APS106 – Lab #7

Preamble

This week you will practice using dictionaries, tuples, sets, and strings to build a program that checks whether a chemical equation is balanced.

Deliverables

For this lab, you must submit the five functions listed below within a single file named 'lab7.py' to MarkUS by the posted deadline.

Functions to implement for this lab:

- `mol_form`
- `expr_form`
- `find_unbalanced_atoms`
- `check_eqn_balance`

Use appropriate variable names and place comments throughout your program.

The name of the source file must be "lab7.py".

Five test cases are provided on MarkUs to help you prepare your solution. **Passing all these test cases does not guarantee your code is correct.** You will need to develop your own test cases to verify your solution works correctly. Your programs will be graded using ten secret test cases. These test cases will be released after the assignment deadline.

IMPORTANT:

- Do not change the file name or function names
- Do not use input() inside your program

Problem

Your task for this lab is to build a program that will check if a chemical equation, stored as a Python dictionary, is properly balanced. To do this, you will write the following three functions:

1. `mol_form`
2. `expr_form`
3. `find_unbalanced_atoms`
4. `check_eqn_balance`

These functions will be discussed in greater detail below.

Understanding the Problem

This section is a brief review of chemical equations and how to determine if they are balanced. If you are familiar with how to balance chemical equations, please feel free to skip directly to the next section.

What is a Chemical Equation?

A chemical equation is a representation of a chemical reaction within a closed system. The equation represents the chemical reactants and products using their symbols and compound formulas. Typically, the reactants are shown on the left-hand side and the products are shown on the right-hand side. Consider the following example:

$$C_3H_8 + 5O_2 \leftrightarrows 4H_2O + 3CO_2$$

In this example, the reactants are C_3H_8 *and* O_2 (propane and oxygen, respectively) and the products are H_2O *and* CO_2 (water and carbon dioxide).

Balanced Chemical Equation

Due to the law of conservation of mass, the number of atoms for each element on both sides of a chemical equation must be equal. The number of each type of atom can be extracted from the subscripts in the chemical equation and the numeric coefficients in front of each molecular formula. Let's analyze the example above to extract the number of atoms for each element on both sides of the equation. Let's begin with carbon (C), on the left-hand side, carbon appears once in the molecule C_3H_8 and has a subscript of 3. The coefficient for C_3H_8 is one, the total number of carbon atoms on the left-hand side is the product of the subscript and the coefficient, or, in this example, $1 \cdot 3 = 3$. On the right-hand side, carbon appears once in CO_2 and has a coefficient of 3, so the number of carbon atoms on the right-hand side is also 3. You can perform the same analysis for the other elements to find the following:

	C	**H**	**O**
Reactants (LHS)	3	8	10
Products (RHS)	3	8	10

If all the elements have the same number of atoms on both sides of the equation, we say that the equation is balanced. If there are any elements that have different numbers of atoms on the right- or left-hand side, we would say that the equation is not balanced.

For more help with balancing chemical equations, see here and here.

Motivating the Problem

Imagine you are a chemical process engineer working on a new chemical process to generate a valuable new compound. After extensive research, you think you've found an efficient process to produce the compound and make a fortune! However, your process includes dozens of steps involving chemical equations like:

$$4\ NH_4OH + KAl(SO_4)_2 \cdot 12H_2O \rightarrow Al(OH)_3 + 2\ (NH_4)_2SO_4 + KOH + 12\ H_2O$$

With this level of complexity and the number of atoms to account for, you think its likely you've made at least one error when checking that the equations were balanced. Worse, if you did make a mistake and you try to execute the process, the yield of the compound would be much lower than you think and cost your company millions of dollars. Luckily, you took APS106 and decide to write a program to check all your equations are balanced and identify any errors…

Part 1 – Extract Elements and Number of Atoms from a Molecular Formula

For the first part of this lab, you will complete the `mol_form` function. This function converts a string representation of a molecule or compound into a Python **dictionary** the molecule's chemical element symbols as keys and the number of atoms within the molecule as values.

You may assume the input string representing the compound structure will specify the number of atoms for each element, even if the IUPAC formula would omit the subscript 1. For instance, ethanol, C_2H_6O, would be passed to the function as the string "C2H6O1". Each element will be representing using either one or two letters according to its atomic symbol. The first letter will always be capitalized and the second letter, should it exist, will be lowercase. A list of all the symbols for chemical elements can be found here.

The keys in the returned dictionary are strings corresponding to the chemical's symbol. The values in the returned dictionary should be integers matching the number of atoms of a particular element within the input molecule.

