

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

Факультет компьютерных наук
Образовательная программа бакалавриата «Программная инженерия»

Вариант 7

Пояснительная записка

Исполнитель
студент группы **БПИ196**
Шилова Ксения Алексеевна

13 декабря 2020 г.

Формулировка задачи

Военная задача. Анчуария и Тарантерия – два крохотных латиноамериканских государства, затерянных в южных Андах. Диктатор Анчуарии, дон Федерико, объявил войну диктатору Тарантерии, дону Эрнандо. У обоих диктаторов очень мало солдат, но очень много снарядов для минометов, привезенных с последней американской гуманитарной помощью. Поэтому армии обеих сторон просто обстреливают наугад территорию противника, надеясь поразить что-нибудь ценное. Стрельба ведется по очереди до тех пор, пока либо не будут уничтожены все цели, либо стоимость потраченных снарядов не превысит суммарную стоимость всего того, что ими можно уничтожить. Создать многопоточное приложение, моделирующее военные действия.

Описание алгоритма работы программы

Для решения программы необходимо взаимодействие двух потоков (двух разных стран). У каждого потока (страны) есть метод shoot, который моделирует разовое нападение на противника. Поясним алгоритм работы этого метода.

Пока данной государство не поразило все цели противника, либо пока не было потрачено столько снарядов, что по стоимости они превосходят всю ценность имущества противника, государство пытается атаковать. Ценность всего имущества для каждого государства задается аргументами командной строки (будут подробно описаны позже).

Возможно 2 исхода: цель была поражена или нет (промах). Меткость для каждого государства задается также аргументами командной строки (будут подробно описаны позже). Меткость – с какой вероятностью государство поражает цель.

1. Государство поражает цель.

Генерируется стоимость пораженной цели. Она может быть от 10% до 20% от стоимости всего имущества противника. Если сгенерированная стоимость цели больше, чем осталось у противника, то считается, что противник будет полностью уничтожен.

Затем вызывается метод противника, который прибавляет нанесенный в этот раз урон к общему урону (который изменяется в ходе действий).

Затем в файл (имя передано в аргументах командной строки) выводится информация о размере нанесенного урона и об оставшейся сумме имущества противника.

2. Государство промахивается.

Выводится информация о том, что произошел промах.

Все действие, описывающее один выстрел (попытку атаки) является критической секцией, так как в условии сказано, что в один момент времени должно стрелять только одно государство [2].

Затем, уже вне критической секции, подсчитывается сумма снарядов, которые были потрачены, с учетом нового выстрела. После этого поток засыпает на 10 миллисекунд. Если говорить в терминах задачи, солдатам нужно время, чтобы перезарядить минометы.

Если выполнилось одно из условий, обязывающих прекратить стрельбу, выводится информация об этом в файл (в критической секции).

Каждое государство продолжает стрелять, даже если другое уже остановилось, так как было бы не логично прекращать атаковать при собственном разорении, если все еще есть доступные снаряды и цели.

Описание аргументов командной строки

Аргументы командной строки должны быть следующие:

- 1) Вероятность попадания в цель для Анчуарии (меткость солдат Анчуарии). Вероятность должна лежать в пределах отрезка $[0,1]$.
- 2) Первоначальная стоимость всего имущества Анчуарии. Стоимость должна быть неотрицательной.
- 3) Вероятность попадания в цель для Тарантерии (меткость солдат Тарантерии). Вероятность должна лежать в пределах отрезка $[0,1]$.
- 4) Первоначальная стоимость всего имущества Тарантерии. Стоимость должна быть неотрицательной.
- 5) Имя выходного файла, где будет описана последовательность действий. Файл будет перезаписан в случае существования.

Пример: MilitaryTask 0.6 100 0.8 500 output.txt

Структура класса Country

Класс Country описывает государство, которое ведет войну (Анчуария или Тарантерия).

