Правительство Российской Федерации

Национальный исследовательский университет

«Высшая школа экономики»

Факультет компьютерных наук

Департамент программной инженерии

Микропроект №2

Вариант 15

Выполнил

Студент группы БПИ193

Минец Максим

mvminets@edu.hse.ru

Москва 2020

Оглавление

1. Задание

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. Решение задачи

2.1 Формулировка задания

Заселяем новоприбывших гостей, пока гостиница не будет заполнена (то есть пока есть свободные номера). Заселить гостя мы можем в номер только с представителем своего пола. Отсюда логично будет сначала заполнять двуместные номера, а после того, как в двуместных закончатся места, перейти к заполнению одноместных. На каждого гостя выделяется отдельный работник (поток) отеля.

2.2 Решение задачи

Запустить фабрику потоков, собрать потоки во избежание утечки памяти. На каждой итерации (приход гостя) вызываем отдельный поток. К решению данной задачи применяем модель взаимодействующих равных.

2.3 Формат ввода данных

В самом начале пользователь видит эскиз пустого отеля (рис. 1). Здесь ‘x’ – это пустой одноместный номер, а ‘x_x’ – пустой двухместный номер. Также, эскиз отеля имеет некоторые декоративные элементы: название – ‘Four Seasons ’, кафе на последним этаже отеля – ‘Cafe’ и ресепшн на первом этаже при входе в отель – ‘Rec’. Эти элементы никак не влияют на работу программы.

рис. 1

После этого пользователю предлагается выбор (рис. 2) того, как будут приходить гости в отель:

1) Рандомно, с помощью самой программы, пока все места в отеле не будут заняты;

2) Пользователь будет сам вводить каждого следующего приходящего гостя в отель, пока все места в нём не будут заняты.

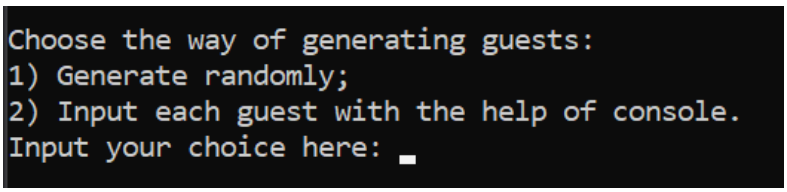


рис. 2

В первом случае пользователь больше ничего не вводит. Во втором случае пользователю необходимо ввести пол следующего гостя: ‘G’ или ‘g’, если гость – Джентльмен, и ‘L’ или ‘l’, если гость – Леди (рис. 3).

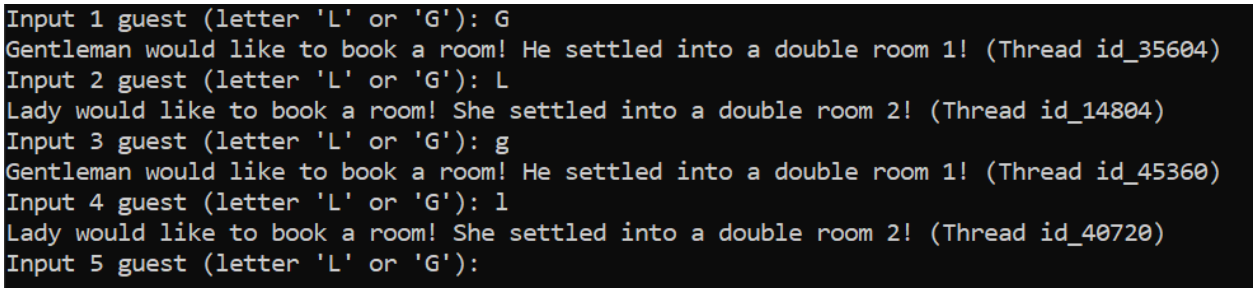


рис. 3

2.4 Формат выводы данных

Несмотря на выбор пользователя при выборе того, как будут приходить гости вывод данных происходит одинаково. После того, как гость пришел в отель возможны два варианта развития событий:

