Programming Fundamentals

Submission

```
5206267 Luo, Zheng                              3707/3 AEROAH

Submissions:-

S 0     Wed Jun 24 20:25:41 2020        5206267 thu18b ass1_freefall -17:-21
S 1     Sun Jul  5 18:47:25 2020        5206267 thu18b ass1_freefall -6:-23

Sat Jul 18 09:57:14 2020                ## bongo12.orchestra.cse.unsw.EDU.AU ##
```

Listing

freefall.c

```
 1  // Zheng Luo (z5206267@cse.unsw.edu.au)
 2  // Date: 21/June/2020
 3  // UNSW Freefall assignment.
 4  // Update 24/June: Everything is finished except for most of the stage 4.
 5
 6  #include <stdio.h>
 7  #include <stdlib.h>
 8  #include <math.h>
 9
10  // The size of the array.
11  #define SIZE 15
12  // The meaning of each command number.
13  #define EMPTY 0
14  #define STONE 1
15  #define MARCHING_BLOCK 2
16  // Define the number of stone will be destroyed by the laser.
17  #define LASERPOWER 3
18  // For marching blocks, define 4 is left, 6 is right, 2 is down.
19  // MOVED is used to prevent repeated movements
20  #define MOVED 1
21  #define DOWN 2
22  #define LEFT 4
23  #define RIGHT 6
24  // Define the valid input range for placing stone in stage 1.1.
25  #define EXPRESSION_FOR_VALID_ROW ((row[counterForSets] >= 0)\
26  && (row[counterForSets] < 15))
                                ^
                                + ========================================================== +
                                + CONSTANTS: you should use the provided #defines e.g. SIZE +
                                + ========================================================== +
27  #define EXPRESSION_FOR_VALID_COLUMN ((column[counterForSets] >= 0)\
28  && (column[counterForSets] < 15))
29  #define EXPRESSION_FOR_VALID_LENGTH ((length[counterForSets] >= 0)\
30  && (length[counterForSets] <= 15))
                                          ^
+ ========================================================== +
+ Be very very careful when you do this, it is not good C     +
+ style and is not recommended. While doing this may          +
+ make your code more readable, it also hides functional code +
+ - making it difficult to debug. It also hides a "magic      +
+ number" 15, if someone was to change the #define SIZE to 20 +
+ instead, this expression will still check 15 rather than    +
+ 20. While it may seem superfluous, you could instead create +
+ a function for these defines. E.g. int valid_row().         +
+ ========================================================== +
31
32  // The functions below follow same order as appeared in main function.
33  // Scan inputs and placing stones.
34  void placingStone(int map[SIZE][SIZE], int playerX);
35
36  // Print out the map.
37  void printMap(int map[SIZE][SIZE], int playerX);
38
39  // Move the player for stage 1.2.
40  int playerMovement(int command, int direction, int playerX);
41
42  // Detect the objects(stone/TNT) above the player, clear them as required.
43  void clearObject(int map[SIZE][SIZE], int playerX);
44
45  // Check the end game condition for stage 2.2.
46  int wonConditionCheck(int map[SIZE][SIZE]);
47  int lostConditionCheck(int map[SIZE][SIZE]);
48
49  // Limit the range of counterForMarchingBlocks to stay between 1-4.
50  int rangeForCounterMarchingBlocks(int counterForMarchingBlocks);
51
52  // Shifting for both non-marching and marching blocks.
53  void shiftBlocks(int map[SIZE][SIZE], int playerX, int numberForDirection,
54  int gameLost);
55
56  // Vertically flip the map.
```

```
57  void verticallyFlip(int map[SIZE][SIZE], int playerX);
58
59
                ^
                + ====================== +
                + Good use of functions! +
                + ====================== +
60
61  int main (void) {
62      // This line creates our 2D array called "map" and sets all as EMPTY.
63      int map[SIZE][SIZE] = {EMPTY};
64      // This line creates out playerX variable.
65      int playerX = SIZE / 2;
66      // Scan input and placing stones.
67      placingStone(map, playerX);
68      // Print out the map.
69      printMap(map, playerX);
70
71      int gameLost = 0; // For stage 2.1, 0 means game is still running.
                          ^
                          + ========================================================== +
                          + INLINE COMMENTS: it's usually better for comments to go on  +
                          + the                                                         +
                          + line above                                                  +
                          + You could #define RUNNING 0 rather than make a comment    +
                          + about it. Code is easier to maintain than comments.        +
                          + ========================================================== +
72      int command4Used = 0; // For stage 3.1, 0 means command 4 havent been used.