Поля класса (рис.1):

```
private:
    double totalPrice; //цена всего имущества
    double damage; //нанесенный урон
    double hitProbability; //вероятность попадания в цель этой страны
    double minBound, maxBound; //границы, в которых может находиться стоимость нанесенного ущерба
    double totalProjectilePrice; //сумма, потраченная на снаряды
    std::string name; //название государства
```

Рис.1 – поля класса Country

Методы класса:

- 1) Конструктор, где задаются переменные из командной строки и имя государства, границы, в которых может находиться размер урона (от 10% до 20%) (рис.2).

```
//конструктор
Country(std::string _name, double _totalPrice, double _hitProbability) {
    name = _name;
    totalPrice = _totalPrice;
    hitProbability = _hitProbability;
    damage = 0;
    //Пусть противник может поразить за один раз от 10% до 20%
    minBound = 0.1 * totalPrice;
    maxBound = 0.2 * totalPrice;
    totalProjectilePrice = 0;
}
```

Рис.2 – конструктор класса Country

- 2) void shoot(Country* rival)

Основной метод класса, моделирующий стрельбу государства по противнику, который передается в аргументах (rival). Этот метод был описан в пункте **Описание алгоритма работы программы.**

- 3) void causeDamage(double singleDamage)

Метод, прибавляющий размер разового урона к общему урону (рис.3).

```
//Метод нанесения ущерба (прибавление к общему урону)
void causeDamage(double singleDamage) {
    damage += singleDamage;
}
```

Рис.3 – метод нанесения ущерба класса Country

- 4) Методы, которые возвращают оставшуюся стоимость имущества государства, его первоначальную стоимость, минимальную и максимальную границу, в пределах которых находится разовый нанесенный урон (10% и 20% от общей стоимости) (рис.4).

```

//Метод возвращает текущую стоимость не уничтоженного имущества
double getCurrentPrice() {
    return totalPrice - damage;
}

//Метод возвращает общую стоимость имущества государства
double getTotalPrice() {
    return totalPrice;
}

double getMinBound() {
    return minBound;
}

double getMaxBound() {
    return maxBound;
}

```

Рис.4 – геттеры класса Country

Описание работы метода main

Вначале метода производится проверка корректности аргументов командной строки (рис.5).

```

//Генерируем стоимость одного снаряда
projectilePrice = std::max(totalPr1, totalPr2)/30.0;

```

Рис.5 – генерация стоимости боеприпасов

Я выбираю такой способ генерирования стоимости снаряда, чтобы даже при нулевой вероятности попадания, государство обязано было остановить попытку стрелять максимум на 30 по счету попытке.

Затем генерируются страны с помощью конструктора и необходимых параметров (рис.6).

```

Country* Anchuarria = new Country("Anchuarria", totalPr1, hitProb1);
Country* Taranteria = new Country("Taranteria", totalPr2, hitProb2);

```

Рис.6 – метод создание стран через конструктор

Затем открывается поток для записи в файл (рис.7).

```

fout.open(argv[5]); //открываем поток для записи

```

Рис.7 – поток для записи

Затем запускаются методы потоков каждого государства (рис.8).

```
std::thread AnchuariaThread(&Country::shoot, Anchuaria, Taranteria);  
std::thread TaranteriaThread(&Country::shoot, Taranteria, Anchuaria);
```

Рис.8 – метод создание потоков

Формат: метод; объект, метод которого вызывается; аргумент метода (противник).

Затем выполняются заключительные необходимые действия (рис. 9).

```
AnchuariaThread.join();  
TaranteriaThread.join();  
std::cout << std::endl << "The war is over!" << std::endl;  
fout.close();  
delete Anchuaria;  
delete Taranteria;  
return;
```

Рис.9 – заключительные действия

Способ генерации случайного числа

Мне бы хотелось отдельно упомянуть генерацию случайных чисел, так как в процессе работы возникли проблемы с их генерацией в разных потоках.

Если генерировать случайное число в отрезке [min; max) стандартным способом, то есть $(\text{rand}()/\text{RND_MAX}) * (\text{max} - \text{min}) + \text{min}$, то генерация чисел в двух потоках будет одинаковая. То есть какой урон Анчуария нанесла Тарантерии, такой урон и Тарантерия нанесет Анчуарии (в процентах), так как в классе один метод, который разные потоки выполняют с разными аргументами. Так как потоки запускаются единовременно, то генерация `rand()` начнется с одного и того же числа, и будет перебирать одни и те же варианты.

