

## Реализовать класс, объявленный в заголовочном файле

Вам дан заголовочный файл `phone_number.h`, содержащий объявление класса `PhoneNumber`. При этом в комментариях описано поведение, которое ожидается от реализации этого класса.

```
1  #pragma once
2  #include <string>
3  using namespace std;
4  class PhoneNumber {
5  public:
6  /* Принимает строку в формате +XXX-YYY-ZZZZZZ
7  Часть от '+' до первого '-' - это код страны.
8  Часть между первым и вторым символами '-' - код города
9  Всё, что идёт после второго символа '-' - местный номер.
10 Код страны, код города и местный номер не должны быть пустыми.
11 Если строка не соответствует этому формату, нужно выбросить исключение invalid_argument.
    Проверять, что номер содержит только цифры, не нужно.
12 Примеры:
    • +7-495-111-22-33
    • +7-495-1112233
    • +323-22-460002
    • +1-2-coursera-cpp
    • 1-2-333 - некорректный номер - не начинается на '+'
    • +7-1233 - некорректный номер - есть только код страны и города
13 */
14 explicit PhoneNumber(const string &international_number);
15 string GetCountryCode() const;
16 string GetCityCode() const;
17 string GetLocalNumber() const;
18 string GetInternationalNumber() const;
19 private:
20 string country_code_;
21 string city_code_;
22 string local_number_;
23 };
```

Пришлите `cpp`-файл с **определениями методов** класса `PhoneNumber`, которые реализуют описанное поведение.

### Как будет тестироваться ваш код

Автоматическая тестирующая система добавит ваш `cpp`-файл в проект, содержащий другой `cpp`-файл с юнит-тестами для класса `PhoneNumber`, а также файл `phone_number.h`. Затем она соберёт этот проект и, если компиляция будет выполнена успешно, запустит получившийся исполняемый файл. Если он завершится успешно (т. е. юнит-тесты не найдут ошибок в вашей реализации), то ваша посылка будет засчитана.