# Задание по программированию: Декомпозиция программы — 2

Условие

В задаче [«Декомпозиция программы»](https://www.coursera.org/learn/c-plus-plus-yellow/programming/k6Xm2/diekompozitsiia-proghrammy) мы разбили монолитный код на набор функций и классов. Теперь мы сделаем ещё один шаг и разделим нашу программу на несколько файлов. В этой задаче вам нужно создать проект, состоящий из следующих файлов:

1.query.h, в него кладём:

* enum class QueryType
* struct Query
* объявление istream& operator >> (istream& is, Query& q)

2. query.cpp, в него кладём

* определение istream& operator >> (istream& is, Query& q);

3. responses.h:

* struct BusesForStopResponse
* ostream& operator << (ostream& os, const BusesForStopResponse& r)
* struct StopsForBusResponse
* ostream& operator << (ostream& os, const StopsForBusResponse& r)
* struct AllBusesResponse
* ostream& operator << (ostream& os, const AllBusesResponse& r)

4. responses.cpp: определения всего, что объявлено в responses.h

5. bus\_manager.h: объявление класса BusManager

6. bus\_manager.cpp: определения методов класса BusManager

7. main.cpp: функция main

## Как отправлять проект на проверку

В заданиях по программированию Coursera позволяет отправлять на проверку только один файл, поэтому для того, чтобы сдать свой проект в тестирующую систему, вам надо запаковать все свои файлы в архив и отправить его на проверку. Поддерживаемые форматы архивов: rar, zip, 7z, tar, tar.gz, а также многие другие (для распаковки мы используем программу [dtrx](http://manpages.ubuntu.com/manpages/trusty/en/man1/dtrx.1.html), на странице с её описанием приведён полный список поддерживаемых форматов).

## Как будет тестироваться ваше решение

Ниже вам дана заготовка для файла main.cpp, содержащая функцию main. Это точно такая же заготовка, которая была дана в задаче «Декомпозиция программы». Тестирование вашего решения будет выполняться в два этапа. На первом этапе автоматическая тестирующая система распакует присланный вами архив и соберёт извлечённые файлы в исполняемый файл. При этом функция main в вашем проекте будет заменена на ту, которая дана в заготовке файла main.cpp. Затем этот исполняемый файл будет запущен на наборе тестов. Тестирование выполняется так же, как и для большинства задач на нашем курсе: тест подаётся в stdin, замеряется время выполнения программы, а затем анализируется stdout.

На втором этапе будет выполняться тестирование отдельных файлов вашего проекта. Проверяется, что файл query.h действительно содержит перечислимый тип QueryType, что BusManager::GetBusesForStop возвращает корректно сформированный объект BusesForStopResponse т.д.

Заготовка файла main.cpp:

[main.cpp](https://d3c33hcgiwev3.cloudfront.net/_253c2923ab70e0bccccc3007c3d1a673_main.cpp?Expires=1616976000&Signature=Hc-GITW3fnkyconeiWEhM138oWcBKlHLh3kUy-X0PIlioxfmKQNJ4M-TDrv69bGJVs8dOjtdmETLmsOv53coBwXxBKnzB2dx-7MnCX7wiD1DY6t-vqvRLRz5iG2hd3GPfQa~6c6m5BmsP5BpcC1-KLmGYiFnPWpepCcj93ebysU_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A)