73      // Initiate a counter for counting the number of downshift it has made,
74      // in order to decide next movement for stage 4.1.
75      int counterForMarchingBlocks = 0;
76      int command, directionForPlayer; // Initiate command and direction variables.
77
78      // This programme continue to operate until the command is end of file (EOF).
79      while (scanf("%d", &command) != EOF) {
80          // Process the different types of command below.
81          if (command == 1) { // Stage 1.2: Moving the player.
82              scanf("%d", &directionForPlayer);
83              // It will continue from last position by equating these.
84              playerX = playerMovement(command, directionForPlayer, playerX);
85          }
86          else if (command == 2) { // Stage 1.3: Firing the laser.
                                     ^
                                     + ================================================ +
                                     + To make code more readable #define FIRE_LASER 2 +
                                     + ================================================ +
87              clearObject(map, playerX);
88          }
89          else if (command == 3) { // Stage 2.1: Shift everything down.
                                     ^
                                     + ============================================================== +
                                     + CONSTANTS: you should create use #defines e.g. SHIFT_DOWN      +
                                     +                       (+ 2 other constants errors)             +
                                     + ============================================================== +
90              // Check lost game condition first, if not lose yet, proceed.
91              while (lostConditionCheck(map) == 1) {
92                  gameLost = 1;
93                  break;
94              }
95              // The numbers of downshift command +1 if within range,
96              // and always counting within the range of 1 to 4.
97              counterForMarchingBlocks =
98              rangeForCounterMarchingBlocks(counterForMarchingBlocks);
99
100             shiftBlocks(map, playerX, counterForMarchingBlocks, gameLost);
101         }
102         // Stage 3.1, vertically flip.
103         else if (command == 4 && command4Used == 0) {
                                     ^
                                     + ========================================================== +
                                     + CONSTANTS: you should create use #defines e.g. VERTICAL_FLIP +
                                     + ========================================================== +
```

```
            // command4 has been used now, this function cannot be used again.
            command4Used = 1;

            verticallyFlip(map, playerX);
        }
        // Print out the map for the lastest command.
        printMap(map, playerX);
        // Print out end game statement.
        if (wonConditionCheck(map) == 1) {
            printf("Game Won!\n");
            break;
        } else if (gameLost == 1) {
            printf("Game Lost!\n");
            break;
        }
    }
    return 0;
}



// Functions below:


void placingStone(int map[SIZE][SIZE], int playerX) {
    // Scan in the number of lines of blocks.
    int linesOfStone;
    printf("How many lines of stone? ");
    scanf("%d", &linesOfStone);

    // Scan the locations of the lines as a group of four integers.
    // Create 4 different arrays for row, column, length, value correspondently
    // with the same size as input above.
    int row[linesOfStone], column[linesOfStone],
    length[linesOfStone], value[linesOfStone];

    printf("Enter lines of stone:\n");
    // To consist with the number of lines,
    // corresponded amount of sets of four integers have to be scanned.
    int counterForScanLocation = 0;
    while (counterForScanLocation < linesOfStone) {
        scanf("%d %d %d %d", &row[counterForScanLocation],
        &column[counterForScanLocation],
        &length[counterForScanLocation], &value[counterForScanLocation]);

        counterForScanLocation++;
    }

    // Stage 1.1: Placing stone.

    int counterForSets = 0;
    // Moving the inputs from array to map by
    // dealing with each set of 4 integer command once at a time.
    while (counterForSets < linesOfStone) {
        int counterForLength = 0;
        // Dealing with each square in each set once at a time
        while (counterForLength < length[counterForSets]) {
            if (EXPRESSION_FOR_VALID_ROW && EXPRESSION_FOR_VALID_COLUMN &&
                EXPRESSION_FOR_VALID_LENGTH) {
                map[row[counterForSets]][column[counterForSets] +
                counterForLength] = value[counterForSets];
            }

            counterForLength++;
        }
        counterForSets++;
    }
}



```

```
176 // Print out the contents of the map array. Then print out the player line
177 // which will depends on the playerX variable.
178 void printMap(int map[SIZE][SIZE], int playerX) {
179
180     // Print values from the map array.
181     int i = 0;
182     while (i < SIZE) {
183         int j = 0;
184         while (j < SIZE) {
185             printf("%d ", map[i][j]);
186             j++;
187         }
188         printf("\n");
189         i++;
190     }
191     // Print the player line.
192     i = 0;
193     while (i < playerX) {
194         printf("  ");
195         i++;
196     }
197     printf("P\n");
198 }
199
200
201
202 int playerMovement(int command, int directionForPlayer, int playerX) {
              ^
              + ============================================================ +
              + Comment above functions implementations with your approach  +
              + and what it is meant to do!                                 +
              + ============================================================ +
203     // command == 1: Valid command to initiate the movement.
