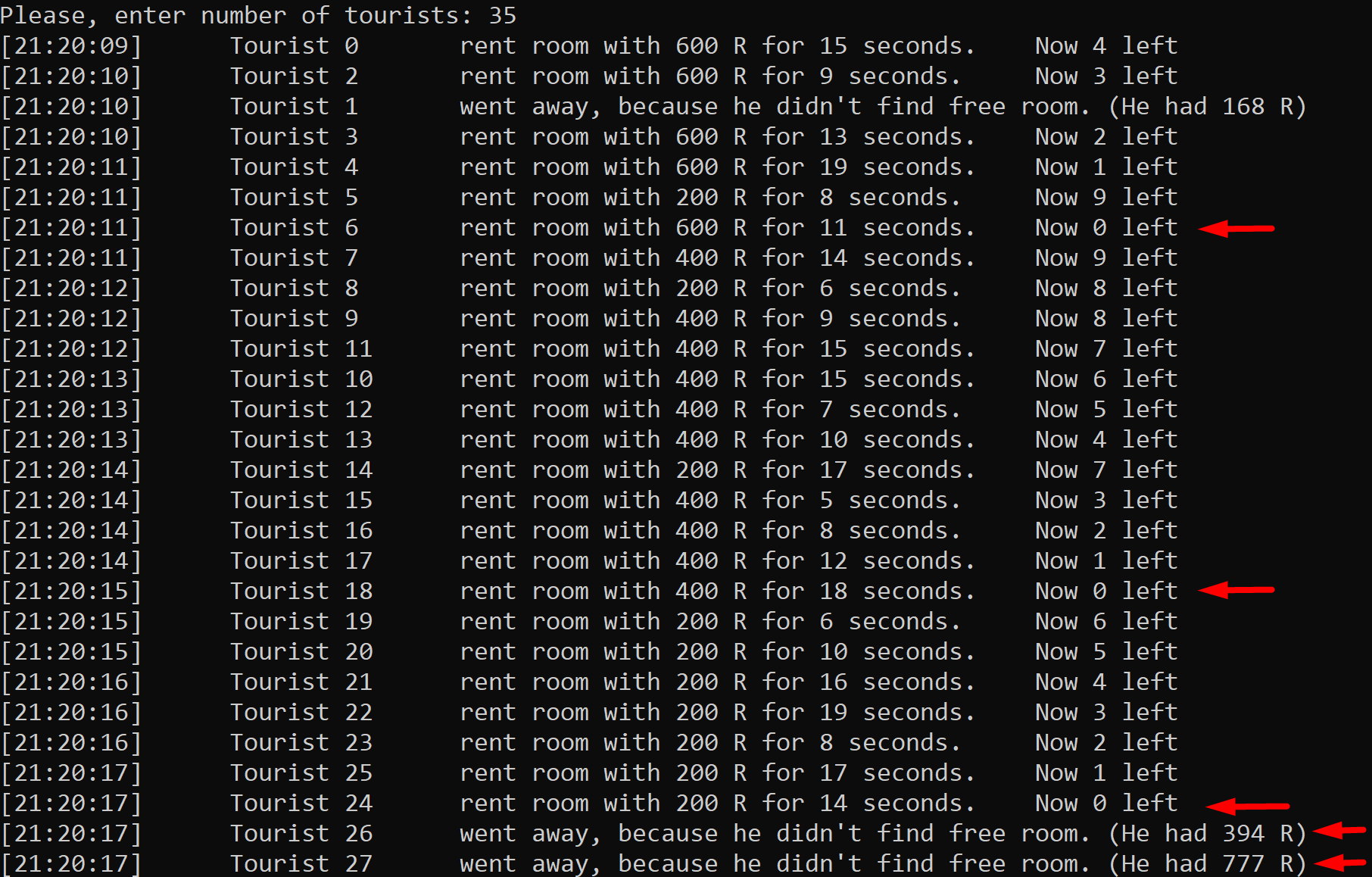
Несложно заметить, что у всех туристов хватило денег на какой-нибудь свободный номер. Аренда и освобождение комнаты на временных метках соответствует планируемому количеству секунд аренды комнаты (рис. 3).