1) **{Gentleman/Lady}** would like to book a room! **{He/She}** settled into a **{single/double}** room **{number}**! (Thread id_**{Thread_id}**) – если в отеле есть свободные места для ночлега, то заселяем гостя в номер отеля (рис. 4);

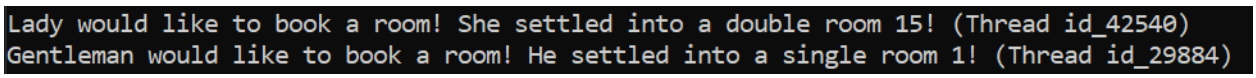


рис. 4

2) **{Gentleman/Lady}** would like to book a room! There are no free rooms. **{Gentleman/Lady}** have to go away and find another place to sleep (Thread id_**{Thread_id}**) – если мест для данного пола уже нет, то он отправляется искать другое место для ночлега (рис. 5).

рис. 5

Здесь: {Gentleman/Lady} и {He/She} – выводятся в зависимости от пола пришедшего гостя, {single/double} – тип номера (одноместный или двухместный), {number} – номер, который заселили гостя, {Thread_id} – идентификатор потока, который выполнил заселение гостя.

После того, как все гости заселены в отель выводится эскиз, на котором демонстрируются номера отеля с проживающими в них гостями и информация о том, что все гости были заселены (рис. 6). Здесь ‘G’ – Джентльмен, а ‘L’ – Леди. Также, отель имеет некоторые

декоративные элементы: название – ‘Four Seasons ’, кафе на последним этаже отеля – ‘Cafe’ и ресепшн на первом этаже при входе в отель – ‘Rec’. Эти элементы никак не повлияли на работу программы.

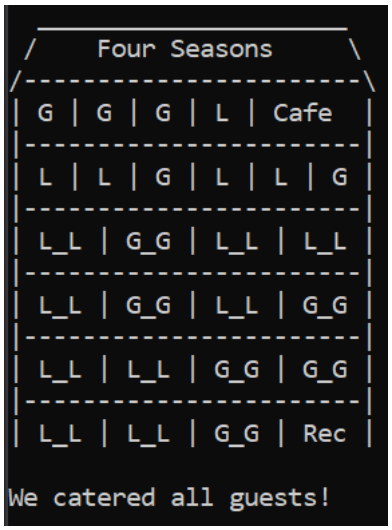


рис. 6

В случае если кто-то из гостей не смог заселиться, вместо “We catered all guests!” будет выведено: “We did not mahage to cater {num} visitors…”, где {num} – число гостей, которых не удалось заселить (рис. 7).

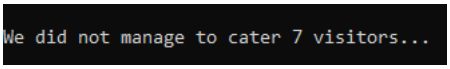


рис. 7

3. Тестирование программы

3.1 Тест № 1

При выборе метода прихода гостей пользователь может ввести только числа 1 или 2. При некорректном вводе программа предупреждает пользователя и просит ввести его цифру заново.

```
Choose the way of generating guests:
1) Generate randomly;
2) Input each guest with the help of console.
Input your choice here: 0
Incorrect input. You should input '1' or '2': -1
Incorrect input. You should input '1' or '2': 3
Incorrect input. You should input '1' or '2': 10
Incorrect input. You should input '1' or '2': 14
Incorrect input. You should input '1' or '2': -2
Incorrect input. You should input '1' or '2': 1000
Incorrect input. You should input '1' or '2': 2
Input 1 guest (letter 'L' or 'G'):
```

