Regular Expression Assignments

1. Haigy Paigy

Haigy Paigy is as a children's invented language which sounds exactly like English, except that “aig” is inserted before the vowel sound of each syllable. E.g., the English word “hello” becomes “haigellaigo” in Haigy Paigy. Write a program in Perl that will automatically translate sentences from English into Haigy Paigy. The input of your program will be a string containing one English sentence. Write a set of regular expressions that will translate this sentence into Haigy Paigy.

Simplifications: only vowels can be syllable nuclei; "y" is always a vowel; several consecutive vowel letters always make a single vowel sound; the final "e" is always silent, except when it is the only vowel letter in the word.

Example input: This is a test.

Example output: Thaigis aigis aiga taigest.

2. Russian Declensions

As an inflected language, Russian uses suffixes to mark noun cases. Specifically, the count form of the noun is governed by the preceding cardinal number. For example, if you count students in Russian, you go like this:

- 1 student
- 2 studenta
- 3 studenta
- 4 studenta
- 5 studentov
- 6 studentov

  ...
- 11 studentov
- 12 studentov

  ...
- 20 studentov
- 21 student
- 22 studenta
- 23 studenta
- 24 studenta
- 25 studentov
- 26 studentov

  ...
- 30 studentov
- 31 student
- 32 studenta

  ...

For numbers (n) over a hundred, the endings are the same as for (n – 100).

Write a program in Perl that will automatically inflect the word "student" based on the quantity. The input of your program will be a positive integer number. Write a set of regular expressions that will add the word “student” to this number in the proper noun case.

Example input: 111

Example output: 111 studentov

3. Turkish Plurals

As many other agglutinative languages, Turkish has a feature called **vowel harmony**. Turkish has eight vowels that are divided into two groups as follows:

the **back** vowels are A I O U   the **front** vowels are E İ Ö Ü

Note: Iı and İi are **different** vowels!

There is a single plural suffix in Turkish, but it has two variants (allomorphs): -ler and -lar. The vowel in the suffix mirrors the last vowel of its noun. We must add -lar to words whose final vowel is any of the back vowels, and -ler to words whose final vowel is any of the front vowels.

Examples:

- balta 'axe' : baltalar 'axes'
- kapı 'door' : kapılar 'doors'
- palto 'overcoat' : paltolar 'overcoats'
- boncuk 'bead' : boncuklar 'beads'
- ev 'house' : evler 'houses'
- kedi 'cat' : kediler 'cats'
- göz 'eye' : gözler 'eyes'
- ödül 'award' : ödüller 'awards'

Important: For simplicity and in order to avoid dealing with Unicode, we will encode ö and ü as the upper-case letters O and U, respectfully. Also, we will encode the undotted ı as the upper-case I. For example, we will write **gözler** as **gOzler**, **ödül** as **OdUl**, and **kapı** as **kapI**.

Write a program in Perl that will pluralize Turkish nouns. The input of your program will be a singular Turkish noun. Write a set of regular expressions that will make this noun plural.

Example input: gOz

Example output: gOzler