

You shall submit a zipped, **and only zipped**, archive of your homework directory, hw3. The directory shall contain the files `find_sum.cpp` and `find_sum.h`. Name the archive submission file `hw3.zip`

Description

Consider word puzzles in which you search for words hidden in a scramble of letters. Consider an algorithmic solution to this. You are tasked with solving a different problem that essentially removes repetitive code:

Determine whether you can create one or more summations given a goal value from the addition of the sub-rows or sub-columns of a matrix of integers.

You will provide me with a library containing (at least) the function `FindSum` as described in the file `find_sum.h`. You should read the header file as well as my test file to understand what is expected from the library. Do note that the preconditions keep you from any unpleasantness due to array bounds and the like.

I have provided you a basic test application which you can use to ensure that your code is, at least partially, correct. I would suggest a more rigorous testing scheme to ensure that your methods handle any edge cases.

To call the program with the provided test files, reference the makefile rule for `test-find-sum`. The tests correspond to

- `./test-find-sum dat/row_matrix.txt 0` : row summations
- `./test-find-sum dat/col_matrix.txt 1` : column summations
- `./test-find-sum dat/matrix.txt 4` : no summations
- `./test-find-sum dat/matrix.txt 5` : single digit summations,

Points

You will receive points if your `FindSum`:

- (i) compiles (1 point)
- (ii) finds row summations (1.5 points)
- (iii) finds column summations (1.5 points)
- (iv) determines no summations found (0.5 points)
- (v) finds single-digit summations (0.5 points)