Муниципальное автономное общеобразовательное учреждение

«Лицей инновационных технологий»

**Пояснительная записка к решению задания отборочного этапа хакатона «IT-Education HACK» по направлению «Искусственный интеллект».**

Выполнили: Кучун Дмитрий, Свиридов Георгий, Павлюченко Артем, Павлов Александр

Руководитель проекта: Назина Александра Александровна, учитель

информатики и ИКТ.

Хабаровск, 2023 г.

Оглавление

[Введение 3](#_Toc161860723)

[Уровень 1 4](#_Toc161860724)

[Задание 1 4](#_Toc161860725)

[Задание 2 4](#_Toc161860726)

[Задание 3 4](#_Toc161860727)

[Задание 4 5](#_Toc161860728)

[Задание 5 5](#_Toc161860729)

[Уровень 2 5](#_Toc161860730)

[Задание 6 5](#_Toc161860731)

[Задание 7 5](#_Toc161860732)

[Задание 8 5](#_Toc161860733)

[Задание 9 5](#_Toc161860734)

[Уровень 3 5](#_Toc161860735)

# Введение

Для начала работы мы использовали стандартную библиотеку Python под названием csv, которая позволяет работать с файлами этого типа. С ее помощью мы переводим данные из students\_data.csv в двумерный массив data, на основе которого строится вся дальнейшая работа.

Далее создается словарь main\_dict, содержимое которого представляет такой вид: {номер колонки: «название колонки»} (учитывая, что в массивах отсчет идет с нуля). Данный словарь помогает в решении задания 1.2, 2.1 и других.

Позже для читаемости вывода создается функция \_separation, которая вызывается после вызова функций-решений заданий. Она из себя представляет просто вывод двадцати нижних подчеркиваний (\_).

# Уровень 1

## Задание 1

Задание 1.1: Создание двух массивов с соответствующими названиями (cat\_features, num\_features). В cat\_features находятся названия категориальных признаков, а в num\_features числовых соответственно.

Задание 1.2: Была создана функция mistakes, в котором создается DataFrame с помощью библиотеки pandas, которая принимает данные из файла students\_data.csv (исходный файл таблицы). После чего на основе условий элементов каждого признака функция проверяет правильность исходных данных. В результате работы функция возвращает количество ошибок, учитывая также пропуски, но именно количество опечаток с ошибками выводится, используя функцию misses (задание 2.1).

Примечание: При нахождении ошибок и опечаток мы выводили каждую ошибку, относящуюся к тому или иному признаку. На основе полученных данных мы меняли регистр или символ на правильный, используя функцию replace().

Задание 1.3: Для определения ошибок и опечаток в таблице элементы некоторых признаков были приведены к единому типу (float, int)

## Задание 2

Задание 2.1: Создается функция misses, которая принимает как аргумент двумерный массив data с исходными данными из файла students\_data.csv. Далее она считает общее количество пропусков и возвращает это значение, а также она с помощью словаря main\_dict возвращает признаки, в которых были найдены пропуски.

Задание 2.2: Как уже было сказано, функция misses возвращает признаки, в которых были пропущены значения. Ими оказались: Dalc, Walc, cheating, famrel, romantic. В Dalc и Walc мы решили заполнять пропуски минимально возможным значением, потому что, если человек не заполнил этот признак, значит значение этого показателя равно минимальному - единице. В famrel прослеживается такая же логика. А в cheating и romantic работает так: если человек не указал в этой графе ничего, значит значение равно no. Все исправления пропусков прописаны в функции correction, которая далее помогает решать другие задания. Она вызывается и исправляет все пропуски после вызовов функций, отвечающих за решения первых заданий

## Задание 3

Задание 3.1: Пропущено

Задание 3.2: Была создана функция count\_schools, принимающая как аргумент двумерный массив data. Итерируясь по массиву, функция считает количество учащихся в частной и государственной школах, возвращая число для каждого показателя.

