**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ МОРСКОЙ**

**ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

(СПб ГМТУ)

**Лабораторная работа №1**

По дисциплине:

**«Программирование»**

Специальность 09.03.01.03

Подготовил студент группы 20121:

Наталенко И. А.

Проверил преподаватель:

Поделенюк П.П.

Санкт-Петербург

2022 г.

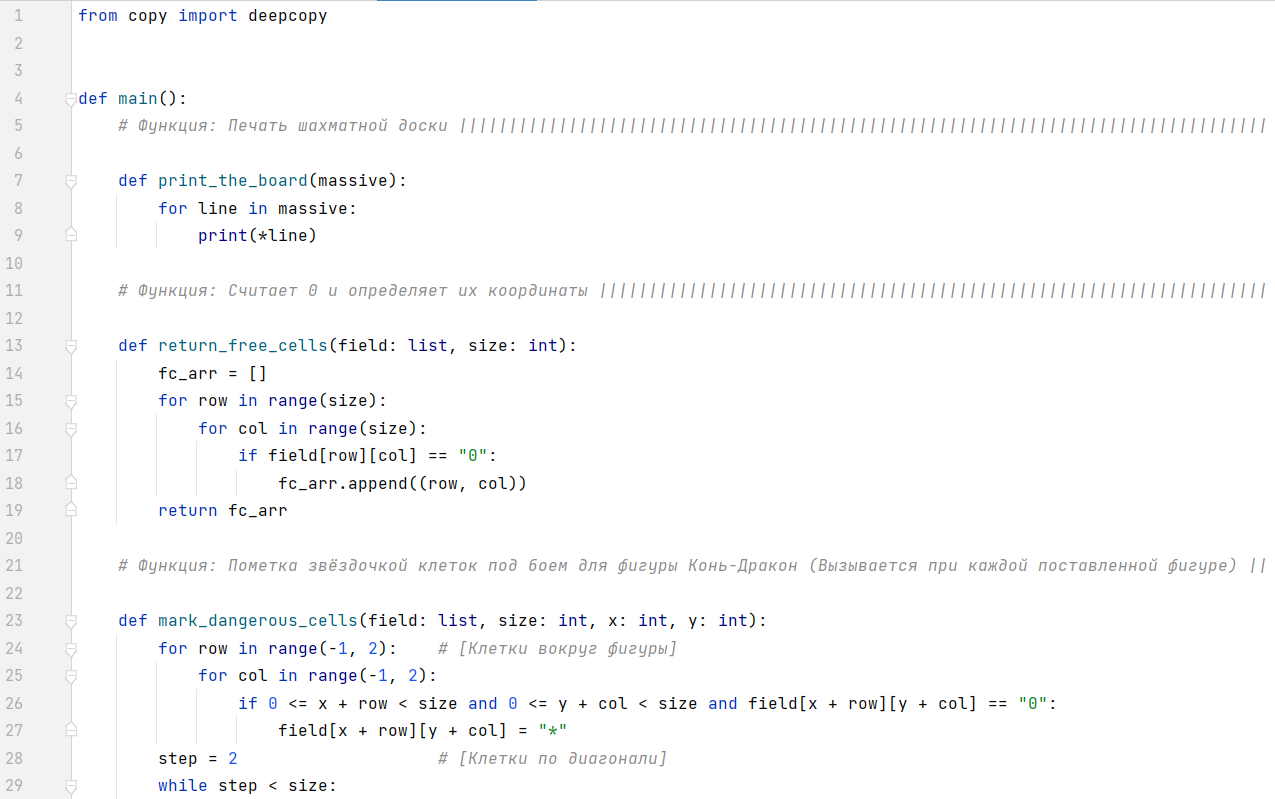
**Цель работы**:

Научиться применять на практике знания по работе с файлами, работе с несколькими функциями, работе по нисходящему методу, работе с локальными и глобальными переменными и применению рекурсии.