204     // playerX < (SIZE - 1): Make sure the player will stay within the range.
205     if (command == 1 && playerX < (SIZE - 1)) {
206         if (directionForPlayer == -1) { // Left
                                          ^
                                          +
============================================================== +
                                          + INLINE COMMENTS: it's usually better for comments to go on the
+
                                          + line above                              (+ 4 other inline
+
                                          + comments errors)
+
                                          +
============================================================== +
207             playerX--;
208         } else if (directionForPlayer == 1) { // Right
209             playerX++;
210         }
211     }
212     return playerX;
213
214 }
215
216
217 int wonConditionCheck(int map[SIZE][SIZE]) {
218     int gameWon = 1;
219     int i = 0;
220     while (i < SIZE) {
221         int j = 0;
222         while (j < SIZE) {
223             // If map is not empty for all squares, then no gameWon.
224             if (map[i][j] != EMPTY) {
225                 gameWon = 0;
226             }
227             j++;
228         }
229         i++;
230     }
231     return gameWon;
```

```
232
233 }
234
235 int lostConditionCheck(int map[SIZE][SIZE]) {
236     int gameLost = 0;
237     int row = SIZE - 1; // Bottom line
238     int column = 0;
239
240     while (column < SIZE) {
241         if (map[row][column] == STONE) {
242             gameLost = 1;
243         }
244         column++;
245     }
246
247     return gameLost;
248 }
249
250
251
252 int rangeForCounterMarchingBlocks(int counterForMarchingBlocks) {
253     if (counterForMarchingBlocks < 4) {
254         counterForMarchingBlocks++;
255     } else {
256         counterForMarchingBlocks = 1;
257     }
258     return counterForMarchingBlocks;
259 }
260
261
262
263 void shiftBlocks(int map[SIZE][SIZE], int playerX, int numberForDirection,
264 int gameLost) {
265
266     // Create an array to make sure that marching block will only move once.
267     // If compiler detects one, which means moved already,
268     //  it will not move again by single shifting commmand.
269     int arrayForRepeatMoving[SIZE + 1][SIZE + 1] = {0};
270
271     // Scan and filter everything in reverse manner,
272     // i.e. from bottom right to top right then (column - 1) and repeat.
273     int counterForColumn = SIZE - 1;
274     while (counterForColumn >= 0 && gameLost == 0) {
275         int counterForRow = SIZE - 1;
276
277         while (counterForRow > 0) {
278
279             // Specfic the direction.
280             // Calculate the next moving direction for marching block
281             // by counterForMarchingBlocks.
282             int direction = numberForDirection;
283             if (numberForDirection == 1 || numberForDirection == 3) {
                                                                ^
                                + ========================================================= +
                                + CONSTANTS: you should create use #defines e.g. SHIFT_DOWN +
                                + ========================================================= +
284                 direction = 2; // 2 is down
285             } else if (numberForDirection == 2) {
286                 direction = 6; // 6 is right
287             } else if (numberForDirection == 4) {
288                 direction = 4; // 4 is left
289             }
290
291             // If marching block(2) has been detected from one square below.
292             if (map[counterForRow - 1][counterForColumn] == MARCHING_BLOCK) {
293
294                 if (direction == DOWN)
295                 {
296                     // Move 2 to the desire location.
297                     map[counterForRow][counterForColumn] =
298                     map[counterForRow - 1][counterForColumn];
299                     // Make original position empty.
```

```
300                     map[counterForRow - 1][counterForColumn] = EMPTY;
301                     // Assume next new line come from above is empty.
302                     if (counterForRow == 1) {
303                         map[0][counterForColumn] = EMPTY;
                            ^
                            + ============================================================== +
                            + OVERDEEP_NESTING: nesting too deep: 6 (try moving some of the     +
                            + logic into another function)                                      +
                            + ============================================================== +
304                     }
305
306                 }
307                 else if (direction == RIGHT)
308                 {
309                     // Do not move if outside the range, and mark.
310                     if (counterForColumn + 1 >= SIZE) {
311                         arrayForRepeatMoving[counterForRow - 1]
312                         [counterForColumn] = MOVED;
313                     } else {  // Move 2 to the desire location.
314                         map[counterForRow - 1][counterForColumn + 1] =
315                         map[counterForRow - 1][counterForColumn];
316                         // Make original position empty.