Также, счетчик свободных комнат соответствующей стоимости увеличивается и уменьшается, когда комната арендуется и освобождается (рис. 3).

*Рисунок 3.* Верное освобождение комнат соответствующих стоимостей

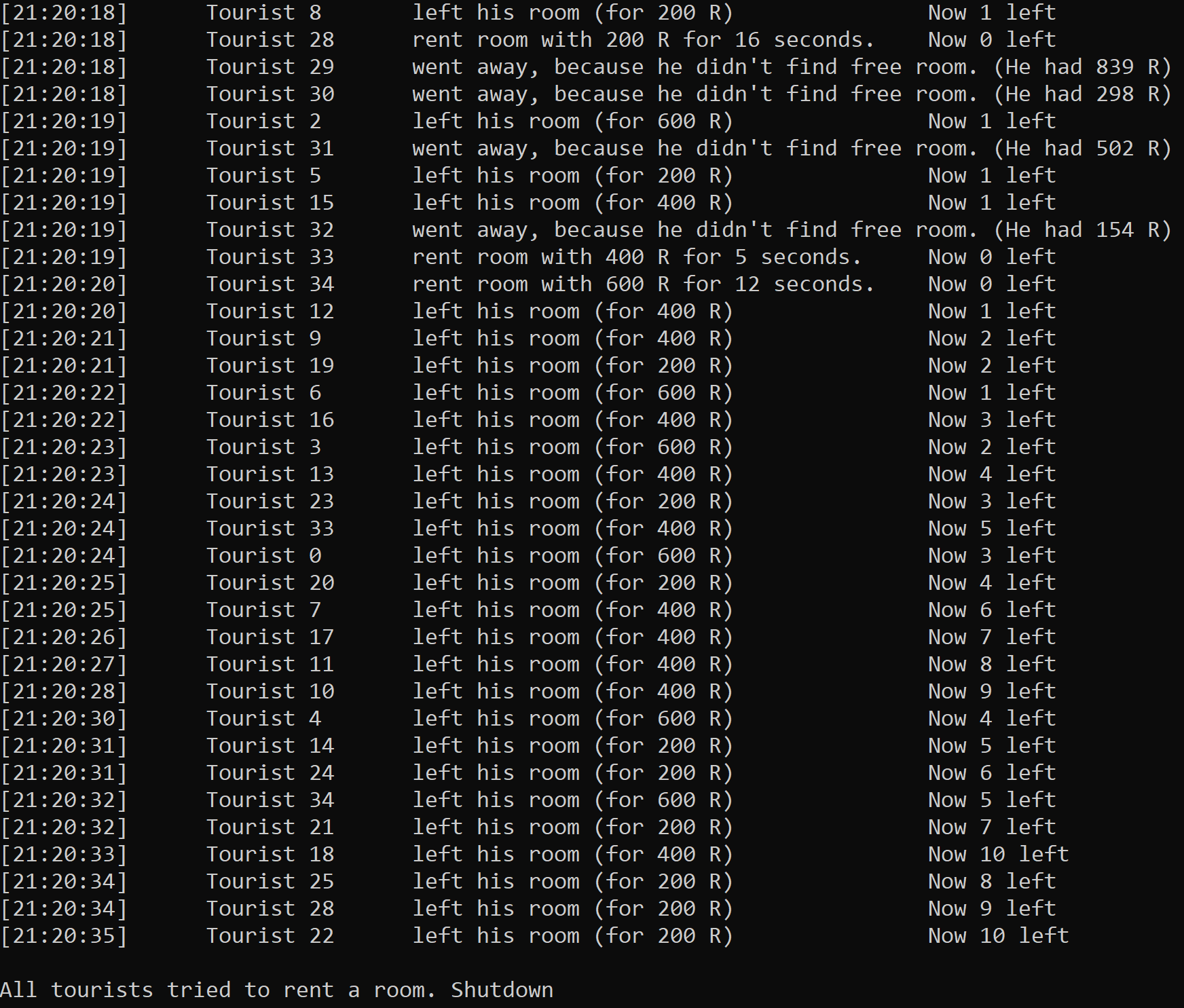
Программа отработала успешно.