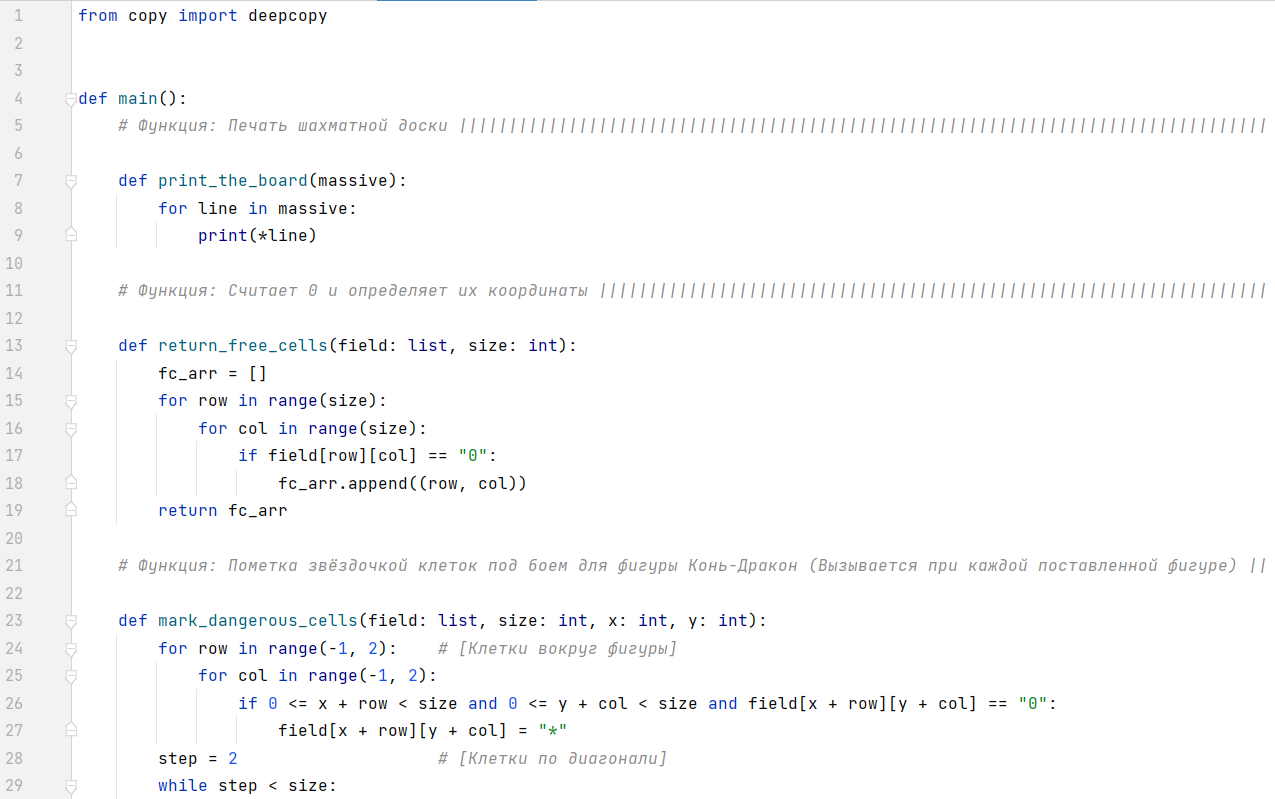
**Ход работы**:

Я написал функции, облегчающие дальнейшую работу с данными.

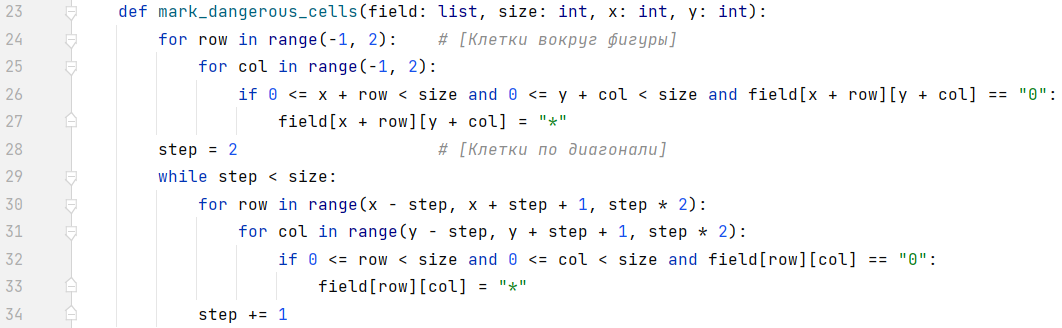
Функция print\_the\_board принимает двухмерный массив (шахматную доску) и выводит его в консоль в удобном формате – через пробелы.