317                         map[counterForRow - 1][counterForColumn] = EMPTY;
318                     }
319
320                 }
321                 else if (direction == LEFT &&
322                 arrayForRepeatMoving[counterForRow - 1][counterForColumn + 1] !=
323                 MOVED && counterForColumn - 1 >= 0)
324                 {
325                     // Move 2 to the desire location.
326                     map[counterForRow - 1][counterForColumn - 1] =
                        ^
                        + ============================================================== +
                        + OVERDEEP_NESTING: nesting too deep: 5 (try moving some of the     +
                        + logic into another function)  (+ 2 other overdeep_nesting errors) +
                        + ============================================================== +
327                     map[counterForRow - 1][counterForColumn];
328                     // Make original position empty.
329                     map[counterForRow - 1][counterForColumn] = EMPTY;
330                     // Mark the old position has been moved.
331                     arrayForRepeatMoving[counterForRow - 1]
332                     [counterForColumn] = MOVED;
333                 }
334             }
335
336             // If non-marching block has been detected.
337             else if (map[counterForRow - 1][counterForColumn] !=
338             MARCHING_BLOCK &&
339                 arrayForRepeatMoving[counterForRow][counterForColumn + 1] != MOVED
340                 /* To prevent moved square to be erased*/)
341                 {
342                     // Move 2 to the desire location.
343                     map[counterForRow][counterForColumn] =
344                     map[counterForRow - 1][counterForColumn];
345                     // Make original position empty.
346                     map[counterForRow - 1][counterForColumn] = EMPTY;
347                     // Assume next line come from above is empty.
348                     if (counterForRow == 1) {
349                         map[0][counterForColumn] = EMPTY;
350                     }
351                 }
352
353                 counterForRow--;
354             }
355             counterForColumn--;
356         }
357     }
358
359
360
361 void clearObject(int map[SIZE][SIZE], int playerX) {
```

```
362     // Detect the existence of stone above the player.
363     int counterForLaser = 0;
364     while (counterForLaser < SIZE) {
365
366         // If non-empty is detected, proceed.
367         // Otherwise it will keep scanning until non-empty is detected.
368         if (map[SIZE - 1 - counterForLaser][playerX] != EMPTY) {
369
370             // Clear each stone/TNT once at a time
371             int counterForClearing = 1;
372             while (counterForClearing < LASERPOWER + 1 &&
373             (SIZE - counterForClearing - counterForLaser) >= 0 &&
374             (SIZE - counterForClearing - counterForLaser) < SIZE) {
375
376                 // If stone is detected, clear itself in this while loop.
377                 if (map[SIZE - counterForClearing -
378                 counterForLaser][playerX] == STONE) {
379                     map[SIZE - counterForClearing -
                        ^
                        + ============================================================== +
                        + OVERDEEP_NESTING: nesting too deep: 5 (try moving some of the     +
                        + logic into another function)  (+ 1 other overdeep_nesting errors) +
                        + ============================================================== +
380                     counterForLaser][playerX] = EMPTY;
381                 }
382                 // If TNT is detected (value 3 - 9), clear circle of stones.
383                 else if (map[SIZE - counterForClearing -
384                 counterForLaser][playerX] != STONE &&
385                 map[SIZE - counterForClearing -
386                 counterForLaser][playerX] != EMPTY)
387                 {
388
389                     // Determine the radius of TNT base on its value.
390                     int radiusForTNT = map[SIZE - counterForClearing -
391                     counterForLaser][playerX];
392                     // EMPTY if (within the range && STONE in the square only).
393                     int counterForRow = 0;
394                     while (counterForRow < SIZE) {
395                         int counterForColumn = 0;
396                         while (counterForColumn < SIZE) {
397
398                             // Calculate the range for TNT.
399                             int rangeForTNT = sqrt((counterForRow -
400                             (SIZE - counterForClearing - counterForLaser)) *
401                             (counterForRow -
402                             (SIZE - counterForClearing - counterForLaser)) +
403                             (counterForColumn - playerX) *
404                             (counterForColumn - playerX));
405                             // If distance between TNT and chosen point is
406                             // smaller than the power range of TNT,
407                             // all stones within the area will be cleared.
