

Задан CSV-файл ([comma-separated values](#)) с заголовком, в котором перечислены названия столбцов. Строки нумеруются целыми положительными числами, необязательно в порядке возрастания. В ячейках CSV-файла могут храниться или целые числа или выражения вида

`= ARG1 OP ARG2`

где ARG1 и ARG2 – целые числа или адреса ячеек в формате Имя_колонки Номер_строки, а OP – арифметическая операция из списка: +, -, *, /.

Например, таблица

	A	B	Cell
1	1	0	1
2	2	=A1+Cell130	0
30	0	=B1+A1	5

Будет представлена в нашем CSV-формате следующим образом:

```
,A,B,Cell
1,1,0,1
2,2,=A1+Cell130,0
30,0,=B1+A1,5
```

(обратите внимание на *пропуск первого значения* в первой строке CSV-представления, он обозначает пустую левую верхнюю ячейку таблицы).

Требуется написать программу, которая читает произвольную CSV-форму из файла (количество строк и столбцов может быть любым), вычисляет значения ячеек, если это необходимо, и выводит получившуюся табличку в виде CSV-представления в консоль. Решением задания будет: файл или несколько файлов с исходным кодом программы на языке C++, инструкции по сборке и тестовые примеры (количество тестов – на усмотрение разработчика).

Программа должна компилироваться компилятором gcc или clang в Linux, mingw/cygwin в Windows.

Рекомендуется использование стандартной библиотеки (STL). Использование любых сторонних библиотек, кроме STL, запрещено. Имя CSV-файла необходимо передавать первым аргументом командной строки. В решении, кроме файлов с исходным кодом, требуется предоставить инструкции по компиляции программы для проверки. Можно, но необязательно, использовать следующие системы автоматизации сборки: make, automake, cmake, gradle; проприетарные средства сборки не допускаются: например, файлы проектов MS Visual Studio не подойдут.

Пример запуска программы:

```
$ csvreader.exe file.csv
```

где csvreader.exe – имя исполняемого файла (может быть любым), file.csv – имя файла с таблицей (произвольно задаётся пользователем), \$ - приглашение командного интерпретатора.

Для приведённого примера результат, напечатанный в консоль, может быть таким:

```
,A,B,Cell
1,1,0,1
2,2,6,0,
30,0,1,5
```

Рекомендуем решение задачи проверять

- На приведённом примере
- На дополнительных тестах (csv-файлах), выбор которых отводится на ваше усмотрение; постарайтесь в своих тестах учесть различные правильные и ошибочные ситуации (например, к ошибочным можно отнести деление на 0, некорректные адреса ячеек, неверные номера строк, неправильный формат таблицы и т.п.)
- Можно (но необязательно) написать дополнительно проверочные юнит-тесты с использованием доступной библиотеки (Google Test, CppUTest, и т.п.)

Рекомендуется в решении задания использовать современные возможности C++, структуры и типы данных из стандартной библиотеки (STL). Будьте готовы защитить своё решение тестового задания на этапе очного интервью (ответить на вопросы технических специалистов по домашнему заданию или по смежным темам).

Проверка домашнего задания осуществляется в два этапа:

1. Обработка наших эталонных тестовых CSV-файлов, содержащих как корректные, так и некорректные таблицы.
2. Оценивание исходных кодов программы проверяющими.