Функция return\_free\_cells принимает ту же шахматную доску и её размер, проверяет, находятся ли клетки под боем и возвращает список кортежей, в которые записаны координаты свободных клеток.

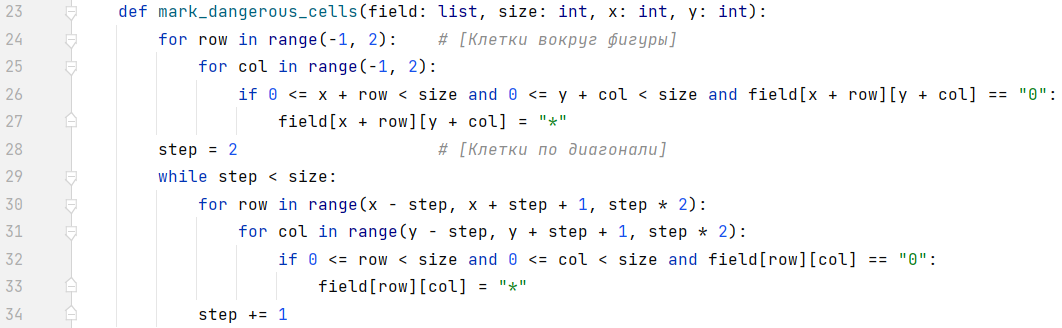


Функция mark\_dangerous\_cells принимает доску, её размер, а также координаты, куда до вызова ставится фигура и отмечает звёздочками клетки, находящиеся под боем.

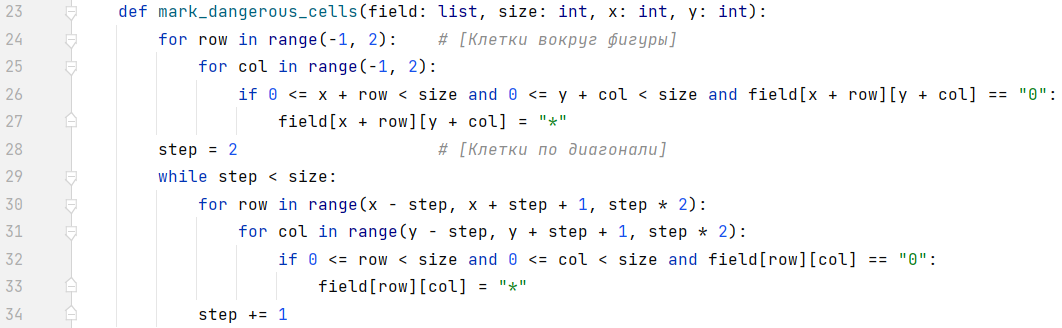


Для фигуры Конь-дракон (смесь короля со слоном) я выбрал следующий алгоритм. Клетки вокруг фигуры (то есть те, на которые может походить король) помечаются отдельно от клеток по диагонали (клетки слона).

Чтобы просканировать клетки вокруг, я использовал два цикла for, из которых перебирают значения на промежутке [-1;1]. Эти значения прибавляются к координатам x и y, что позволяет получить координаты вокруг исходной клетки.