408
409                             if ((rangeForTNT< radiusForTNT) &&
                                    ^
                                    + ============================================================== +
                                    + WHITESPACE: You had 'rangeForTNT< radiusForTNT', where you        +
                                    + should have had space around operators (e.g. i + 3 not i+3)       +
                                    + ============================================================== +
410                             (map[counterForRow][counterForColumn] == STONE
411                             || map[counterForRow][counterForColumn] ==
412                             radiusForTNT))
413                             {
414                                 map[counterForRow][counterForColumn] = EMPTY;
                                    ^
                                    + ============================================================== +
                                    + OVERDEEP_NESTING: nesting too deep: 8 (try moving some of the     +
                                    + logic into another function)                                      +
                                    + ============================================================== +
415                             }
416                             counterForColumn++;
417                         }
418                         counterForRow++;
```

```
                    }
                }

                counterForClearing++;
            }
            // The program will break when it clear 3 stones,
            // Otherwise it will keep firing laser utill
            // whole column is empty.
            break;
        }
        counterForLaser++;
    }

}



void verticallyFlip(int map[SIZE][SIZE], int playerX) {
    int temporaryArray[SIZE][SIZE]; // For swapping numbers

    // Copy the old array into a temporary array
    int counterForColumn = 0;
    while (counterForColumn < SIZE) {
        int counterForRow = 0;
        while (counterForRow < SIZE) {
            temporaryArray[counterForRow][counterForColumn] =
            map[counterForRow][counterForColumn];
            counterForRow++;
        }
        counterForColumn++;
    }
    // Flip the array by moving data from temporary array to map[][].
    counterForColumn = 0;
    while (counterForColumn < SIZE) {
        int counterForRow = 0;
        while (counterForRow < SIZE) {
            map[SIZE - 1 - counterForRow][counterForColumn] =
            temporaryArray[counterForRow][counterForColumn];
            counterForRow++;
        }
        counterForColumn++;
    }
}
```

Style Summary

```
^
+
==========================================================================================================================
 +
+
+
+ ========================================
+
+ =====    Style feedback summary:   =====
+
+ ========================================
+
+
+
+ ====== Header Comment =====
+
+ Header comment has 4 lines (2 lines of description)
+
+ Header comment contains zID!
+
+
+
+ ====== #defines =====
+
+ 9 additional constants #defined:
+
+     15  #define MARCHING_BLOCK 2
+
+     17  #define LASERPOWER 3
+
+     20  #define MOVED 1
+
+     21  #define DOWN 2
+
+     22  #define LEFT 4
+
+     23  #define RIGHT 6
+
+     25  #define EXPRESSION_FOR_VALID_ROW ((row[counterForSets] >= 0)\
+
+     27  #define EXPRESSION_FOR_VALID_COLUMN ((column[counterForSets] >= 0)\
+
+     29  #define EXPRESSION_FOR_VALID_LENGTH ((length[counterForSets] >= 0)\
+
+
+
+ ====== Nesting Depth =======
+
+ Nesting depth was too much: max depth of 8!
+
+
+
+ ====== Whitespace Errors =====
+
+ You had 1 whitespace errors:
+
+ On line 409, you had 'rangeForTNT< radiusForTNT', where you should have had space around operators (e.g. i + 3
not i+3) +
+    409                              if ((rangeForTNT< radiusForTNT) &&
+
+
+
+ ====== Indentation =======
+
+ No indentation issues!
+
+
+
+ ====== Over-long Lines =======
+
+ No lines over 80 characters!
```

```
+
+
+
+ ====== Complex If Statements =======
+
+ No complex if statements!
+
+
+
+ ====== Functions and Prototypes =====
+
+
+
+ Function implementations:
+
+
+
+ freefall.c functions:
+
+     61  int main (void) {
+
+          -> 60 lines long (56 code lines)
+
+          -> (has a 1 line function comment)
+
+    129  void placingStone(int map[SIZE][SIZE], int playerX) {
+
+          -> 43 lines long (37 code lines)
+
+          -> (has a 1 line function comment)
+
+    178  void printMap(int map[SIZE][SIZE], int playerX) {
+
+          -> 20 lines long (19 code lines)
+
+          -> (has no function comment!)
+
+    202  int playerMovement(int command, int directionForPlayer, int playerX) {
+
+          -> 12 lines long (11 code lines)
+
+          -> (has a 2 line function comment)
+
+    217  int wonConditionCheck(int map[SIZE][SIZE]) {
+
+          -> 16 lines long (15 code lines)
+
+          -> (has no function comment!)
+
+    235  int lostConditionCheck(int map[SIZE][SIZE]) {
+
+          -> 13 lines long (11 code lines)
+
+          -> (has no function comment!)
+
+    252  int rangeForCounterMarchingBlocks(int counterForMarchingBlocks) {
+
+          -> 7 lines long (7 code lines)
+
+          -> (has no function comment!)
+
+    263  void shiftBlocks(int map[SIZE][SIZE], int playerX, int numberForDirection,
+
+          -> 94 lines long (84 code lines)
+
+          -> (has no function comment!)