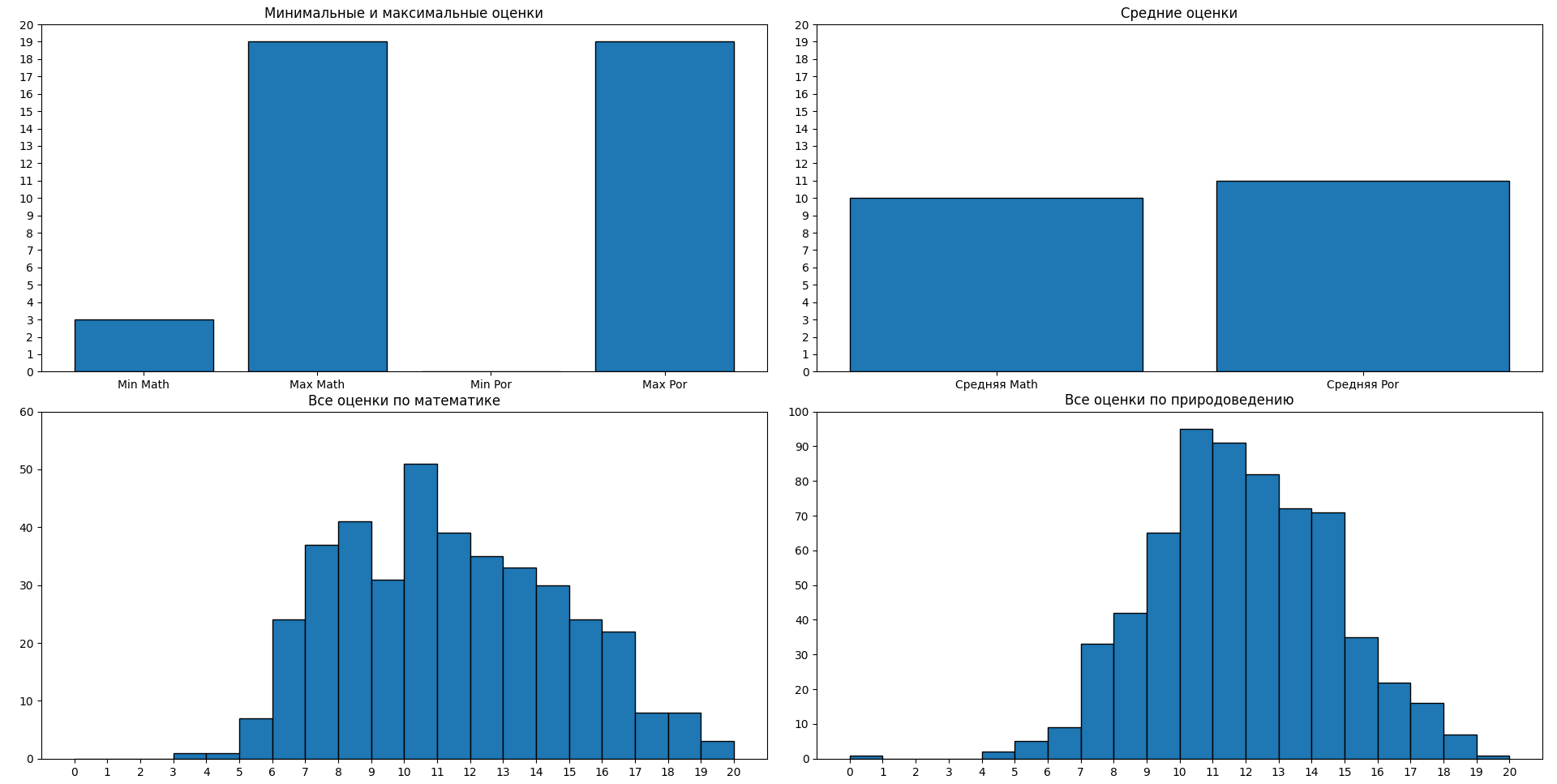
Задание 3.3: Была создана функция count\_students, принимающая как аргумент двумерный массив data. Итерируясь по массиву, функция считает количество учащихся на природоведенье и математике, возвращая число для каждого показателя.

