

For this problem, you are only allowed to use standard python libraries. You may not use third-party libraries or call any shell/bash functions.

1 Problem 1

You are given a list of tuples of the form $(\langle \text{float} \rangle x, \langle \text{float} \rangle y, \langle \text{float} \rangle r)$ (let's call these *c-tuples*). Each *c-tuple* represents a circle on a rectangular coordinate space, with **x** and **y** being the coordinates of the center, and **r** being the radius. Assume that each *c-tuple* has a unique radius.

Let a *cluster* of circles be a group of circles where each circle in the group overlaps with at least one other circle in that group. Formally, first let a path be formed between two circles when they overlap. Define a *cluster* as a group of n circles, where each circle is reachable from every other circle through the formed paths.

Write a python script that does the following: For each *cluster*, the circle with the largest area is kept, and all other circles in that *cluster* are removed. Return the resulting list of *c-tuples*. Some examples are shown.

Input: $[(0.5, 0.5, 0.5), (1.5, 1.5, 1.1), (0.7, 0.7, 0.4), (4, 4, 0.7)]$
Output: $[(1.5, 1.5, 1.1), (4, 4, 0.7)]$

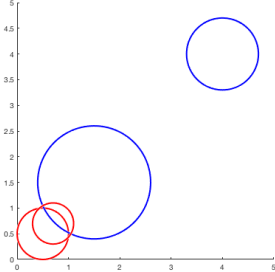


Figure 1: The bottom left circles form a cluster. Red circles are removed.

Input: $[(1.5, 1.5, 1.3), (4, 4, 0.7)]$
Output: $[(1.5, 1.5, 1.3), (4, 4, 0.7)]$

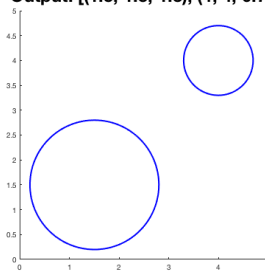


Figure 2: No clusters are found, and no circles are removed.

Input: $[(1, 3, 0.7), (2, 3, 0.4), (3, 3, 0.9)]$
Output: $[(3, 3, 0.9)]$

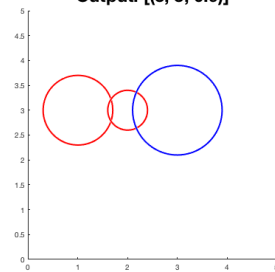


Figure 3: The three circles form a cluster. Red circles are removed.