+
+    361  void clearObject(int map[SIZE][SIZE], int playerX) {
+
+          -> 71 lines long (63 code lines)
+
+          -> (has a 1 line function comment)
```

```
+
+    436  void verticallyFlip(int map[SIZE][SIZE], int playerX) {
+
+          -> 25 lines long (24 code lines)
+
+          -> (has a 2 line function comment)
+
+
+
+ ====== Variables =======
+
+ Declared 30 additional variables:
+
+     63  int map[SIZE][SIZE] = {EMPTY};
+
+     65  int playerX = SIZE / 2;
+
+     71  int gameLost = 0;
+
+     72  int command4Used = 0;
+
+     75  int counterForMarchingBlocks = 0;
+
+     76  int command, directionForPlayer;
+
+    131  int linesOfStone;
+
+    144  int counterForScanLocation = 0;
+
+    155  int counterForSets = 0;
+
+    159  int counterForLength = 0;
+
+    181  int i = 0;
+
+    183  int j = 0;
+
+    218  int gameWon = 1;
+
+    219  int i = 0;
+
+    221  int j = 0;
+
+    236  int gameLost = 0;
+
+    237  int row = SIZE - 1;
+
+    238  int column = 0;
+
+    273  int counterForColumn = SIZE - 1;
+
+    275  int counterForRow = SIZE - 1;
+
+    282  int direction = numberForDirection;
+
+    363  int counterForLaser = 0;
+
+    371  int counterForClearing = 1;
+
+    393  int counterForRow = 0;
+
+    395  int counterForColumn = 0;
+
+    437  int temporaryArray[SIZE][SIZE];
+
+    440  int counterForColumn = 0;
+
+    442  int counterForRow = 0;
+
+    453  int counterForRow = 0;
+
+
```

```
+
+
================================================================================================================
 +
^
+ ============================================================================================================ +
+                                                                                                              +
+ Great:                                                                                                       +
+ ------                                                                                                       +
+   Header comment                                                                                             +
+   Indentation                                                                                                +
+   Whitespace                                                                                                 +
+   Variable names                                                                                             +
+   Functions                                                                                                  +
+   Line length                                                                                                +
+                                                                                                              +
+                                                                                                              +
+ A few issues:                                                                                                +
+ -------------                                                                                                +
+   Constants (#defines) (you should use the provided #defines, and should create and use your own #defines)   +
+   Comments (you should write a comment above each function explaining what it does)                          +
+                                                                                                              +
+                                                                                                              +
+ Poor:                                                                                                        +
+ -----                                                                                                        +
+   Nesting depth (see summary above)                                                                          +
+                                                                                                              +
+ ============================================================================================================ +
```

Assessment

```
Test 01_fire_double_middle (./freefall) - passed
Test 01_fire_double_top (./freefall) - passed
Test 01_fire_move_left (./freefall) - passed
Test 01_fire_move_right (./freefall) - passed
Test 01_fire_quadruple_middle (./freefall) - passed
Test 01_fire_quadruple_top (./freefall) - passed
Test 01_fire_single_middle (./freefall) - passed
Test 01_fire_single_top (./freefall) - passed
Test 01_fire_triple_middle (./freefall) - passed
Test 01_fire_triple_top (./freefall) - passed
Test 01_move_invalid_negative (./freefall) - passed
Test 01_move_invalid_positive (./freefall) - passed
Test 01_move_left_and_right (./freefall) - passed
Test 01_move_left_lots (./freefall) - passed
Test 01_move_left_lots_back (./freefall) - passed
Test 01_move_left_multiple (./freefall) - passed
Test 01_move_left_once (./freefall) - passed
Test 01_move_right_lots (./freefall) - passed
Test 01_move_right_lots_back (./freefall) - passed
Test 01_move_right_multiple (./freefall) - passed
Test 01_move_right_once (./freefall) - passed
Test 01_place_bottom_edge (./freefall) - passed
Test 01_place_bottom_invalid (./freefall) - passed
Test 01_place_bottom_invalid_edge (./freefall) - passed
Test 01_place_left_edge (./freefall) - passed
Test 01_place_left_invalid (./freefall) - passed
Test 01_place_left_invalid_edge (./freefall) - passed
Test 01_place_left_partial_invalid (./freefall) - passed
Test 01_place_lots (./freefall) - passed
Test 01_place_multiple_invalid (./freefall) - failed (errors)
Your program produced these errors:

freefall.c:164:17: runtime error - index 15 out of bounds for type 'int [15]'

dcc explanation: You are using an illegal array index: 15
Valid indices for an array of size 15 are 0..14
Make sure the size of your array is correct.
Make sure your array indices are correct.

