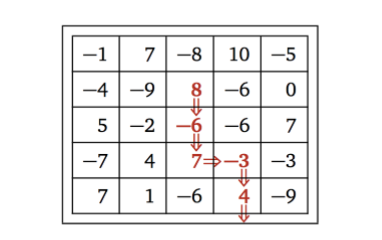
Vankin’s Mile is an American solitaire game played on an n×n square grid. The player starts by placing a token on any square of the grid. Then on each turn, the player moves the token either one square to the right or one square down. The game ends when player moves the token oﬀ the edge of the board. Each square of the grid has a numerical value, which could be positive, negative, or zero. The player starts with a score of zero; whenever the token lands on a square, the player adds its value to his score. The object of the game is to score as many points as possible.



For example, given the grid below, the player can score 8−6+7−3+4 = 10 points by placing the initial token on the 8 in the second row, and then moving down, down, right, down, down. (This is not the best possible score for these values.) In this assignment, you describe and analyze an eﬃcient algorithm to compute the maximum possible score for a game of Vankin’s Mile, given the n×n array of values as input.

麻烦用中文描述和解释一下算法和时间复杂度

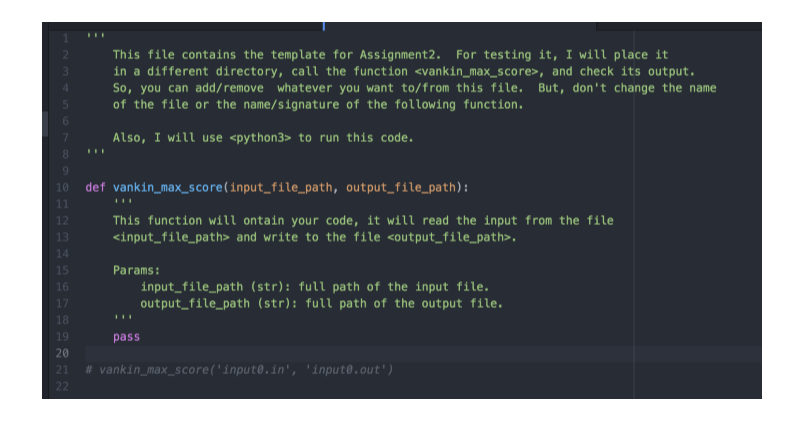
Code

Named assignment2.py, to compute the maximum possible score for a given game. The following template is provided.

You need to implement the function *vankin\_max\_score*.

This function has two parameters: *input\_ﬁle\_path* and *output\_ﬁle\_path*, specifying the full path of the input and output ﬁles, respectively.

When called, the function *vankin\_max\_score* should read the game from the input ﬁle, and writes maximum possible score of the game in the output ﬁle.



Tests

Your program will be tested against several test cases, for correctness and eﬃciency. For each test case, the program will be automatically stopped after 20 seconds if it is not done in that time. In this case, the group will miss the points of that test case. Note: it is important that your output is formatted as described below, since your codes will be tested automatically.

Input/Output

The input ﬁle is formatted as follows. The ﬁrst line is one integer 1 ≤ n ≤ 1000. The following n lines each is a row of the matrix. Each line is composed of n integers, each between −100 and 100, separated by commas.

The output ﬁle must contain a single integer: the maximum possible score on the n×n board of the input.

