

Periodic Words

This problem concerns two different tasks using the Periodic table of elements shown below.

Hydrogen 1 H 1.00794																	Helium 2 He 4.002602				
Lithium 3 Li 6.941	Beryllium 4 Be 9.0122															Boron 5 B 10.811	Carbon 6 C 12.011	Nitrogen 7 N 14.007	Oxygen 8 O 15.999	Fluorine 9 F 18.998	Neon 10 Ne 20.180
Sodium 11 Na 22.990	Magnesium 12 Mg 24.305															Aluminum 13 Al 26.982	Silicon 14 Si 28.086	Phosphorus 15 P 30.974	Sulfur 16 S 32.06	Chlorine 17 Cl 35.453	Argon 18 Ar 39.948
Potassium 19 K 39.098	Calcium 20 Ca 40.078	Scandium 21 Sc 44.956	Titanium 22 Ti 47.887	Vanadium 23 V 50.942	Chromium 24 Cr 51.996	Manganese 25 Mn 54.938	Iron 26 Fe 55.845	Cobalt 27 Co 58.933	Nickel 28 Ni 58.693	Copper 29 Cu 63.546	Zinc 30 Zn 65.39	Gallium 31 Ga 69.723	Germanium 32 Ge 72.61	Arsenic 33 As 74.922	Selenium 34 Se 78.96	Bromine 35 Br 79.904	Krypton 36 Kr 83.80				
Rubidium 37 Rb 85.468	Strontium 38 Sr 87.62	Yttrium 39 Y 88.906	Zirconium 40 Zr 91.224	Niobium 41 Nb 92.906	Molybdenum 42 Mo 95.94	Technetium 43 Tc 98	Ruthenium 44 Ru 101.07	Rhodium 45 Rh 102.91	Palladium 46 Pd 106.42	Silver 47 Ag 107.87	Cadmium 48 Cd 112.41	Indium 49 In 114.82	Tin 50 Sn 118.71	Antimony 51 Sb 121.76	Tellurium 52 Te 127.60	Iodine 53 I 126.905	Xenon 54 Xe 131.29				
Cesium 55 Cs 132.91	Barium 56 Ba 137.33	* 57-70		Lanthanum 57 La 138.905	Hafnium 72 Hf 178.49	Tantalum 73 Ta 180.95	Tungsten 74 W 183.84	Rhenium 75 Re 186.21	Osmium 76 Os 190.23	Iridium 77 Ir 192.22	Platinum 78 Pt 195.08	Gold 79 Au 196.97	Mercury 80 Hg 200.59	Thallium 81 Tl 204.38	Lead 82 Pb 207.2	Bismuth 83 Bi 208.98	Polonium 84 Po [209]	Astatine 85 At [210]	Radon 86 Rn [222]		
Francium 87 Fr [223]	Radium 88 Ra [226]	* 89-102																			
* Lanthanide series				Lanthanum 57 La 138.91	Cerium 58 Ce 140.12	Praseodymium 59 Pr 140.91	Nd 60 Nd 144.24	Promethium 61 Pm [145]	Samarium 62 Sm 150.36	Europium 63 Eu 151.96	Gadolinium 64 Gd 157.25	Terbium 65 Tb 158.93	Dysprosium 66 Dy 162.50	Ho 67 Ho 164.93	Erbium 68 Er 167.26	Thulium 69 Tm 168.93	Ytterbium 70 Yb 173.04				
** Actinide series				Actinium 89 Ac [227]	Thorium 90 Th 232.04	Protactinium 91 Pa 231.04	Uranium 92 U 238.03	Np 93 Np [237]	Pu 94 Pu [244]	Am 95 Am [243]	Cm 96 Cm [247]	Bk 97 Bk [247]	Cf 98 Cf [251]	Es 99 Es [252]	Fm 100 Fm [257]	Md 101 Md [258]	No 102 No [259]				

To complete this task you have been given the completed `Element` class representing the Periodic Table and stores just the element symbol and number. Each element symbol is change to all Capital letters. That is, He becomes HE, Uub becomes UUB and Al becomes AL.

```

public class Element
{
    private String symbol;
    private int number;

    public Element(String s, int n)
    {
        symbol = s.toUpperCase();
        number = n;
    }

    public String getSymbol()
    {
        return symbol;
    }

    public int getAtomicNumber()
    {
        return number;
    }

    public boolean equals(Object obj)
    {
        Element e = (Element)obj;
        return symbol.equals(e.getSymbol()) && number == e.getAtomicNumber();
    }

    public int hashCode()
    {
        Integer n = new Integer(number);
        return symbol.hashCode() + n.hashCode();
    }
}

```

The following table shows results of the `Element` class and its methods.