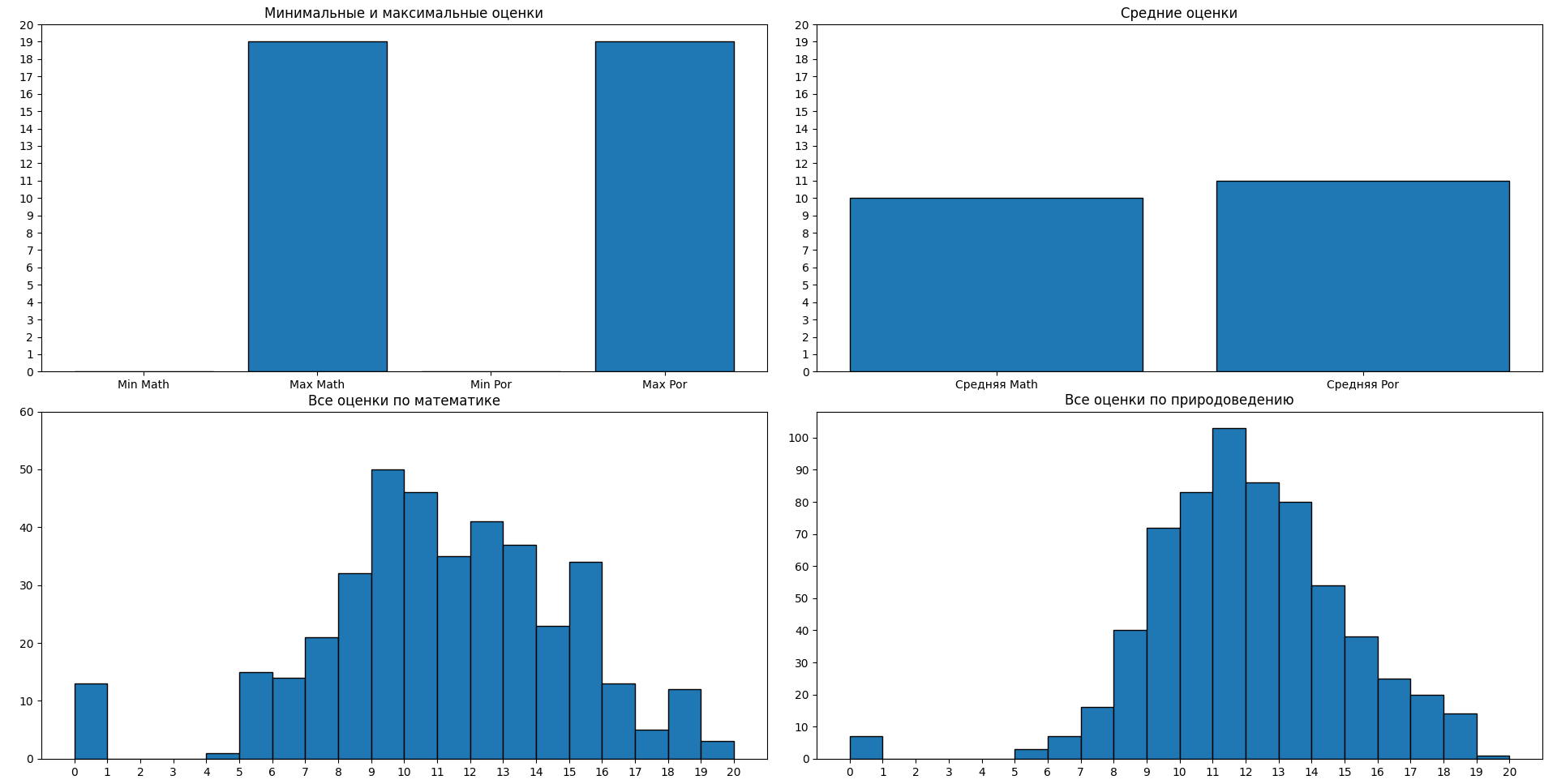
Задание 3.4: Была создана функция amount\_failures, принимающая как аргумент двумерный массив data. Итерируясь по массиву, функция считает количество учащихся, которые завалили предметы, отдельно математику и природоведение. Функция возвращает количество завалов по каждому предмету и в результате простого сравнения выводит строку с тем, какой предмет все же заваливают чаще.