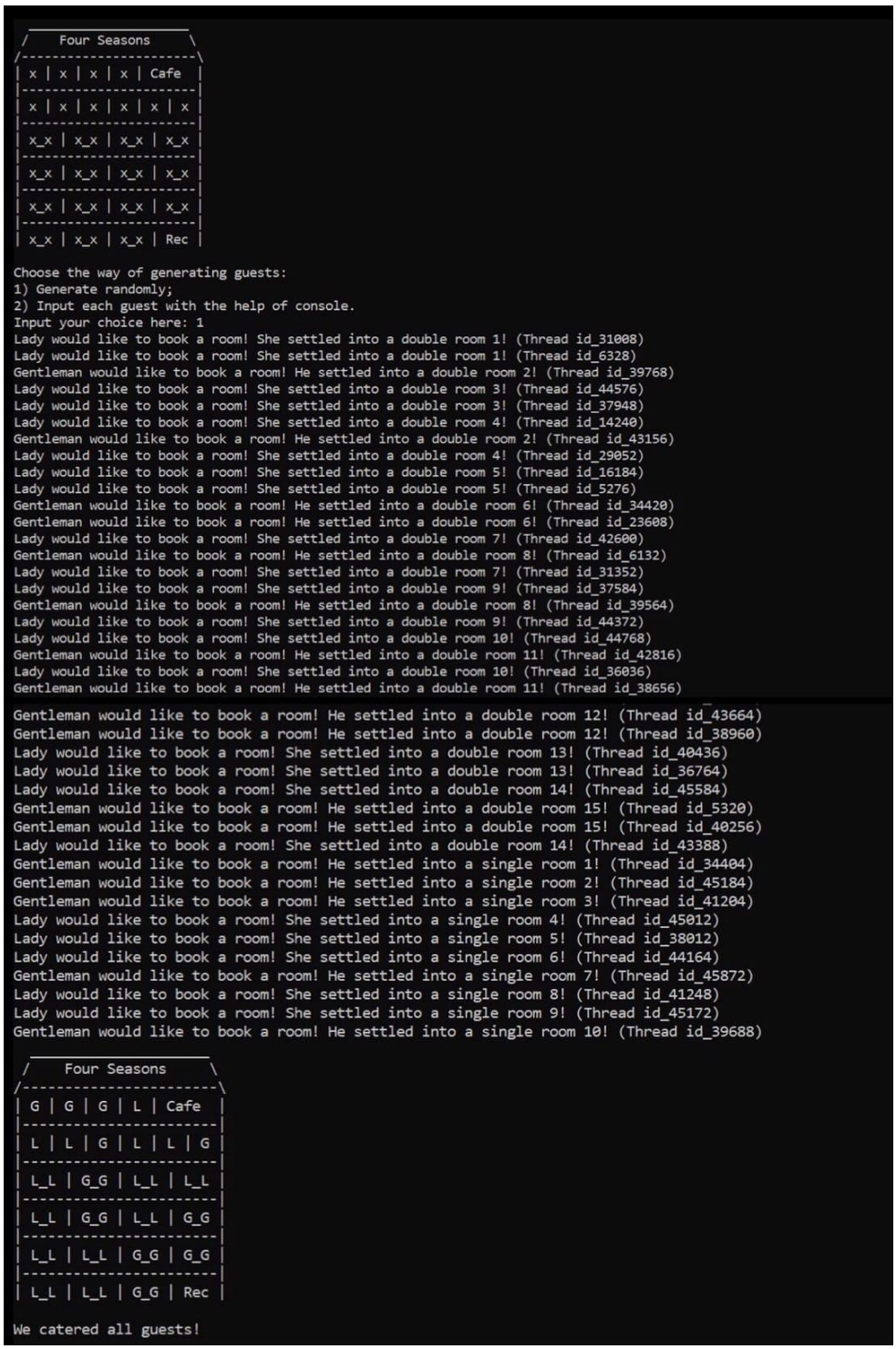
3.2 Тест № 2

При некорректном вводе текущего гостя программа предупреждает пользователя о некорректном вводе и просит попробовать еще раз ввести пол гостя (‘G’ или ‘L’).

```
Choose the way of generating guests:
1) Generate randomly;
2) Input each guest with the help of console.
Input your choice here: 2
Input 1 guest (letter 'L' or 'G'): -1
Incorrect input. You should input 'G' or 'L' (Gentleman and Lady respectively): 0
Incorrect input. You should input 'G' or 'L' (Gentleman and Lady respectively): e
Incorrect input. You should input 'G' or 'L' (Gentleman and Lady respectively): gentle
Incorrect input. You should input 'G' or 'L' (Gentleman and Lady respectively): lusya