Your program produced no output
Test 01_place_multiple_invalid_desc (./freefall) - failed (errors - same as Test 01_place_multiple_invalid)
Test 01_place_multiple_rows (./freefall) - passed
Test 01_place_one_row (./freefall) - passed
Test 01_place_overlapping (./freefall) - passed
Test 01_place_right_edge (./freefall) - passed
Test 01_place_right_invalid (./freefall) - passed
Test 01_place_right_invalid_edge (./freefall) - passed
Test 01_place_right_partial_invalid (./freefall) - passed
Test 01_place_top_edge (./freefall) - passed
Test 01_place_top_invalid (./freefall) - passed
Test 01_place_top_invalid_edge (./freefall) - passed
Test 02_lose_left_edge (./freefall) - passed
Test 02_lose_right_edge (./freefall) - passed
Test 02_lose_single (./freefall) - passed
Test 02_shift_down_centre (./freefall) - passed
Test 02_shift_down_left (./freefall) - passed
Test 02_shift_down_lots (./freefall) - passed
Test 02_shift_down_multiple (./freefall) - passed
Test 02_shift_down_right (./freefall) - passed
Test 02_shift_down_square (./freefall) - passed
Test 02_shift_down_square_multiple (./freefall) - passed
Test 02_shift_down_to_bottom (./freefall) - passed
Test 02_shift_down_top (./freefall) - passed
Test 02_win_multiple (./freefall) - passed
Test 02_win_single (./freefall) - passed
Test 03_tnt_huge_tnt (./freefall) - passed
Test 03_tnt_radius_6_center (./freefall) - passed
Test 03_tnt_radius_7_center (./freefall) - passed
Test 03_tnt_radius_8_center (./freefall) - passed
Test 03_tnt_radius_9_center (./freefall) - passed
Test 03_vertical_flip_bottom (./freefall) - passed
```

```
Test 03_vertical_flip_diagonal (./freefall) - passed
Test 03_vertical_flip_left (./freefall) - passed
Test 03_vertical_flip_multiple (./freefall) - passed
Test 03_vertical_flip_one_line (./freefall) - passed
Test 03_vertical_flip_right (./freefall) - passed
Test 03_vertical_flip_square (./freefall) - passed
Test 03_vertical_flip_top (./freefall) - passed
Test 04_above_invalid_left (./freefall) - failed (Incorrect output)
Your program produced these 129 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_above_laser_dont_destroy_above (./freefall) - failed (Incorrect output)
Your program produced these 129 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_above_line (./freefall) - failed (Incorrect output)
Your program produced these 97 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_above_no_marching (./freefall) - failed (Incorrect output)
Your program produced these 193 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
+ 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 2 2 2 2 2 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 2 2 2 2 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_above_top_limit (./freefall) - failed (Incorrect output)
Your program produced these 258 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
+ 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
- Game Won!
+ 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+             P
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Test 04_above_win_blocks_still_above (./freefall) - failed (Incorrect output)
Your program produced these 50 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
...
Test 04_gravity_invalid (./freefall) - failed (Incorrect output)
Your program produced these 81 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
-               P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 1 1 0 1 1 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Test 04_gravity_left (./freefall) - failed (errors)
Your program produced these errors:

Runtime error: uninitialized variable accessed.

Execution stopped in playerMovement(command=1, directionForPlayer=<uninitialized value>, playerX=7) in freefall.c
at line 206:
```

```
int playerMovement(int command, int directionForPlayer, int playerX) {
// command == 1: Valid command to initiate the movement.
// playerX < (SIZE - 1): Make sure the player will stay within the range.
if (command == 1 && playerX < (SIZE - 1)) {
-->     if (directionForPlayer == -1) { // Left
playerX--;
} else if (directionForPlayer == 1) { // Right
playerX++;

Values when execution stopped:

command = 1
directionForPlayer = <uninitialized value>
playerX = 7

Function Call Traceback
playerMovement(command=1, directionForPlayer=<uninitialized value>, playerX=7) called at line 84 of freefall.c
main()
Your program produced these 33 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
+ 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
+ 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
+ 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_gravity_marching_blocks (./freefall) - failed (Incorrect output)
Your program produced these 49 lines of output:
How many lines of stone? Enter lines of stone:
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
+ 0 0 0 0 2 2 2 2 2 0 0 2 2 2 2
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
-               P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 2 2 2 2 2 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_gravity_multiple (./freefall) - failed (Incorrect output)
Your program produced these 49 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 3 3 3 3 3 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 3 3 3 3 3 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 3 3 3 3 3 1 1 1 0 0 0 0 0
+ 0 0 0 0 0 0 0 3 3 3 3 3 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
-               P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 3 3 3 3 3 1 1 1 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_gravity_no_gravity_above_map (./freefall) - failed (Incorrect output)
Your program produced these 130 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0
+ 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
+ 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+               P
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
+ 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Test 04_gravity_right (./freefall) - failed (Incorrect output)
Your program produced these 49 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...