3.1.3. Тест 3

Проверим программу на 35 «туристах», чтобы все комнаты были заняты в один определенный момент:

*Рисунок 4.* Тестирование программы на тексте (с пробелами)

Tourist 1 ушел, т.к. у него не хватало денег даже на самый дешевый номер (200 R). Постепенно все комнаты разных стоимостей были заняты и к времени [21:20:17] не осталось свободных номеров, поэтому Tourist 26 и Tourist 27 развернулись и ушли, хотя у них хватало денег.



*Рисунок 5.* Завершение работы программы

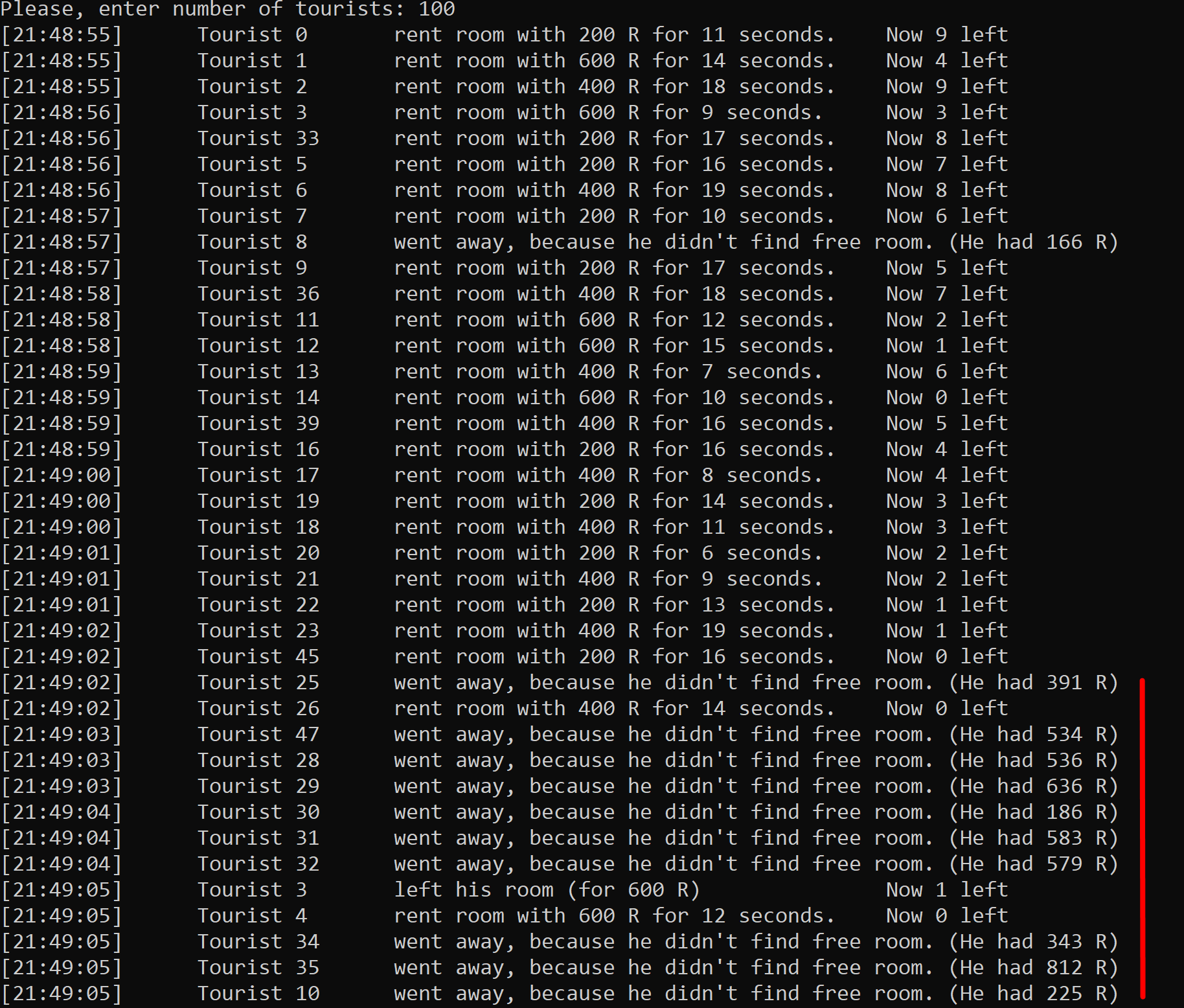
Далее нехватка мест продолжилась и «туристам» 29, 30 и 31 пришлось уйти примерно в [21:20:18], хотя у них хватало финансов на самый дешевый номер.

