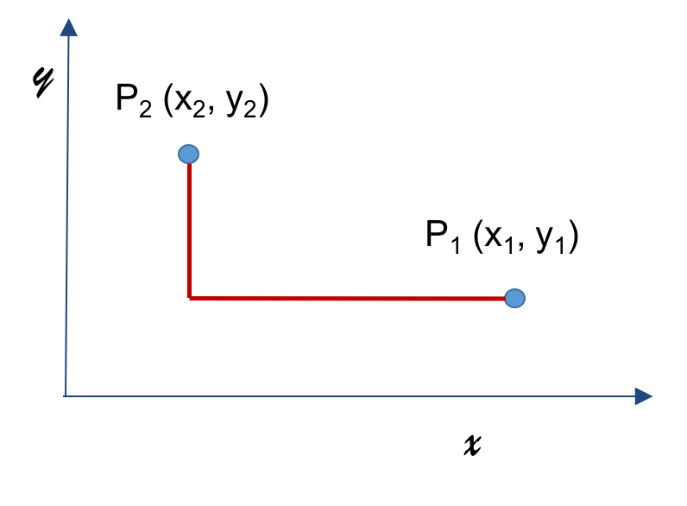
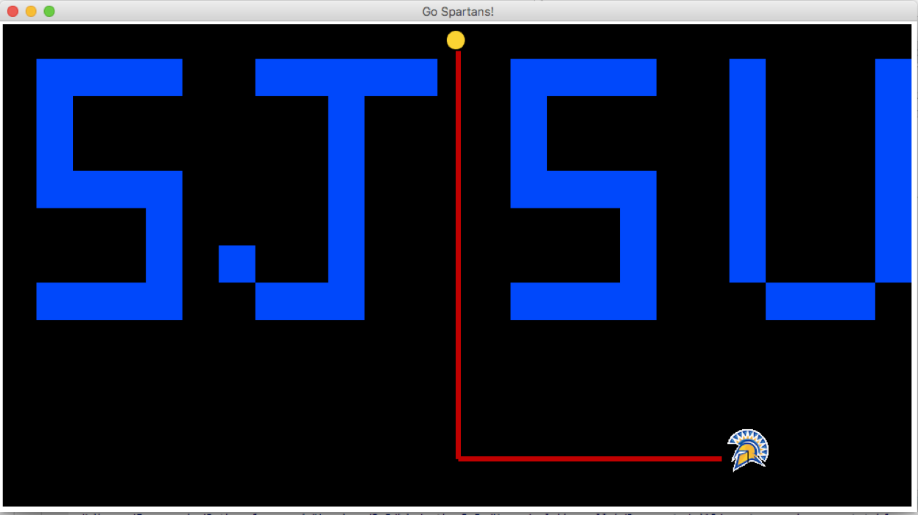
The Manhattan distance between two points P1 and P2 is the sum of the absolute differences of  their coordinates: abs(x1- x2) + abs(y1 - y2)



We'll use the Manhattan distance in our upcoming assignments to guide Sammy the Spartan in his quest to collect medals in a grid environment.



Your task is to write the 4 Python function definitions described below.  Distances are all Manhattan distances.

1. Write a function, *manhattan\_distance* that takes as arguments**two tuples** representing the coordinates of two points in a plane, and returns the Manhattan distance between them.

For example:

manhattan\_distance((2, 7), (1, 3)) should return 5.  
manhattan\_distance((0, 1), (1, 3)) should return 3.  
manhattan\_distance((2, 7), (4, 2)) should return 7.  
manhattan\_distance((5, 2), (4, 4)) should return 3.  
manhattan\_distance((1, 5), (1, 5)) should return 0.

1. Write a function, *max\_distance* that takes as arguments two tuples: *point1*and *other\_points*.

The first tuple, *point1*, represents the coordinates of a point in a plane.The second tuple, *other\_points*,  contains an unspecified number of tuples each of which represents a point in a plane. The function returns the Manhattan distance between *point1* and the farthest other point.