Incorrect input. You should input 'G' or 'L' (Gentleman and Lady respectively): G
Gentleman would like to book a room! He settled into a double room 1! (Thread id_41548)
Input 2 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a double room 2! (Thread id_7068)
Input 3 guest (letter 'L' or 'G'): g
Gentleman would like to book a room! He settled into a double room 1! (Thread id_40144)
Input 4 guest (letter 'L' or 'G'): l
Lady would like to book a room! She settled into a double room 2! (Thread id_35164)
Input 5 guest (letter 'L' or 'G'):
```

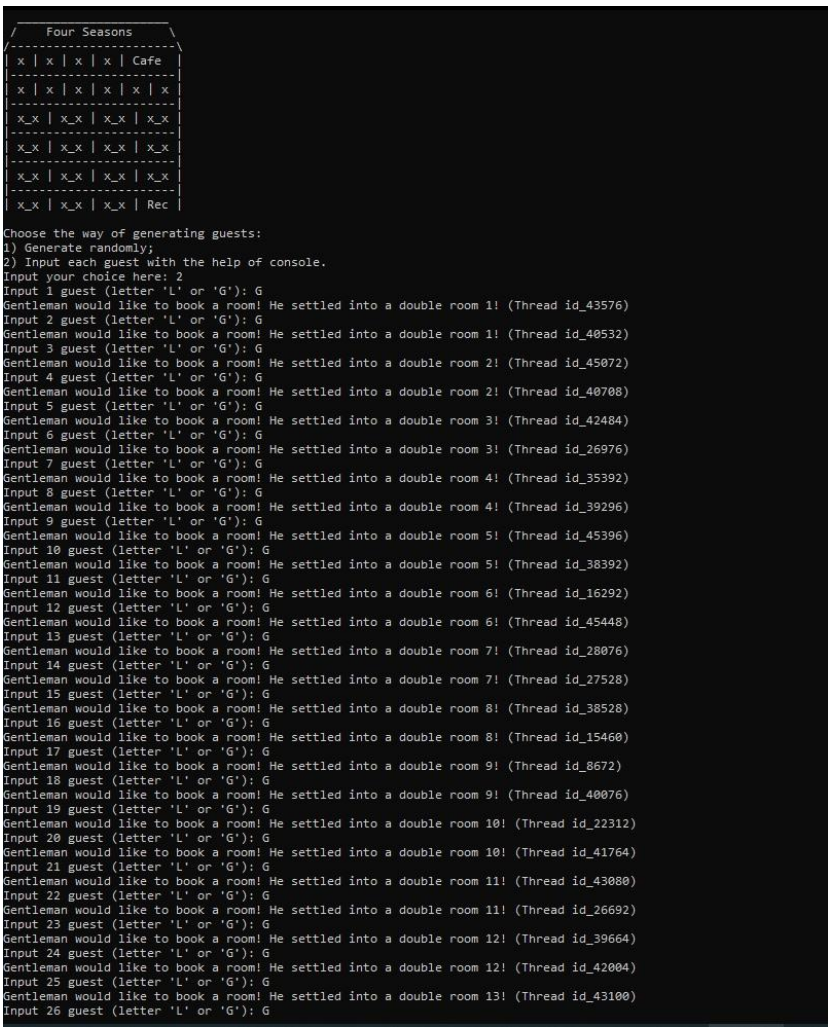
3.3 Тест № 3

Работа программы при корректных данных. В данном тесте рассматривается случай, когда приходящие гости генерируется рандомно программой.



