

Challenge Set#2

Problem#1

Canada is cold in winter, but some parts are colder than others. Your task is very simple, you need to find the coldest city in Canada. So, when given a **list** of cities and their temperatures, you are to determine which city in the list has the lowest temperature and is thus the coldest.



Input:

- The input should be a **list of strings**, like the ones shown below.
- There should be a single space between the city name and the temperature.
- No city name should contain any whitespace.
- There should be at least 7 cities in the list.
- Each element in the **list** should be a **string** (*no integers*, just characters).
- Some of the temperatures should be below zero and some should be above zero.

Output:

You are to output the name of the coldest city on a single line with no whitespace before or after the name. You may assume that there will not be more than one city which is the coldest.

Sample Input:

Saskatoon -20
Toronto -2
Winnipeg -40
Vancouver 8
Halifax 0
Montreal -4
Waterloo -3

Output for Sample Input Winnipeg

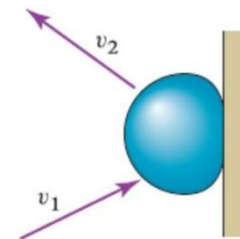
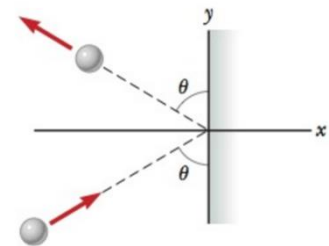
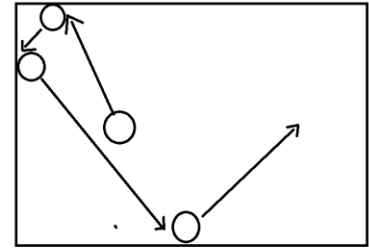
Problem#2

Use python turtle or the Pygame library to create a visual simulation of a ball (or object) that **appears to bounce of the walls of square area enclosed by 4 walls.**

Your program must adhere to the following:

- The object should always start in a random location in a random direction.
- The object should bounce in a way that is **not** symmetrical. Meaning the outward angle should *not* equal the entering angle. **The exiting angle should be random, but in a range that is 10° greater or less than the inbound angle.**
- The speed of the simulation should allow for collision to occur about every second (approximately). We don't want to wait several seconds between each collision.

Bonus: animate each collision so it looks like the ball is **deformed when it hits the wall** and then reformed when it bounces away again.



Problem#3

When a credit card number is sent through the Internet it must be protected so that other people cannot see it. Some older browsers used a protection method involving "RSA Numbers." RSA numbers are numbers that have **exactly four divisors**. In other words, **there are exactly four numbers that divide into it evenly**.



For example:

10 is an RSA number because it has exactly four divisors (1, 2, 5, 10).

11 is not an RSA number it has only two divisors (1,11)

12 is not an RSA number because it has too many divisors (1, 2, 3, 4, 6, 12).

13 is not an RSA number because it has only two divisors (1,13)

14 is an RSA number because it has exactly four divisors (1, 14, 2, 7)

15 is an RSA number because it has exactly four divisors (1, 15, 3, 5)

Write a program that inputs a range of numbers and then **counts how many numbers from that range are RSA numbers**. Your program should also list all the RSA numbers are in the given range. You may assume that the numbers in the range are less than 1000

Sample Session 1

Enter lower limit of range: 10

Enter upper limit of range: 13

Output: The number of RSA numbers between 10 and 13 is: 1

RSA List:

12

Sample Session 2

Enter lower limit of range User Input: 11

Enter upper limit of range User Input: 15

Output: The number of RSA numbers between 11 and 15 is: 2

RSA List:

14

15