For example:

max\_distance((3, 4), ((1, 2), (4, 5),(4, 3), (9, 2), (0, 1))) should return 8.  
max\_distance((3, 4), ()) should return 0.  
max\_distance((3, 4), ((1, 2),(4, 5),(4, 3),(0, 1)))  should return 6.  
max\_distance((3, 4), ((1, 2), (4, 5), (4, 3))) should return 4.

1. Write a function, *closest\_point* that takes as arguments two tuples: *point1*and *other\_points*.

The first tuple, *point1*, represents the coordinates of a point in a plane.The second tuple, *other\_points*,  contains an unspecified number of tuples each of which represents a point in a plane. The function returns the coordinates of the closest other point to point1.  If there are two close points at the same distance, any one of the two points may be returned.

For example:

closest\_point((3, 4),((1, 2),(4, 5),(9, 2), (0, 1))) should return (4, 5).  
closest\_point((3, 4), ()) should return the Python constant None (not the string “None”).  
closest\_point((3, 4), ((1, 2), (4, 5), (3, 4), (0, 1))) should return (3, 4).  
closest\_point((3, 4), ((1, 2), (3, 3), (4, 3))) should return (3, 3).

1. Write a function, *farthest\_points\_distance* that takes as an argument a tuple, *points*, that  contains an unspecified number of tuples each of which represents a point in a plane.

The function returns maximum distance between all the points given.

For example:

farthest\_points\_distance(((1, 2), (4, 3), (9, 2), (4, 5),(0, 1))) should return 10.

farthest\_points\_distance(()) should return 0.

A starter file, [homework1.pyPreview the documentView in a new window](https://sjsu.instructure.com/courses/1259343/files/48995931/download?verifier=oKo0rQoPEYvLNCpUwwrKGDiJ78DfDihRPSSzyl9m&wrap=1) and a unit testing program [testhomework1.pyPreview the documentView in a new window](https://sjsu.instructure.com/courses/1259343/files/48995999/download?verifier=u6Fi8cKf9bW0gNoabUXzyHHsC4c2OqUJdRCDS5Nc&wrap=1) are provided for your convenience.

Please note that to get any credit on this assignment, the function must be implemented in Python 3.5 or later.

**Make sure you read and follow the grading rubric to ensure full credit.**

Rubric

**Homework 1 Grading Rubric**

| Homework 1 Grading Rubric | | |
| --- | --- | --- |
| **Criteria** | **Ratings** | **Pts** |
| This criterion is linked to a Learning Outcomemanhattan\_distance correctness | |  |  | | --- | --- | | 1.0 pts  Full Marks | 0.0 pts  No Marks | | 1.0 pts |
| This criterion is linked to a Learning Outcomemax\_distance correctness | |  |  | | --- | --- | | 2.0 pts  Full Marks | 0.0 pts  No Marks | | 2.0 pts |
| This criterion is linked to a Learning Outcomemax\_distance style  *The implementation uses Python* ***generator expressions and the built-in function max.*** | |  |  | | --- | --- | | 1.0 pts  Full Marks | 0.0 pts  No Marks | | 1.0 pts |
| This criterion is linked to a Learning Outcomeclosest\_point correctness | |  |  | | --- | --- | | 2.0 pts  Full Marks | 0.0 pts  No Marks | | 2.0 pts |
| This criterion is linked to a Learning Outcomeclosest\_point style  *The implementation uses Python built-in function* ***min with a key.*** | |  |  | | --- | --- | | 1.0 pts  Full Marks | 0.0 pts  No Marks | | 1.0 pts |
| This criterion is linked to a Learning Outcomefarthest\_points\_distance correctness | |  |  | | --- | --- | | 2.0 pts  Full Marks | 0.0 pts  No Marks | | 2.0 pts |
| This criterion is linked to a Learning Outcomefarthest\_points\_distance style  *The implementation uses Python* ***generator expressions and the built-in function max.*** | |  |  | | --- | --- | | 1.0 pts  Full Marks | 0.0 pts  No Marks | | 1.0 pts |
| Total Points: 10.0 | | |

[Previous](https://sjsu.instructure.com/courses/1259343/modules/items/9392345)[Next](https://sjsu.instructure.com/courses/1259343/modules/items/9393510)