Example Test Cases

Input Argument	Return Value
"C2H6O1"	{'C': 2, 'H': 6, 'O': 1}
"H2O1"	{'H': 2, 'O': 1}
"K1Fe4"	{'K': 1, 'Fe': 4}

Part 2 – Summing Atoms in a Chemical Expression

In this part of the lab you will complete the `expr_form` function. This function counts the number of atoms for each element in a chemical expression (i.e. one side of a chemical equation). There are two inputs to this function. The first input, `expr_coeffs`, is a **tuple** of integers**.** These integers represent the coefficients for molecules within a chemical expression. The second input, `expr_molecs,` is a tuple of dictionaries representing the molecules within the expression. These dictionaries have the form `{'atomic symbol' : number of atoms}` (similar to the dictionary returned by the `mol_form` function). The order of the coefficients corresponds to the order of molecule dictionaries. For example, if we were interested in counting the atoms belonging to each element in the following expression,

$$2NaCl + H_2 + 5NaF$$

The inputs to the function would be:

```
(2,1,5)
```

and

```
({"Na":1, "Cl":1}, {"H":2}, {"Na":1, "F":1})
```

for `expr_coeffs` and `expr_molecs` respectively.

Note that each tuple contains 3 elements because the chemical expression contains 3 molecules ($NaCl, H_2, and\ NaF$). The first tuple contains the integers (2, 1, 5) corresponding to the numerical coefficient for each molecule in the expression. The second tuple contains dictionaries representing each molecule by storing the elements that make up the molecule as keys and the subscript for each element in the molecular formula as values.

This function **creates and returns a new dictionary** that contains all the elements within the expression as keys and the total number of atoms for each element as values. For the input tuple above, the function would return:

```
{"Na" : 7, "Cl" : 2, "H" : 2, "F" : 5}
```

Example Test Cases

Chemical Expression	**expr_coeffs input**	**expr_molecs input**	**Return Value**
$2CO_2 + 4H_2O$	`(2,4)`	`({"C":1,"O":2}, {"H":2,"O":1})`	`{"C":2, "H":8,"O":8}`
$Al + O_2$	`(1,1)`	`({"Al":1},{"O":2})`	`{"Al" : 1, "O" : 2}`

Part 3 – Comparing Expressions to Find Unbalanced Elements

In this part of the lab you will complete the `find_unbalanced_atoms` function. This function accepts two dictionaries as input parameters representing two sides of a chemical equation and **returns a set** containing any elements that are not balanced in the equation.

The dictionaries input to the function are similar to the dictionaries returned by the `expr_form` function. That is, the atomic symbols for the elements of an expression are the keys and the corresponding number of atoms for each element in the expression are the values.

For example, the equation

$$C_3H_8 + 5O_2 \leftrightarrows 4H_2O + 3CO_2$$

would be input as the following two dictionaries:

```
{"C":3, "H":8, "O":10}
```

and

```
{"H":8, "O":10, "C":3}
```

In this example, because the dictionaries have the same key-value pairs, the function would return an empty set.

If the function were passed an unbalanced equation, such as

$$C_3H_8 + 5O_2 \leftrightarrows 4H_2O + 2CO_2$$

as

```
{"C":3, "H":8, "O":10}
```

and

```
{"H":8, "O":8, "C":2}
```

the function would return the following set to indicate the carbon (C) and oxygen (O) atoms are not balanced:

```
{"C", "O"}
```

Developing test cases for this function, in addition to those presented above is left as an exercise. You should not assume that the set of keys in both dictionaries will be the same,

Part 4 – Putting Everything Together

In this final part of the lab, you will complete the `check_eqn_balance` function to check whether a chemical equation is balanced. The function should call the three functions you wrote in parts 1-3.This function accepts **two nested tuples** representing a chemical equation as input parameters and **returns a set** containing any unbalanced elements in the equation.

Each one of the input tuples represents one side of a chemical equation. Each of these tuples contains two elements. The first element is a tuple of molecule coefficients. The second element is a tuple of strings containing molecule compound formulas. For example, the equation

$$C_3H_8 + 5O_2 \leftrightarrows 4H_2O + 3CO_2$$

would be passed into this function as two tuples. First the reactant (left-hand side) tuple would be:

```
((1,5),("C3H8","O2")).
```

The product (right-hand side) tuple would be:

```
((4,3),("H2O1","C1O2")).
```

In this example, the equation is balanced, so the function would return an empty set.

The following snippets illustrate how the function can be used. You are encouraged to read through this function and understand how it works as an additional exercise.

Example Test Cases

Chemical Eqaution	Reactants input	Products input	Function Output
$C_3H_8 + 5O_2 \leftrightarrows 4H_2O + 3CO_2$	`((1,5),("C3H8","O2"))`	`((4,3),("H2O1","C1O2"))`	set()
$C_3H_8 + 5O_2 \leftrightarrows 4H_2O + 2CO_2$	`((1,5),("C3H8","O2"))`	`((4,2),("H2O1","C1O2"))`	`{"C", "O"}`