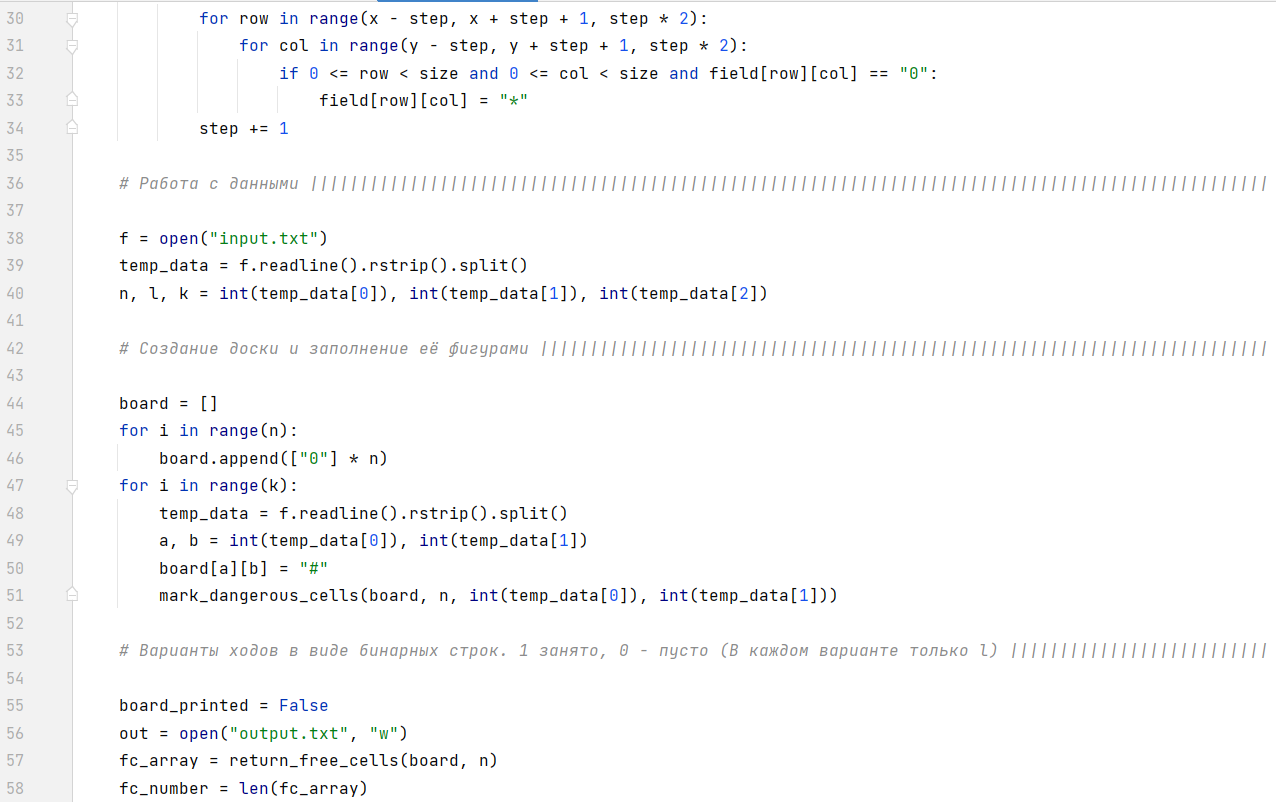


Для пометки клеток по диагонали алгоритм следующий: берётся переменная-шаг, не позволяющая координатам выйти далеко за пределы координатной системы доски. Изначально она равна 2, так как клетки на расстоянии 1 от фигуры были отмечены ранее. Чтобы не рассматривать лишние клетки, шаг в циклах равняется двойному значению переменной-шага step. При отдалении от фигуры каждый раз step увеличивается, и шаг в циклах увеличивается так, чтобы клетка с рассматриваемыми координатами всегда находилась по диагонали к фигуре.

