Word Puzzle

Data Structures Assignment 2
Stacks and Queues
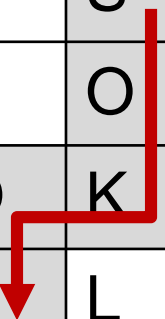
NTHU EECS
2019

https://acm.cs.nthu.edu.tw/problem/12195/

Objective

- Find all possible 'words' in a given matrix

X	S	C	A	T
C	O	Z	D	P
O	K	X	B	X
B	L	C	P	U
L	N	Q	A	V

Rules

■Select a *starting cell* from left to right and from top to bottom

X	S	C	A	T
C	O	Z	D	P
O	K	X	B	X
B	L	C	P	U
L	N	Q	A	V
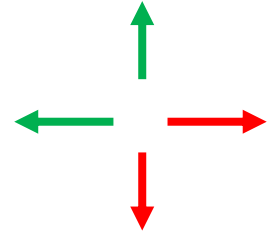
Rules

■For each *starting cell*, you should output *all paths* producing legal words according to the following *priorities (優先順序)*
   ■down
   ■right
   ■up
   ■left

■Each word/path cannot use the same cell more than one time

Rules

- Matrices consist of 26 lowercase characters, where:
  - "a", "e", "i", "o", and "u" are vowels
  - Others are consonants子音

- Legal word formats are:
- The length of the word >= 5
- The regular expression is: $cv^+(c(v)^+)^+c$
  - Where c is a consonant and v is a vowel
  - '+' means once or more

Examples

- Bad words
  - book
  - boook
  - break
  - apple
- Legal words
  - bokok
  - bokaeiouk
  - nation
  - national

Rules

- You should output all possible words in one path
  - The following path contains two legal words, "nation" first and then "national"

n	a			
	t	i		
		o		
		n		
	l	a		

Rules

- You should also output the rearranged word in a different format
  - Vowels should be put in the beginning of the word
  - Where "nation" becomes "aiontn"

Sample Input

Dimension of the matrix

The matrix

```
5↵
xscat↵
cozdp↵
okxbx↵
blcpu↵
lnqav↵
```

Output

- All possible 'words' in the puzzle
  - The 'word' in the traversal order
  - The rearranged format of the word (vowels first)

Sample Output

The second half…

```
sokob ooskb ↵
sokoc ooskc ↵
socob ooscb ↵
socok oosck ↵
cokoz oockz ↵
cokos oocks ↵
cokob oockb ↵
zokob oozkb ↵
zokoc oozkc ↵
zocob oozcb ↵
zocok oozck ↵
kocob ookcb ↵
kocoz ookcz ↵
kocos ookcs ↵
xuvap uaxvp ↵
xuvaq uaxvq ↵
xupav uaxpv ↵
xupaq uaxpq ↵
```

```
bokoz oobkz ↵
bokos oobks ↵
bokoc oobkc ↵
bocok oobck ↵
bocoz oobcz ↵
bocos oobcs ↵
pavux aupvx ↵
puvaq uapvq ↵
qavux auqvx ↵
qavup auqvp ↵
qapuv auqpv ↵
qapux auqpx ↵
vupaq uavpq ↵
vapux auvpx ↵
```