

# HW1: Python

---

<b>Due</b>	Jan 20 by 11:59pm	<b>Points</b>	25	<b>Submitting</b>	a file upload	<b>File Types</b>	py
------------	-------------------	---------------	----	-------------------	---------------	-------------------	----

---

**Problem 1 (5 points)** Write a function in python

def fun(x, y):

that if called with

fun(1, 2)

returns 3

and if called with

fun("hi", "there")

returns "hithere"

**Your function should work for any input as long as both inputs are strings or both inputs are numbers.**

**Please name the file you upload for this problem: HW1P1Lastname.py (where Lastname is your last name)**

**Problem 2 (10 points)**

Write a function in python

def report(xs):

that generates a report about the residents of an apartment building in the following way. It takes in a list formatted so that each apartment number is followed by the names of everyone who lives in that apartment. It should return the average number of people who live in an apartment, and the largest number of people who live in an apartment, as a list.

For instance,

report([100, "Jill Johnson", "Billy Ray Cyrus", 110, "Shweta Agarwal", 120, "Miguel Rosas", "Elena Rosas", "Mateo Rosas", 200, "Jason Chan", 210, "Rosalia Torres"])

returns

[1.6, 3]

Your function should work for any input list formatted similarly to the example.

**Please name the file you upload for this problem: HW1P2Lastname.py (where Lastname is your last name)**

**Problem 3 (10 points)** Translate the C++ program [here](#)

(<https://camino.instructure.com/courses/87658/files/6534967?wrap=1>). ↓

([https://camino.instructure.com/courses/87658/files/6534967/download?download\\_frd=1](https://camino.instructure.com/courses/87658/files/6534967/download?download_frd=1)) to Python (Quicksort)

**Please name the file you upload for this problem: HW1P3Lastname.py (where Lastname is your last name)**

**Note: The following problem will be due with Programming HW 2 , but I HIGHLY suggest you start it early (But wait to submit it with HW2). For some of you, this will be the hardest programming problem you've ever been asked to solve.**

1. (15 points) In this problem, you will write a Python program to solve the 8-Queens problem. In chess, a queen can move any number of squares horizontally, vertically, or diagonally. The 8-queens problem is the problem of trying to place eight queens on an empty chessboard in such a way that no queen can attack any other queen. This problem is intriguing because there is no efficient algorithm known for solving the general problem. Rather, the straightforward algorithm of trying all possible placements is most often used in practice, with the only optimization being that each queen must be placed in a separate row and column:
  1. Starting with the first row, try to place a queen in the current column.
  2. If you can safely place a queen in that column, move on to the next column
  3. If you are unable to safely place a queen in that column, go back to the previous column, and move that queen down to the next row where it can safely be placed. Move on to the next column.

Write a program in python to solve the 8-queens problem. Your program should produce as output an 8X8 diagram of the chessboard, with a 1 indicating the presence of a queen in a square and a 0 indicating the absence of a queen. You need to show the **first** solution found. Solutions that solve N Queens or find all solutions will not be accepted. If you have solved this or a similar problem in the past, I ask that you not consult your old code, but start from scratch. Googling the problem is cheating.

Hints: You can represent the board as either an 8X8 list or as a one-dimensional list with the ith item representing the row number of the queen in column i. You can solve this problem recursively or iteratively, but the recursive solution is usually much easier.