Tartalomjegyzék

Exercises

Capital Letter

Write a function that has one parameter: **word** and returns whether it starts with a capital letter or not (return **True** or **False**).

Bird Talk

Here is a strange language *birdtalk* which takes every word from the English language and translates them into bird language. You have to decode a bird word into English. Implement the **birdtalk_decode** function with two parameters:

- **birdword** a word to decode
- **dictionary** a list of all known English words which are the possible decoded words. The function should return an English word if it's birdtalk variant is the searched (bird)word. If the birdword is not a translation of any English words, then return **None**.

The birdtalk is not a complicated language, but the function **tobird** can do this for us. This function encodes an English word into birdtalk. Don't modify the **tobird** function, write you code into **birdtalk_decode**.

CODE:

```
def tobird(word):
    vowels = "http://wiki.math.bme.huaeiou"http://wiki.math.bme.hu
    birdword = "http://wiki.math.bme.hu"http://wiki.math.bme.hu
    for letter in word:
        if letter in vowels:
            birdword = birdword + letter + "http://wiki.math.bme.huv"http://wiki.math.bme.hu + let
        else:
            birdword = birdword + letter
    return birdword

def birdtalk_decode(birdword, dictionary):
```

Packing

You want to go to a hike and you have more stuff than your backpack.Write a function that decides what to bring along! The function should be called **packing** and have two parameters:

- **volume** the size of your backpack
- **stuff** a list of your stuff, it is a list of the volumes of each item ordered by relevance: most important first

The function should have to return the total volume what you managed to pack. If you pack the first k items then return the sum of their sizes. Adding the next element would exceed the backpack. You don't pack less important stuff even if it would fit better then the more important stuff.

Annagramma

Two words are anagrammas if they consist of the same letters but in a different order. Or they are the same strings.

For example: silent and listen

Write a function **anagram** with one parameter: a list of words. Return True or False whether there is an anagramma or not.

Highway Rest

Highway rest According to an EU regulation, there should be a 200 meters long rest place along the road at every 10th kilometer.

There are some rests already built. Write a python function that calculates the amount of additional rests to build. Call that function **highwayrest** with two parameters:

- **length** the length of the road (a positive number)
- **former** a dictionary of the formerly built rests. The keys are the points where the rest is and the values are the length of that rest in meters.

It may be that you have to build a new rest place or make an existing rest place longer. The rest stations should be placed at 10, 20, 30... kilometers. If the road is 40 kilometers long then you have to build a rest at the 10th, 20th, 30th and 40th kilometers. If a given rest place is not long enough then you shold build the rest of it. The function should return the amount of newly built rests in meters. "http://wiki.math.bme.hu