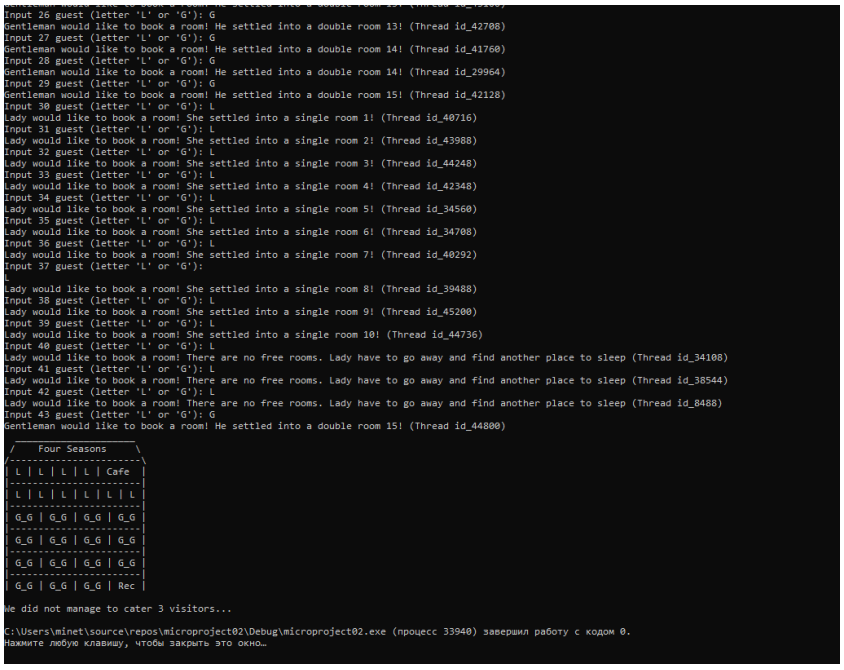
3.4 Тест № 4

Работа программы при корректных данных. В данном тесте рассматривается случай, когда приходящие гости вводятся пользователем с консоли. Также, тест демонстрирует случай, когда у нас не получилось заселить 3 гостей, которым пришлось уйти искать другое место для ночлега.



```
 _____________________
/    Four Seasons     \
|----------------------\
| x | x | x | x | Cafe  |
|----------------------|
| x | x | x | x | x | x |
|----------------------|
| x_x | x_x | x_x | x_x |
|----------------------|
| x_x | x_x | x_x | x_x |
|----------------------|
| x_x | x_x | x_x | x_x |
|----------------------|
| x_x | x_x | x_x | Rec |

Choose the way of generating guests:
1) Generate randomly;
2) Input each guest with the help of console.
Input your choice here: 2
Input 1 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 1! (Thread id_43576)
Input 2 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 1! (Thread id_40532)
Input 3 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 2! (Thread id_45072)
Input 4 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 2! (Thread id_40708)
Input 5 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 3! (Thread id_42484)
Input 6 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 3! (Thread id_26976)
Input 7 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 4! (Thread id_35392)
Input 8 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 4! (Thread id_39296)
Input 9 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 5! (Thread id_45396)
Input 10 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 5! (Thread id_38392)
Input 11 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 6! (Thread id_16292)
Input 12 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 6! (Thread id_45448)
Input 13 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 7! (Thread id_28076)
Input 14 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 7! (Thread id_27528)
Input 15 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 8! (Thread id_38528)
Input 16 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 8! (Thread id_15460)
Input 17 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 9! (Thread id_8672)
Input 18 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 9! (Thread id_40076)
Input 19 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 10! (Thread id_22312)
Input 20 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 10! (Thread id_41764)
Input 21 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 11! (Thread id_43080)
Input 22 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 11! (Thread id_26692)
Input 23 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 12! (Thread id_39664)
Input 24 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 12! (Thread id_42004)
Input 25 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 13! (Thread id_43100)
Input 26 guest (letter 'L' or 'G'): G
```



```
Input 26 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 13! (Thread id_42708)
Input 27 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 14! (Thread id_41760)
Input 28 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 14! (Thread id_29964)
Input 29 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 15! (Thread id_42128)
Input 30 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 1! (Thread id_40716)
Input 31 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 2! (Thread id_43988)
Input 32 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 3! (Thread id_44248)
Input 33 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 4! (Thread id_42348)
Input 34 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 5! (Thread id_34560)
Input 35 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 6! (Thread id_34708)
Input 36 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 7! (Thread id_40292)
Input 37 guest (letter 'L' or 'G'):
L
Lady would like to book a room! She settled into a single room 8! (Thread id_39488)
Input 38 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 9! (Thread id_45200)
Input 39 guest (letter 'L' or 'G'): L
Lady would like to book a room! She settled into a single room 10! (Thread id_44736)
Input 40 guest (letter 'L' or 'G'): L
Lady would like to book a room! There are no free rooms. Lady have to go away and find another place to sleep (Thread id_34108)
Input 41 guest (letter 'L' or 'G'): L
Lady would like to book a room! There are no free rooms. Lady have to go away and find another place to sleep (Thread id_38544)
Input 42 guest (letter 'L' or 'G'): L
Lady would like to book a room! There are no free rooms. Lady have to go away and find another place to sleep (Thread id_8488)
Input 43 guest (letter 'L' or 'G'): G
Gentleman would like to book a room! He settled into a double room 15! (Thread id_44800)
 _____________________
/    Four Seasons     \
|----------------------\
| L | L | L | L | Cafe  |
|----------------------|
| L | L | L | L | L | L |
|----------------------|
| G_G | G_G | G_G | G_G |
|----------------------|
| G_G | G_G | G_G | G_G |
|----------------------|
| G_G | G_G | G_G | G_G |
|----------------------|
| G_G | G_G | G_G | Rec |

We did not manage to cater 3 visitors...

C:\Users\minet\source\repos\microproject02\Debug\microproject02.exe (процесс 33940) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно…
```