Примерная картина происходящего следующая (Все действия происходят одинаково в двух потоках) (рис.10):

```
x86_x64 Cross Tools Command Prompt for VS 2019
out: MilitaryTask.exe
MilitaryTask.obj

\Users\Пользователь\Desktop\C++\avs>MilitaryTask 0.5 100 0.5 100
Incorrect format of probability argument. Probability must be real number between 0 and 1
Inchuaria caused damage to the rival in the amount of 15.6359
Now total price of rival is 84.3641
Aranteria caused damage to the rival in the amount of 15.6359
Now total price of rival is 84.3641
Inchuaria caused damage to the rival in the amount of 18.0874
Now total price of rival is 66.2767
Aranteria caused damage to the rival in the amount of 18.0874
Now total price of rival is 66.2767
Aranteria missed
Inchuaria missed
Aranteria caused damage to the rival in the amount of 13.5029
Now total price of rival is 52.7738
Inchuaria caused damage to the rival in the amount of 13.5029
Now total price of rival is 52.7738
Aranteria missed
Inchuaria missed
Inchuaria missed
Aranteria missed
Inchuaria missed
Aranteria missed
Inchuaria caused damage to the rival in the amount of 18.5894
Now total price of rival is 34.1844
Aranteria caused damage to the rival in the amount of 18.5894
Now total price of rival is 34.1844
Inchuaria missed
Aranteria missed
Aranteria missed
Inchuaria missed
Aranteria caused damage to the rival in the amount of 10.1498
Now total price of rival is 24.0346
```

Рис.10 – описание ошибочного генерирования числа

Чтобы обойти эту проблему я создаю локальный генератор случайных чисел для потока.

```
std::default_random_engine gen(std::chrono::system_clock::now().time_since_epoch().count());
std::uniform_real_distribution<> distribution1(0, 1), distribution2(rival->getMinBound(), rival->getMaxBound());
```

Рис.11 – локальный для потока генератор случайных чисел

Идея описана на сайте <http://www.quizful.net/post/random-number-generation-in-cpp11> [1] .

Тестирование

1. Некорректные аргументы командной строки

1.1 Некорректная длина аргументов

```
C:\Users\Пользователь\Desktop\C++\avs>MilitaryTask 0.7 100 0.6
Incorrect number of arguments: 4
There must be 5 arguments: hit probability and total price for Anchuria; hit probability and to
tal price for Taranteria; name of output file
```

1.2 Некорректный формат

```
C:\Users\Пользователь\Desktop\C++\avs>MilitaryTask 0.7 100 0.6 a68g output.txt
Incorrect format of arguments. It must be real number
```

1.3 Некорректный диапазон

```
C:\Users\Пользователь\Desktop\C++\avs\MilitaryTask 1.7 100 0.6 200 output.txt
Incorrect format of probability argument. Probability must be real number between 0 and 1
```

2. Корректные входные данные.

2.1 У одной из стран нулевая вероятность попасть в цель

```
>MilitaryTask 0 100 0.6 200 output.txt
```

```

Anchuaria missed
Taranteria missed
Anchuaria missed
Taranteria missed
Anchuaria missed
Taranteria caused damage to the rival in the amount of
16.7926
    Now total price of rival is 83.2074
Anchuaria missed
Taranteria caused damage to the rival in the amount of
15.3352
    Now total price of rival is 67.8722
Anchuaria missed
Taranteria caused damage to the rival in the amount of
16.4152
    Now total price of rival is 51.457
Anchuaria missed
Taranteria missed
Anchuaria missed
Taranteria caused damage to the rival in the amount of
17.7878
    Now total price of rival is 33.6692
Anchuaria missed
Taranteria caused damage to the rival in the amount of
15.7053
    Now total price of rival is 17.9639
Anchuaria missed
Taranteria missed
Anchuaria missed
Taranteria missed
Anchuaria missed
Taranteria caused damage to the rival in the amount of
10.2696
    Now total price of rival is 7.69433

```

```
Anchuaria missed  
Taranteria missed  
Anchuaria missed  
Taranteria missed  
Anchuaria missed  
Taranteria missed  
Anchuaria missed  
Taranteria caused damage to the rival in the amount of  
7.69433  
Now total price of rival is 0  
Anchuaria missed  
ATTENTION!!! Taranteria destroyed all rival targets  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
Anchuaria missed  
ATTENTION!!! Anchuaria has spent too much projectiles:  
206.667 (more than total price of the rival)
```

Все происходит закономерно: Анчуария потратила все доступные боеприпасы и остановила попытки попасть в цель.