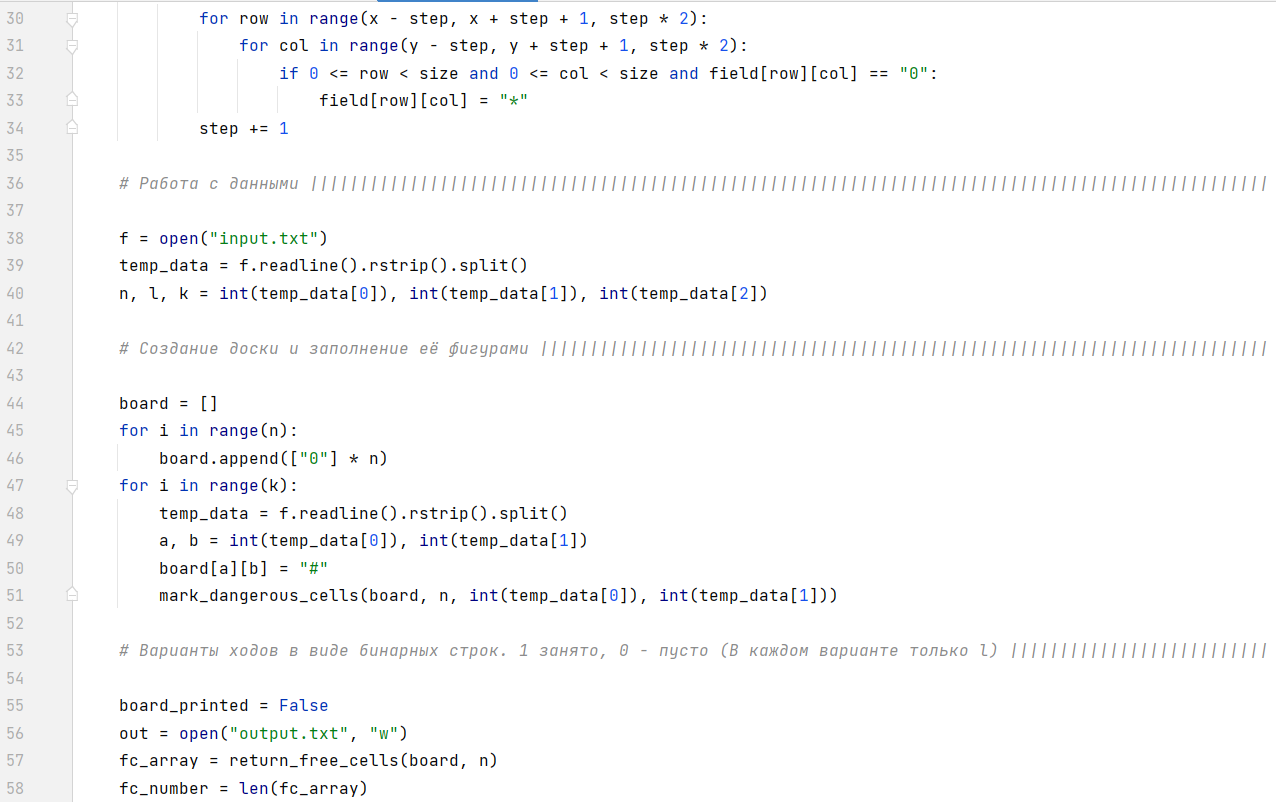


Так, на первой итерации между двумя клетками соседних диагоналей при том, что клетки находятся на одном расстоянии от фигуры, есть ещё три другие клетки. Чтобы дойти от первой до второй, всегда требуется вдвое больше шагов, чем step, в данном случае 4.

Далее идёт распределение трёх входных чисел из первой строки вводного файла по буквенным переменным так, как они идут в условии.



Следующим шагом, программа создаёт двухмерный массив размерами N на N, заполненный строками «0» – доску. Отдельным циклом на доску ставятся K фигур из входного файла, а клетки под боем помечаются с помощью вышеупомянутой функции mark\_dangerous\_cells.



Далее открывается выводной файл, создаётся список координат свободных клеток fc\_array и список бинарных строк (строк только с нулями и единицами). Длина каждой строки – fc\_number. В каждой такой строке должно быть только L единиц, так как каждая единица –это поставленная на доске фигура.

Так же создаётся булева переменная board\_printed, которая нужна в дальнейшем для вывода только одной доски (в условии сказано, что на консоль выводится один любой вариант расстановки, но здесь будет выводиться самый первый правильный вариант).



Следующим этапом идёт перебор вариантов, на основе бинарных строк. Происходит это следующим образом: итерируя бинарную строку, программа рано или поздно натыкается на единицу – это значит, что фигура с координатами, которые находятся по адресу fc\_array[digit] (список кортежей fc\_array имеет ту же длину, что и binary) должна быть поставлена.

Если клетка с вытащенными из fc\_array координатами находится под боем, решение считается неверным. В противном случае программа ставит фигуру и переходит к следующей (то есть продолжает итерировать binary, пока снова не увидит единицу или строка не закончится).

Если переменная-флажок is\_salvation не приняла значение False по ходу итерирования строки, то решение считается верным и записывается в выводной файл.

Дополнительно печатается первое правильное решение. Если это происходит, то вышеупомянутый флажок board\_printed меняет значение на True, не позволяя напечатать другую доску

****

**Результат работы**:

Программа работает исправно, однако по ходу написания выяснилось, что время её работы напрямую зависит от количества свободных клеток, оставшихся после поставленных изначально фигур (fc\_number) и количества фигур, которые ещё нужно поставить (L). Оптимизация программы лишь частично помогла уменьшить время, требуемое на выполнение.

**Вывод**:

В программе были использованы знания по всем аспектам программирования, обозначенным в целях, соответственно цели лабораторной работы выполнены.

**Листинг:**