Tourist 32 не смог арендовать номер в [21:20:19] по причине нехватки финансов на самый дешевый номер (он имел 154/200 R).

После обработки всех 35-ти туристов отелем, они потихоньку освобождают номера. После освобождения номера «туристом» 22 (самым последним), программа завершает работу.

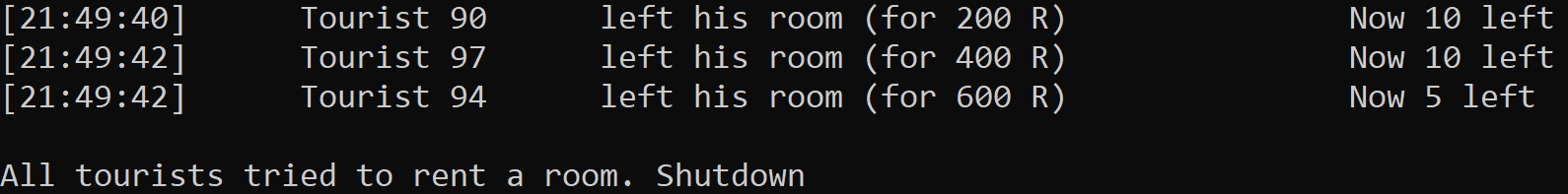
Программа отработала успешно.

3.1.4. Тест 4

Проверим программу на работу с сотней «туристов» (рис. 6):

*Рисунок 6.* Тестирование программы на сотне «туристов».

Довольно быстро не осталось свободных номеров, и очень многие туристы развернулись и ушли, хотя у них были финансы чтобы оформить номера.

В конце концов, все туристы были обработаны (кто-то смог получить номер, кто-то нет), и перед завершением программы видно, что все туристы уже покинули отель и все номера стали свободными (рис. 7):

*Рисунок 7.* Корректное завершение работы отеля.

Программа отработала успешно.

4. Список источников.

1. Документация по conditional\_variable [Электронный ресурс] // MSDN: [сайт]. [2020]. URL: <https://docs.microsoft.com/ru-ru/cpp/standard-library/condition-variable-class?view=msvc-160&viewFallbackFrom=vs-2019>, режим доступа: свободный, дата обращения: 13.12.2020

2. Условия выполнения задания, сайт дисциплины [Электронный ресурс] // SoftCraft: [сайт]. [2020]. URL: <http://softcraft.ru/edu/comparch/tasks/mp02/>, режим доступа: свободный, дата обращения: 13.12.2020

3. Работа с семафорами [Электронный ресурс] // LearnC- обучающий сайт: [сайт]. [2020]. URL: <https://learnc.info/c/pthreads_semaphores.html>, режим доступа: свободный, дата обращения: 13.12.2020

4. Работа с условными переменным [Электронный ресурс] // Уголок системного программиста: [сайт]. [2020]. URL: <https://habr.com/ru/post/182610/>, режим доступа: свободный, дата обращения: 13.12.2020