2.2 Одна из стран всегда попадает в цель (с вероятностью 1)

```
>MilitaryTask 1 100 0.6 200 output.txt
```

```
Anchuaria caused damage to the rival in the amount of 33.0069
  Now total price of rival is 166.993
Taranteria caused damage to the rival in the amount of
14.5926
  Now total price of rival is 85.4074
Taranteria caused damage to the rival in the amount of
16.9746
  Now total price of rival is 68.4328
Anchuaria caused damage to the rival in the amount of 21.6786
  Now total price of rival is 145.314
Taranteria caused damage to the rival in the amount of
12.5776
  Now total price of rival is 55.8552
Anchuaria caused damage to the rival in the amount of 23.5218
  Now total price of rival is 121.793
Anchuaria caused damage to the rival in the amount of 25.071
  Now total price of rival is 96.7216
Taranteria missed
Taranteria caused damage to the rival in the amount of
19.3277
  Now total price of rival is 36.5275
Anchuaria caused damage to the rival in the amount of 32.1937
  Now total price of rival is 64.5279
Taranteria missed
Anchuaria caused damage to the rival in the amount of 35.2694
  Now total price of rival is 29.2586
Taranteria caused damage to the rival in the amount of
13.9384
  Now total price of rival is 22.5891
Anchuaria caused damage to the rival in the amount of 29.2586
  Now total price of rival is 0
Taranteria missed
ATTENTION!!! Anchuarria destroyed all rival targets
Taranteria caused damage to the rival in the amount of
18.8086
  Now total price of rival is 3.78047
Taranteria caused damage to the rival in the amount of
3.78047
  Now total price of rival is 0
ATTENTION!!! Taranteria destroyed all rival targets
```

Анчуария ни разу не промахнулась и уничтожила все цели быстрее, чем Тарантерия.

2.3 Равные вероятности попасть в цель у Анчуарии и Тарантерии

```
>MilitaryTask 0.6 100 0.6 200 output.txt
```

```

Anchuaria caused damage to the rival in the amount of 37.5089
  Now total price of rival is 162.491
Taranteria caused damage to the rival in the amount of 17.5845
  Now total price of rival is 82.4155
Taranteria caused damage to the rival in the amount of 18.0423
  Now total price of rival is 64.3731
Anchuaria caused damage to the rival in the amount of 20.4676
  Now total price of rival is 142.023
Taranteria missed
Anchuaria missed
Taranteria missed
Anchuaria caused damage to the rival in the amount of 33.1418
  Now total price of rival is 108.882
Taranteria missed
Anchuaria missed
Anchuaria missed
Taranteria caused damage to the rival in the amount of 13.4184
  Now total price of rival is 50.9548
Taranteria caused damage to the rival in the amount of 19.529
  Now total price of rival is 31.4258
Anchuaria caused damage to the rival in the amount of 32.0302
  Now total price of rival is 76.8515
Taranteria missed
Anchuaria caused damage to the rival in the amount of 38.2393
  Now total price of rival is 38.6122
Anchuaria caused damage to the rival in the amount of 38.4773
  Now total price of rival is 0.134891
Taranteria caused damage to the rival in the amount of 18.6209
  Now total price of rival is 12.8049
Taranteria caused damage to the rival in the amount of 12.8049
  Now total price of rival is 0
Anchuaria caused damage to the rival in the amount of 0.134891
  Now total price of rival is 0
ATTENTION!!! Taranteria destroyed all rival targets
ATTENTION!!! Anchuaria destroyed all rival targets

```

Как предполагалось, страны уничтожили друг друга почти одновременно.

Список используемых источников

1. <http://www.quizful.net/post/random-number-generation-in-cpp11>
2. <https://en.cppreference.com/w/cpp/thread/mutex>