The difference between your output(-) and the correct output(+) is:
...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
-               P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_gravity_tnt_blocks (./freefall) - failed (Incorrect output)
Your program produced these 49 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 4 4 4 4 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 8 8 8 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 9 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 4 4 4 4 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 8 8 8 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 9 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 4 4 4 4 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 4 4 4 4
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 8 8 8 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 8 8 8
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 9 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 9
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
-               P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 9 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 8 8 8 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 4 4 4 4 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...
Test 04_gravity_top_left (./freefall) - failed (Incorrect output)
Your program produced these 49 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
P
+ 0 0 0 0 0 0 1 1 1 0 1 1 1 1 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
- 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
- 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
-               P
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Test 04_gravity_top_right (./freefall) - failed (Incorrect output - same as Test 04_gravity_top_left)
Test 04_marching_bottom_game_over (./freefall) - failed (Incorrect output)
Your program produced these 50 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 2 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
- Game Won!
+ Game Lost!
Test 04_marching_collide_marching (./freefall) - passed
Test 04_marching_collide_stone (./freefall) - passed
Test 04_marching_fire_laser (./freefall) - passed
Test 04_marching_square (./freefall) - failed (Incorrect output)
Your program produced these 193 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0 0
```

```
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
- 0 2 0 2 0 2 0 0 0 0 0 0 0 0 0
+ 0 2 2 2 2 2 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 2 0 2 0 2 0 0 0 0 0 0 0 0
+ 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0
- 0 0 2 0 2 0 2 0 0 0 0 0 0 0 0
+ 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0
- 0 0 2 0 2 0 2 0 0 0 0 0 0 0 0
+ 0 0 2 2 2 2 2 0 0 0 0 0 0 0 0
- 0 0 2 0 2 0 2 0 0 0 0 0 0 0 0
Test 04_marching_top_left (./freefall) - passed
Test 04_marching_top_right (./freefall) - failed (Incorrect output)
Your program produced these 50 lines of output:
How many lines of stone? Enter lines of stone:
0 0 0 0 0 0 0 0 0 0 0 0 0 0 2
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
P
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 2
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
...


The difference between your output(-) and the correct output(+) is:
...
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2
P
- Game Won!
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+               P
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
+ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Test 04_marching_vertical_flip (./freefall) - passed
71 tests passed 20 tests failed


Marking Summary:

|------------------------------------------|--------------|--------------|
| Test Name                                | Tests Passed | % Gained     |
|------------------------------------------|--------------|--------------|
| Place Rows of Blocks                     | *18/20       | 43.2/45      |
| Move Player                              |  11/11       | 10.0/10      |
| Fire Laser                               |  10/10       | 10.0/10      |
| Shift Everything Down                    |  9/9         | 4.0/4        |
| Win Condition                            |  2/2         | 3.0/3        |
| Lose Condition                           |  3/3         | 3.0/3        |
| TNT Blocks                               |  5/5         | 5.0/5        |
| Vertical Flip                            |  8/8         | 5.0/5        |
| Marching Blocks                          | *5/8         | 4.1/5        |
| Blocks Above the Map                     |  0/6         | 0.0/5        |
| Gravity Storms                           |  0/9         | 0.0/5        |

(* indicates test passed some, but not all tests)

NOTE: The following mark may be slightly different to the sum of
the "Mark Allocation" column above. This is due to rounding, not
a mistake. We will not change marks because of rounding.

Sum of percentage points gained: 87.3/100

This mark gets scaled to be out of 80:

Mark for automarking tests: 69.9/80


You can rerun the tests used in marking by running: 1511 automark freefall

!!specialmark  (automated testing)                      69.9/80

!!marktab       **  MARKER'S  ASSESSMENT  **

                                style  (20)     17???

!!finalmark     **  FINAL  ASSIGNMENT  MARK:    86.9/100

5206267 Luo, Zheng                              3707/3 AEROAH

^
+ ====================================================== +
+ Good job Zheng on your first assignment! I liked how you  +
+ broke down your code into functions. Work on removing     +
+ "magic numbers" that are unclear in your code - hence     +
+ removing unessecary comments that explain those numbers,  +
+ and writing comments above your functions.                +
```

```
+ ================================================== +


Marked by z5205677 on Sat Jul 25 02:07:17 2020
```