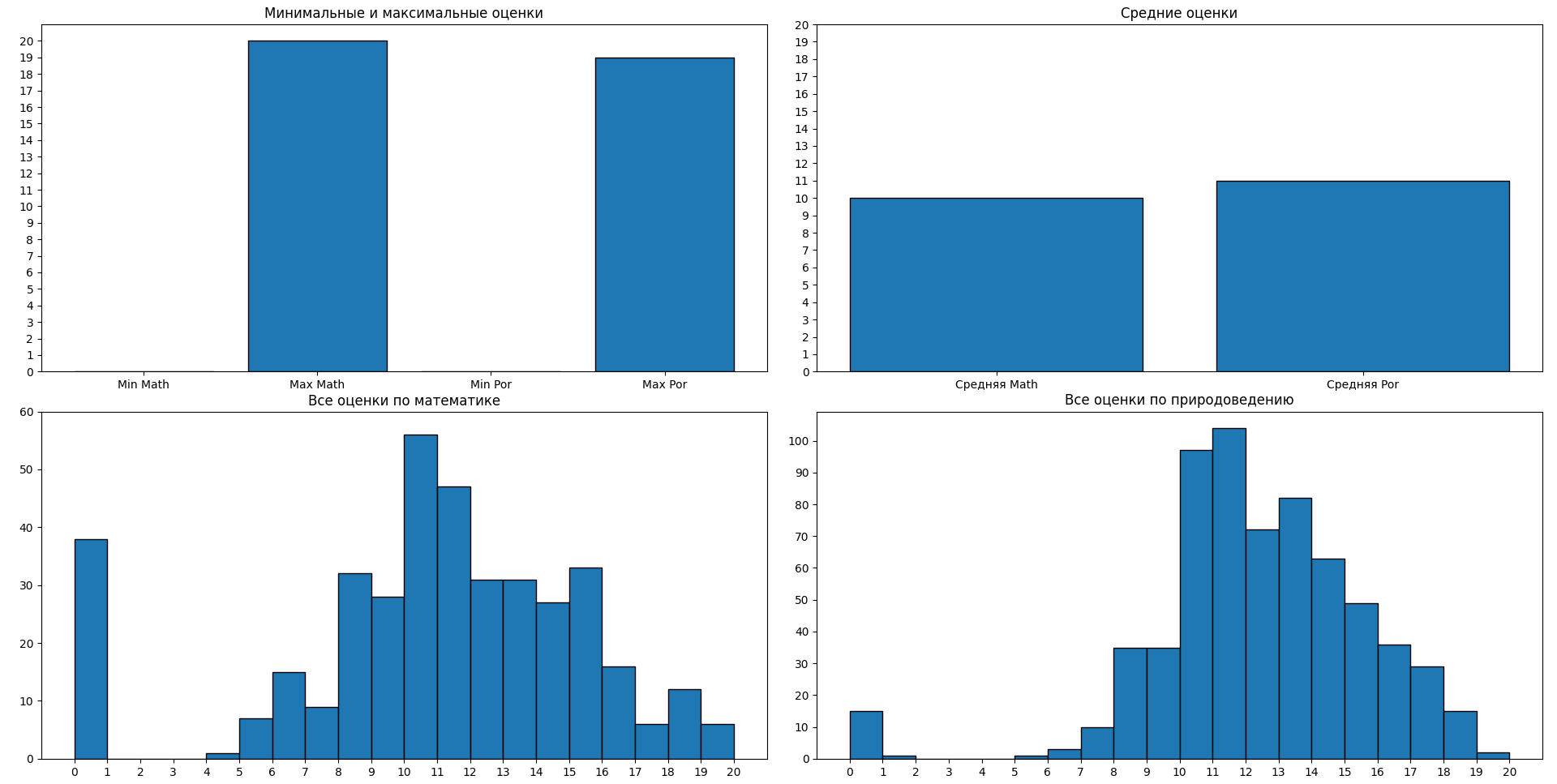
Задание 3.5: Была создана функция amount\_studytime, принимающая как аргумент двумерный массив data. Итерируясь по массиву, функция считает сколько в среднем уходит на учебу по отдельным предметам у учащихся. Функция возвращает это.

Задание 3.6: Была создана функция amount\_reasons, принимающая как аргумент двумерный массив data. Итерируясь по массиву, функция считает разные причины учащихся выбирать школу частную/государственную. Функция сравнивает количество причин для выбора школ и возвращает чего больше

## Задание 4

Создается функция min\_max\_avg\_mark, которая принимает как аргументы двумерный массив data и G (номер семестра). С помощью нетрудных вычислений программа возвращает такие переменные как: min\_math, max\_math, avg\_math, min\_por, max\_por, avg\_por, math\_grades, por\_grades, которые обозначают худшую оценку по математике, наилучшую оценку по математике, среднюю оценку по математике, худшую оценку по природоведению, наилучшую оценку по природоведению, среднюю оценку по природоведению, все оценки соответственно. Все эти данные определены для выбранного пользователем семестра. Далее используя библиотеку matplotlib на одном рисунке, создается четыре диаграммы с данными:





На основе этих данных можно сказать, что средняя оценка по природоведению выше на 1 балл во всех семестрах и по году, что нам свидетельствует о том, что среднее качество знаний по природоведению выше, чем среднее качество знаний по математике (хотя и на 1 балл). Наилучшая оценка по математике и природоведению по результатам обоих семестров одинаковая: 20, что говорит о том, что все же на обоих предметах есть отличники, но по году абсолютного отличника по природоведению нет, потому что как мы можем видеть, наилучшая оценка по году по природоведению: 19, а по математике: 20.

Отдельно стоит рассмотреть распределение оценок по каждому предмету для каждого семестра. Во, первых (это можно так же пронаблюдать в решении задания 3.3) на природоведении учится в разы больше людей.