4. Текст программы

```
/*
    Минец Максим
    БПИ-193
    Вариант 15

    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.
*/

#include <iostream>
#include <thread>
#include <string>
#include <mutex>
#include <chrono>
#include <random>

using namespace std;

constexpr int SINGLE_ROOMS = 10, DOUBLE_ROOMS = 15;
int female = -1, male = -1, lastRoom = -1, counter = 0;
mutex mut;

char* singleRoom = new char[SINGLE_ROOMS]; // NOLINT(cert-err58-cpp)
pair<char, char>* doubleRoom = new pair<char, char>[DOUBLE_ROOMS]; // NOLINT(cert-err58-
cpp)

int settleIntoDoubleRoom(bool man) {
    int last, opp;
    char client;
    if (man) {
        last = male;
        opp = female;
        client = 'G';
    }
    else {
        last = female;
        opp = male;
        client = 'L';
    }

    if (last != -1) {
        if (doubleRoom[last].second == 'x')
            doubleRoom[last].second = client;
        else {
            ++last;
            while (doubleRoom[last].second != 'x' || last == opp) ++last;

            doubleRoom[last].first = client;
        }
    }
    else {
        ++last;
        while (doubleRoom[last].second != 'x' || last == opp)
            ++last;
        doubleRoom[last].first = client;
    }

    return last;
```

```
}

void printRoomNumber(const string& person, const string& type, const int& num) {
    string pronoun;
    if (person == "Gentleman")
        pronoun = "He";
    else
        pronoun = "She";

    cout << pronoun << " settled into a " << type << " room " << num + 1 <<
        "! (Thread id_" << this_thread::get_id() << ")\n";
}

void checkedInSingleRoom(const string& person) {
    if (lastRoom >= SINGLE_ROOMS - 1) {
        mut.lock();
        ++counter;
        mut.unlock();
        cout << "There are no free rooms. " << person <<
            " have to go away and find another place to sleep (Thread id_" <<
this_thread::get_id() << ")\n";
        return;
    }

    mut.lock();
    if (person == "Gentleman")
        singleRoom[++lastRoom] = 'G';
    else
        singleRoom[++lastRoom] = 'L';
    mut.unlock();

    printRoomNumber(person, "single", lastRoom);
}

void checkedInDoubleRoom(const string& person) {
    mut.lock();
    if (person == "Gentleman")
        male = settleIntoDoubleRoom(true);
    else
        female = settleIntoDoubleRoom(false);
    mut.unlock();

    printRoomNumber(person, "double", (person == "Gentleman" ? male : female));
}

void reception(const string& person) {
    cout << person + " would like to book a room! ";
    int lastCurrent, opposite;

    if (person[0] != 'L') {
        lastCurrent = male;
        opposite = female;
    }
    else {
        lastCurrent = female;
        opposite = male;
    }

    if (lastCurrent < DOUBLE_ROOMS && (doubleRoom[DOUBLE_ROOMS - 1].second == 'x' &&