The following code	Returns
<pre>Element e1 = new Element("He", 2); Element e2 = new Element("Uub", 112); Element e3 = new Element("Al", 13);</pre>	
<pre>e1.getSymbol();</pre>	"HE"
<pre>e2.getSymbol();</pre>	"UUB"
<pre>E3.getSymbol();</pre>	"AL"

- Special note: after reviewing this problem, it became apparent that `getNumber()` method is not used in this problem. However, it was too late to remove it from the problem.

In this problem you will complete the unrelated methods `getMissingElements()` and `isPeriodicSpellingPossible` in the `PeriodicWords` class.

There are two constructors in the `PeriodicWords` class. One constructor has a single parameter. The parameter contains a list of `Elements` representing the elements currently in stock and is stored in the List `inStock`. Note that if there are two of the same element in stock, the element is contained twice in the List. In general, an element is listed `n` times indicating there are `n` of that of that elements in stock. The second constructor has no parameters and creates the array `elements` and fills it with all elements in the periodic table. That is, the array `elements` contains every element in the periodic table. In both constructors, elements are constructed from upper case and lower case letters as shown on Periodic table, however the `Element` constructor changes all letters to upper case.

*** This implies you will be working with Upper case letters only.

The `getMissingElements()` method has a single parameter `mix`, a List of `Elements` required to complete a project, and returns a List of the elements not currently in stock. The `getMissingElements()` method returns a List of all elements in `mix` that are not in the List `inStock`. Similar to `inStock`, elements are repeated to indicate the quantity needed and the elements are listed in random order. Remember, no `Elements` are added or removed from `mix` or `inStock`.

The following table show sample results of the `getMissingElements`.

The following code	Returns
<pre>List<Element> inStock = new ArrayList<Element>(); inStock.add(new Element("H", 1)); inStock.add(new Element("He", 2)); inStock.add(new Element("O", 8)); inStock.add(new Element("C", 6)); PeriodicWords pw = new PeriodicWords(inStock);</pre>	
<pre>List<Element> mix = new ArrayList<Element>(); mix.add(new Element("H", 1)); mix.add(new Element("Li", 3)); mix.add(new Element("O", 8)); List<Element> missing = pw.getMissingElements(mix);</pre>	
<pre>missing.size();</pre>	1
<pre>missing.get(0);</pre>	<code>new Element("Li", 3)</code>

The `isPeriodicSpellingPossible()` method has a single `String` parameter `name` containing Upper Case letters. The `isPeriodicSpellingPossible()` method returns a `true` if it is possible to spell (create a `String`) `name` using only the symbols found in the Period table.

For example, BACON can be spelled (using the Symbols: Ba-C-O-N) while both TEST (there is a TE, ES and S, there is no T, E, or St) and COMPUTER cannot be spelled.

You may assume `name` will be given in ALL CAPITAL letters and `name.length() > 0`.

The following table show sample results of the `isPeriodicSpellingPossible`.

The following code	Returns
<code>PeriodicWords pw = new PeriodicWords();</code>	
<code>pw.isPeriodicSpellingPossible("BACON");</code>	<code>true</code>
<code>pw.isPeriodicSpellingPossible("HER");</code>	<code>true</code>
<code>pw.isPeriodicSpellingPossible("HERE");</code>	<code>true</code>
<code>pw.isPeriodicSpellingPossible("SPOONS");</code>	<code>true</code>
<code>pw.isPeriodicSpellingPossible("BYTE");</code>	<code>true</code>
<code>pw.isPeriodicSpellingPossible("LIKES");</code>	<code>true</code>
<code>pw.isPeriodicSpellingPossible("TEST");</code>	<code>false</code>
<code>pw.isPeriodicSpellingPossible("COMPUTER");</code>	<code>false</code>