G1: Наибольшее количество оценок по математике находится в диапазоне 7 – 14 и вообще можно пронаблюдать такую тенденцию, что от 0 до 10 количество оценок растет (за исключением оценки 9), а от 11 до 20 количество оценок уже уменьшается; у природоведения же другая ситуация: до 11 количество оценок распределено хаотично, а от 11 до 20 идет вниз.

G2: Теперь все наоборот: по математике оценки до 9 хаотично распределены, а от 9 до 20 идут вниз; по природоведению количество оценок от 0 до 11 растут по параболе (за исключением 0) и от 11 до 20 идут вниз.

G3: Очень заметным на графике математике оказывается количество оценок 0 (≈ 37 человек), и в общем итоговое распределение оценок крайне хаотично, какую либо закономерность мы можем наблюдать только на участке от 10 до 20: там количество оценок идет примерно вниз (но все же не равномерно).

## Задание 5

Задание 5.1:

Задание 5.2:

# Уровень 2

## Задание 6

Задание 6.1: Была создана функция time\_on\_street, принимающая как аргумент двумерный массив data. Функция, используя этот массив, отделяет от него нужные для решения колонки, а именно sex, Dalc, Walc. Далее итерируясь по двумерному массиву, функция считает время на улице отдельно для девочек и мальчиков. В результате функция, посчитав время, выводит кто проводит время на улице больше.

Задание 6.2:

Задание 6.3:

## Задание 7

## Задание 8

Задание 8.1: Была создана функция both\_subjects, принимающая как аргумент двумерный массив data. Функция, используя этот массив, отделяет от него нужные для решения колонки: sex, age, address, famsize, Pstatus, Medu, Fedu, Mjob, Fjob, guardian, activities, nursery, higher, internet, romantic, famrel, goout, Dalc, Walc, health, добавляя их в отдельный массив sliced\_data. Мы знаем, что колонка ID - уникальная для каждого наблюдения, но не человека, из чего мы делаем вывод, что, сравнивая строчки по определённым признакам, мы сможем найти одинаковые наблюдения, а следовательно, одного человека, учащегося на обоих курсах. Функция итерируется по двумерному массиву сравнивая строчки, находя схожести он считает их в отдельной переменной. Функция выводит строчку «Количество людей, учащихся по обоим предметам: {num\_people}», которая нам и показывает, сколько людей учится на обоих курсах.

Задание 8.2: В старую функцию both\_subjects мы добавляем счетчик сравнения оценок. Функция возвращает 3 строки в которых написано сколько человек лучше знает природоведение, сколько математику, а сколько и то, и другое одинаково. Из результатов работы функции мы можем сделать вывод, что люди, учащиеся на обоих курсах, в основном лучше знают природоведение, чем математику, и совсем маленькая часть из этих людей знают природоведение и математику одинаково.

## Задание 9

Была создана функция G4, принимающая как аргумент исходный двумерный массив data. Функция создает одноименный массив, в который записываются значения по условию. В результате функция возвращает этот массив и после используя эту функцию добавляет колонку в исходный двумерный массив data.

# Уровень 3

## Задание 10

Мы предлагаем схожую систему оценивания G4 по принципу 9 задания, но с изменением оценки за G3 (исходное G3 остается, но добавляется доп. колонка G3\_dif), учитывая другие показатели такие как: failures, absences и cheating. Один «завал» принимается за вычет 1 балла из оценки по году (G3), один пропуск принимается за вычет 0,25 балла из оценки по году (G3), и факт списывания принимается за вычет пяти баллов от итоговой оценки по году (G3). Реализуется такая формула с помощью функции G3\_dif,которая принимает как аргумент двумерный массив data, далее по принципу решения 9 задания инициализируется список с одноименным названием, создается двумерный массив sliced\_data со срезанным двумерным массивом, в котором остается колонки failures, absences, cheating, G3. Мы итерируемся по этому массиву, где заменяем ‘yes’ и ‘no’ в cheating на 5.0 и 0.0 соответственно. Далее по выше предложенной схеме (формуле) мы добавляем в массив G3\_dif новую оценку. Также мы добавляем условие, что если оценка становится отрицательной, то она приравнивается к нулю. Функция возвращает список G3\_dif и потом с помощью двух строчек кода добавляет его в исходный массив data. После создается функция G4\_dif аналогичная функции G4, которая по такому же принципу ставит оценки по 4х бальной системе.

# Заключение

После всей проделанной работы, мы используем ту же библиотеку csv, что и в начале работы, но уже для записи двумерного массива data в новый файл corrected\_data.scv, т. к. в процессе в этот массив добавилась одна колонка G4 и были заполнены пропуски и исправлены ошибки.