

Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

Zadanie Szyfr – LOGIA (2019/2020), etap 3

Treść zadania

Kolejne litery alfabetu łacińskiego (jest ich 26) uporządkowano w sposób przedstawiony poniżej:

	1	2	3	4	5	6	7	8	9	10	11	12	13
20	a	b	c	d	e	f	g	h	i	j	k	l	m
40	n	o	p	q	r	s	t	u	v	w	x	y	z

Maszyna szyfrująca koduje napisy przypisując każdej literze liczbę otrzymaną w następujący sposób: do numeru kolumny z literą dodajemy liczbę przypisaną do wiersza. Następnie otrzymane liczby sklejane są w jedną liczbę. Na przykład kodem napisu *krab* jest liczba 31452122.

Napisz funkcję **deszyfr(lista)**, której parametrem jest lista złożona z nie więcej niż 100 liczb co najwyżej 60 cyfrowych o parzystej długości odpowiadających wyrazom zaszyfrowanym przez maszynę. Wynikiem funkcji jest lista odszyfrowanych napisów.

Przykłady:

Wynikiem **deszyfr([31452122])** jest ["krab"].

Wynikiem **deszyfr([232821473121, 22212252, 30212729])** jest ["chatka", "baby", "jagi"].

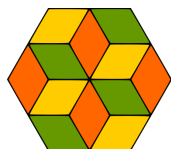
Wynikiem **deszyfr([332131425021, 4321412925413121])** jest ["makowa", "panienka"].

Omówienie rozwiązania

Rozwiązanie rozpoczynamy od napisania pomocniczej funkcji odszyfrowującej pojedynczy wyraz. Ponieważ liczba odpowiadająca zaszyfrowanemu tekstowi jest parzystej długości, możemy brać pod uwagę kolejne pary cyfr z liczby, zaczynając od lewej strony. Wygodniej jest traktować w tym celu liczbę jako napis i wyciąć interesujący nas fragment. Na podstawie tabelki z treści zadania obliczamy, jakie wartości odpowiadają poszczególnym literom. Można zauważyć, że literom od **a** do **m** odpowiadają liczby 21, 22, ..., 33, natomiast literom od **n** do **z** – liczby 41, 42, ..., 53.

Do znalezienia znaku odpowiadającego podanej liczbie można skorzystać z funkcji `chr()`, pamiętając, że kod małej litery **a** to w kodach ASCII liczba 97. Ponieważ szyfrując literę **a** otrzymujemy liczbę 21, to w celu odszyfrowania od kodu litery musimy odjąć 21 i dodać 97. Reguła ta jest poprawna dla pierwszej połowy alfabetu. Dla liter większych od **m** musimy odjąć 41 oraz dodać 13 (ponieważ szyfrujemy drugą połowę alfabetu) i 97 (kod litery **a**). Kolejne odszyfrowane znaki sklejamy budując wyraz zaszyfrowany przez maszynę.

Główna funkcja deszyfrująca składa się z pojedynczej pętli przeglądającej listę liczb i budującej nową listę z odszyfrowanych wyrazów. Przy wywoływaniu funkcji pomocniczej odszyfrowującej pojedyncze wyrazy musimy zadbać o to, by parametrem wywołania był napis, a nie liczba.



Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

Rozwiązanie w języku Python

W rozwiązaniu korzystamy z trzech standardowych funkcji języka Python: `chr()` której wynikiem jest znak o podanym kodzie ASCII, `str()` zamienia liczbę na napis, `int()` zamienia napis na liczbę.

```
1. def deszyfr_w(napis):
2.     wynik = ''
3.     for i in range(0, len(napis) - 1, 2):
4.         liczba = int(napis[i:i+2])
5.         if liczba > 40:
6.             z = chr(liczba - 41 + 13 + 97)
7.         else:
8.             z = chr(liczba - 21 + 97)
9.         wynik += z
10.    return wynik
11.
12. def deszyfr(lista):
13.     wynik = []
14.     for w in lista:
15.         wynik += [deszyfr_w(str(w))]
16.    return wynik
```

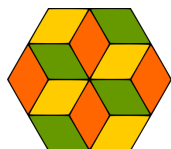
W czwartym wierszu programu wycinamy dwa kolejne znaki z napisu, konwertujemy je na liczbę całkowitą i przypisujemy do zmiennej pomocniczej **liczba**.

Testy

Testując tego typu programy, poza przykładami podanymi w treści zadania, warto wykonać między innymi test dla całego alfabetu, zwracając przy tym szczególną uwagę na kolejność liter. Poniżej przykładowe testy do zadania.

Wywołanie – Python	Wynik
<code>deszyfr([3145423142245232, 32482229, 3229322925, 5042244125])</code>	<code>['krokodyl', 'lubi', 'lilie', 'wodne']</code>
<code>deszyfr([31452122, 3321232821, 46532353524323213329])</code>	<code>['krab', 'macha', 'szczypcami']</code>
<code>deszyfr([235347254552, 27454846533129])</code>	<code>['cztery', 'gruszki']</code>
<code>deszyfr([53292532424152, 4653235352432942452531])</code>	<code>['zielony', 'szczypiorek']</code>
<code>deszyfr([213221, 3321, 31424721, 29, 2450212453292546232921, 432145, 46312145432547])</code>	<code>['ala', 'ma', 'kota', 'i', 'dwadzieścia', 'par', 'skarpet']</code>
<code>deszyfr([4742, 412925, 30254647, 474548244125, 53212421412925, 4121, 4653522645425021412925])</code>	<code>['to', 'nie', 'jest', 'trudne', 'zadanie', 'na', 'szyfrowanie']</code>
<code>deszyfr([212223242526272829303132334142434445464748495051525321222324])</code>	<code>['abcdefghijklmnopqrstuvwxyz']</code>
<code>deszyfr([])</code>	<code>[]</code>

[illegible]



Przedmiotowy Konkurs Informatyczny LOGIA powołany przez Mazowieckiego Kuratora Oświaty

'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz',
'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz',
'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz',
'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz',
'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz',
'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz', 'abcdefghijklmnopqrstuvwxyz',
'abcdefghijklmnopqrstuvwxyz']