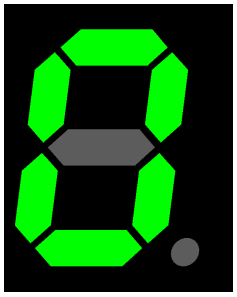
Calculator spelling

A 7-segmentdisplay is used to represent digits (and sometimes also letters). If a screen is held upside down by coincide, the digits may look like letters from the alphabet. This finding is the base of **calculator spelling** (also called **beghilosz**). This is a type of text transformation where words are spelled as a sequence of digits (and letters). After inserting the sequence in a calculator with a 7-segmentdisplay, that is afterwards held upside down, a word can be read on the screen.

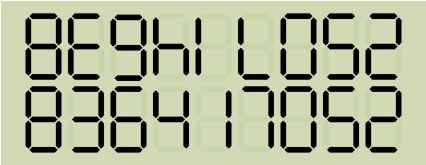


Example of a 7-segmentdisplay with LED.

The base step of a calculator consists of portraying each digit on a letter, as a result of what words that consist of a subset of letters from the alphabet, can be represented on a calculator. The picture below is used in a standard way:

cijfer	0	1	2	3	4	5	6	7	8
letter	O	I	Z	E	h	S	g	L	B

The picture below indicates that the digits 250714638 — based on this image — can be read as BEghILOSZ on a calculator that was turned upside down.



Some calculators leave out the upper segment to represent the digit 6, and the lower segment to represent the digit 9. In that case, an upside down six read a lowercase q, and an upside down nine reads a lowercase b. If we expand the alphabet available to hexadecimal digits (available on most scientific calculators), we can use the following image:

cijfer	0	1	2	3	4	5	6	7	8	9	A	b	C	d	E	F
letter	O	I	Z	E	h	S	g	L	B	b	Y	q	J	P	3	j

If the calculator is held in front of a mirror, we get the following image:

cijfer	0	1	2	3	4	5	6	7	8	9	A	b	C	d	E	F
letter	O	I	S	E	y	Z	a	r	B	e	A	d	J	b	3	z

Assignment

- Write a function digits2letters to which two alphabets of an equal length mus be given. An

alphabet is here represented as a string of unique characters (no character occurs more than once in the alphabet). The function must print a dictionary that portrays every character in the first alphabet on the character that is in the same position in the second alphabet.

- Write a function beghilosz2text to which two arguments must be given: a string and a dictionary that portrays characters on characters. You may assume that each character that occurs in the string, occurs as a key in the dictionary. The function must use the dictionary to portray every character of the given string on its corresponding character. The string that is obtained in this way, must be reversed and printed as the result of the function.
- Write a function letters2digits to which a dictionary must be given that portrays the characters of the first alphabet on the characters of the second alphabet. You may assume that both alphabets have an equal length, and that there is a one-on-one image between the letters of both alphabets. The function must print the upside down image in the format of a dictionary that portrays the characters of the second alphabet on the corresponding characters from the first alphabet.
- Write a function text2beghilosz to which two arguments must be given: a string and a dictionary that portrays characters on characters. You may assume that every character that occurs in the string, occurs as a key in the dictionary, or that if a character is a letter, its form as an uppercase of a lowercase letter at least occurs in the dictionary. The function must use the dictionary to portray every character of a given string on its corresponding character. If the character is a letter that doesn't occur as a key in the dictionary, the image must use the corresponding uppercase/lowercase letter. The string that is obtained in this way, must be turned upside down in order for the result to be printed by the function.

Example

```
>>> c2l = digits2letters('012345678', 'OIZEhSgLB')
>>> c2l
{'1': 'I', '0': 'O', '3': 'E', '2': 'Z', '5': 'S', '4': 'h', '7': 'L', '6': 'g', '8': 'B'}

>>> beghilosz2text('250714638', c2l)
'BEghILOSZ'
>>> beghilosz2text('3722145', c2l)
'ShIZZLE'
>>> beghilosz2text('53177187714', c2l)
'hILLBILLIES'

>>> l2c = letters2digits(c2l)
>>> l2c
{'B': '8', 'E': '3', 'g': '6', 'I': '1', 'h': '4', 'L': '7', 'O': '0', 'S': '5', 'Z': '2'}

>>> text2beghilosz('BEghILOSZ', l2c)
'250714638'
>>> text2beghilosz('SHIZZLE', l2c)
'3722145'
>>> text2beghilosz('hillbillies', l2c)
'53177187714'
```

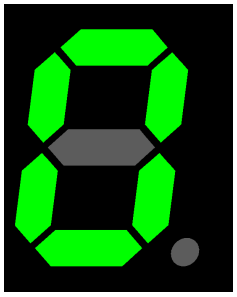
Extra

In the Word Ways issue of November 2010, the American mathematician Mike Keith notices the following happy coincidence:

6499 Blue Springs Pkwy
Mosheim, TN 37818

This is the address of a shop in the east of Tennessee (USA), that is supposedly owned by the Bible family. Insert the digits of the postal code in a calculator and then turn in upside down.

Een 7-segmentdisplay wordt gebruikt om cijfers (en soms ook letters) voor te stellen. Als bij toeval is het zo dat als het scherm ondersteboven gehouden wordt, de voorstelling van de meeste cijfers de vorm aannemen van een letter uit het alfabet. Deze vaststelling ligt aan de basis van **rekenmachinespelling** (ook wel **beghilosz** genoemd). Dit is een vorm van teksttransformatie waarbij woorden gespeld worden als een reeks cijfers (en letters). Nadat de reeks cijfers (en letters) werd ingegeven in een rekenmachine met een 7-segmentdisplay, die daarna wordt omgekeerd, kan het woord op het scherm uitgelezen worden.