        opposite != 14 || doubleRoom[lastCurrent].second == 'x'))
        checkedInDoubleRoom(person);
    else
        checkedInSingleRoom(person);
}
```

```
void arrivalOfGuest() {
    random_device rnd;
    mt19937 mersenne(rnd());

    while (singleRoom[SINGLE_ROOMS - 1] == 'x' || doubleRoom[DOUBLE_ROOMS - 1].second ==
'x') {
        auto t = new thread(reception, mersenne() % 2 == 0 ? "Gentleman" : "Lady");
        t->join();
    }
}

void printHotel(const string& flag) {
    cout << "  _____________________\n";
    cout << " /    Four Seasons     \\\n";
    cout << "/-----------------------\\\n";


    for (int i = 0; i < SINGLE_ROOMS; ++i) {
        if (i == 4)
            cout << "| Cafe  |\n|-----------------------|\n";
        cout << "| " << singleRoom[i] << " ";
    }

    cout << "|\n|-----------------------|\n";

    for (int i = 0; i < 15; ++i) {
        if (i != 0 && i % 4 == 0)
            cout << "|\n|-----------------------|\n";

        cout << "| " << doubleRoom[i].first << '_' << doubleRoom[i].second << ' ';

    }

    cout << "| Rec |\n\n";

    if (flag == "full")
    if (counter == 0)
        cout << "We catered all guests!\n";
    else
        cout << "We did not manage to cater " << counter << " visitors...\n";
}

void emptyHotel() {
    for (int i = 0; i < SINGLE_ROOMS; ++i)
        singleRoom[i] = 'x';

    for (int i = 0; i < DOUBLE_ROOMS; ++i)
        doubleRoom[i] = { 'x', 'x' };

    printHotel("empty");
}

string checkGuestInput() {
    string check;
    cin >> check;

    while (check != "L" && check != "G" && check != "l" && check != "g") {
        cout << "Incorrect input. You should input 'G' or 'L' (Gentleman and Lady
respectively): ";
        cin >> check;
    }

    if (check == "G" || check == "g")
```

```
        check = "Gentleman";
    else
        check = "Lady";

    return check;
}

void inputFromConsole() {
    string guest;
    int i = 1;
    while (singleRoom[SINGLE_ROOMS - 1] == 'x' || doubleRoom[DOUBLE_ROOMS - 1].second ==
'x') {
        cout << "Input " << i++ << " guest (letter 'L' or 'G'): ";
        guest = checkGuestInput();
        auto t = new thread(reception, guest);
        t->join();
    }
}

int checkChoiceInput() {
    int check;
    cin >> check;
    while (check != 1 && check != 2) {
        cout << "Incorrect input. You should input '1' or '2': ";
        cin >> check;
    }
    return check;
}

void chooseInputMethod() {
    cout << "Choose the way of generating guests:\n" <<
        "1) Generate randomly;\n" <<
        "2) Input each guest with the help of console.\n" <<
        "Input your choice here: ";

    int choice = checkChoiceInput();

    if (choice == 1)
        arrivalOfGuest();
    else
        inputFromConsole();
}


int main() {
    emptyHotel();
    chooseInputMethod();
    printHotel("full");
    return 0;
}
```