Voorbeeld van een 7-segmentdisplay met leds.

De basisstap van rekenmachinespelling bestaat er dus in om elk cijfer af te beelden op een letter, waardoor woorden op een rekenmachine kunnen voorgesteld worden die bestaan uit een deelverzameling van de letters van het alfabet. Standaard wordt hierbij de volgende afbeelding gebruikt:

cijfer	0	1	2	3	4	5	6	7	8
letter	O	I	Z	E	h	S	g	L	B

De onderstaande figuur geeft aan dat de cijfers 250714638 — op basis van deze afbeelding — op een omgekeerde rekenmachine kunnen gelezen worden als BEghILOSZ.

Sommige rekenmachines laten het bovenste segment weg om het cijfer 6 voor te stellen, en het onderste segment om het cijfer 9 voor te stellen. In dat geval leest een omgekeerde zes als de kleine letter q, en leest een omgekeerde negen als de kleine letter b. Als we het beschikbare alfabet uitbreiden naar de hexadecimale cijfers (algemeen beschikbaar op de meeste wetenschappelijke rekenmachines), dan kunnen we bijvoorbeeld de volgende afbeelding gebruiken:

cijfer	0	1	2	3	4	5	6	7	8	9	A	b	C	d	E	F
letter	O	I	Z	E	h	S	g	L	B	b	Y	q	J	P	3	j

Als de rekenmachine voor een spiegel gehouden wordt, dan krijgen we de volgende afbeelding:

cijfer	0	1	2	3	4	5	6	7	8	9	A	b	C	d	E	F
letter	O	I	S	E	y	Z	a	r	B	e	A	d	J	b	3	z

Opgave

- Schrijf een functie cijfers2letters waaraan twee alfabetten van gelijke lengte moeten doorgegeven worden. Een alfabet wordt hierbij voorgesteld als een string van unieke karakters (geen enkel karakter komt meerdere keren in het alfabet voor). De functie moet een dictionary teruggeven die elk karakter in het eerste alfabet afbeeldt op het karakter op dezelfde positie in het tweede alfabet.
- Schrijf een functie beghilosz2tekst waaraan twee argumenten moeten doorgegeven worden: een string en een dictionary die karakters afbeeldt op karakters. Je mag ervan uitgaan dat elk karakter dat voorkomt in de string als sleutel voorkomt in de dictionary. De functie moet de dictionary gebruiken om elk karakter van de gegeven string af te beelden op zijn corresponderende karakter. De string die op die manier bekomen wordt, moet omgekeerd worden en als resultaat teruggegeven worden door de functie.
- Schrijf een functie letters2cijfers waaraan een dictionary moet doorgegeven worden die de karakters van een eerste alfabet afbeeldt op de karakters van een tweede alfabet. Je mag ervan uitgaan dat beide alfabetten even lang zijn, en dat er een één-op-één afbeelding bestaat tussen de letters van de twee alfabetten. De functie moet de omgekeerde afbeelding teruggeven onder de vorm van een dictionary die de karakters van het tweede alfabet afbeeldt op de corresponderende karakters van het eerste alfabet.
- Schrijf een functie tekst2beghilosz waaraan twee argumenten moeten doorgegeven worden: een string en een dictionary die karakters afbeeldt op karakters. Je mag ervan uitgaan dat elk karakter dat voorkomt in de string als sleutel voorkomt in de dictionary, of dat als het karakter een letter is, minstens zijn vorm als hoofdletter of als kleine letter als voorkomt in de dictionary. De functie moet de dictionary gebruiken om elk karakter van de gegeven string af te beelden op zijn corresponderende karakter. Als het karakter een letter is die niet als dusdanig als sleutel voorkomt in de dictionary, dan moet de afbeelding gebruikt worden van de overeenkomstige hoofdletter/kleine letter. De string die op die manier bekomen wordt, moet omgekeerd worden en als resultaat teruggegeven worden door de functie.

Voorbeeld

```
>>> c2l = cijfers2letters('012345678', 'OIZEhSgLB')
>>> c2l
{'1': 'I', '0': 'O', '3': 'E', '2': 'Z', '5': 'S', '4': 'h', '7': 'L', '6': 'g', '8': 'B'}

>>> beghilosz2tekst('250714638', c2l)
'BEghILOSZ'
>>> beghilosz2tekst('3722145', c2l)
'ShIZZLE'
>>> beghilosz2tekst('53177187714', c2l)
'hILLBILLIES'

>>> l2c = letters2cijfers(c2l)
>>> l2c
{'B': '8', 'E': '3', 'g': '6', 'I': '1', 'h': '4', 'L': '7', 'O': '0', 'S': '5', 'Z': '2'}

>>> tekst2beghilosz('BEghILOSZ', l2c)
'250714638'
```

```
>>> tekst2beghilosz('SHIZZLE', l2c)
'3722145'
>>> tekst2beghilosz('hillbillies', l2c)
'53177187714'
```

Uitsmijter

In de November 2010 editie van Word Ways, merkt de Amerikaanse wiskundige Mike Keith het volgende gelukkig toeval op:

Bible's Machine and Welding
6499 Blue Springs Pkwy
Mosheim, TN 37818

Dit is het adres van een winkel in het oosten van Tennessee (VSA), die vermoedelijk de eigendom is van de familie Bible. Voer de cijfers van de postcode in op een rekenmachine en draai die daarna ondersteboven.