

## Tłumacz węzowego na nasze

Bajtabrajan Pythończyk jest nadwornym tłumaczem króla Bajtazara. Jego specjalność to języki węzowe – kobrzy i pythoński, które są niezwykle trudnymi i złożonymi językami. Konieczność zażyłej komunikacji z lokalną społecznością pythońską sprawia, że tacy tłumacze są niezwykle cenni i potrzebni. Niestety, Bajtabrajan jest jedynym znawcą tych języków – a większość mieszkańców Bajtocji ma istotne problemy z rozróżnieniem języków węzowych.

Bajtazar wpadł na pomysł automatyzacji procesu tłumaczenia z języków węzowych na *nasze* (tj. bajtockie). W tym celu prosi Cię o to, byś zautomatyzował proces tłumaczenia między językiem bajtockim i językami: pythońskim i kobrzym. Oczywiście możesz liczyć na pomoc Bajtabrajana Pythończyka, który już za chwilę zdradzi Ci sekret swojej językowej biegłości.

### Zadanie

Bajtabrajan wysyczał Ci na ucho: *Tak naprawdę zarówno język kobrzy jak i pythoński są bardzo proste. Ssekretem jest odpowiednie czytanie i zapisywanie wyrazów. Przykładowo, w języku kobrzym dowolny teksst składa się z trzech elementów:*

- *wyssokość tekstu w;*
- *pozycja początkowa p;*
- *zapisany tekst.*

*Teksst pisany w języku kobrzym jest węzem, który idzie początkowo w górę. Patrz:*

```
4 2
..j...
.a.a..
k...k.
.....i
```

*Tutaj masz ssłowo **kajaki** w języku kobrzym z wyssokością 4 i początkową pozycją 2. “Jak to 2?” zapewne zapytasz... Otóż numeracja wyssokości zaczyna ssię od zera. Na wyssokości 0 znajduje ssię tylko litera j. Na wyssokości 1 masz dwie litery a. Rozumiesz już ssschemat?*

*To teraz ssłuchaj dalej – pythońsski jessst bardzo podobny. Różni ssię jedynie tym, że początkowy kierunek wężyka jessst dokładnie przeciwny. To ssamo ssłowo z tymi samymi parametrami będzie wyglądało tak:*

```
4 2
....k.
...a.i
k.j...
.a....
```

*Tak więc sssam widzisz – nic trudnego! Ale jeszcze oprócz tego... musssisz być w ssstanie także zapissyywać wiadomości w języku kobrzym i pythońssskim.*

*Wierzę w Ciebie, dasz radę! Pamiętaj: tekst, który przetłumaczysz, ma przypominać węża!*

*Żeby ułatwić Ci poradzenie ssobie z problemem, przedssstawię Ci za chwilę jeszcze kilka przykładów.*



*Miej na uwadze, że od rana do wieczora możesz zostać wiele razy zapythony o przetłumaczenie jakiejś wypowiedzi.*

Napisz program, który będzie mógł być wykorzystany jako automatyczny tłumacz między językiem bajtockim i językami węzowymi.

## Opis wejścia

Wejście składa się z pewnej liczby wiadomości, które należy przetłumaczyć. Każda wiadomość ma jedną z trzech postaci:

- **Wiadomość do odszyfrowania** – składa się z polecenia `odsycz`, po którym następuje wiersz zawierający dwie rozdzielone spacją liczby całkowite  $w$  oraz  $p$ , opisujące wysokość tekstu i pozycję początkową. Następnie pojawia się blok tekstu o pewnej (nieznanej z góry) szerokości  $k$ , zawierający tekst zapisany w którymś węzowym języku.

Z uwagi na brak kompetencji automatu w rozpoznawaniu języka ze słuchu, musisz samodzielnie określić czy dana wiadomość jest w języku kobrzym czy pythońskim. Blok tekstu składa się z kropek oraz z małych liter alfabetu, które odpowiadają kolejnym literom wiadomości (tłumaczony tekst nie zawiera spacji, ani znaków przestankowych, by ułatwić jego tłumaczenie – kropki służą jedynie ustawieniu tekstu w odpowiedniej formie).

- **Wiadomość do zaszyfrowania w języku kobrzym** – składa się z polecenia `SSS`, po którym w kolejnej linii następują dwie liczby  $w$  i  $p$ . Opisują one odpowiednio wysokość tekstu i początkową pozycję, z której zaczynamy pisać. Kolejna linia zawierać będzie wiadomość do przetłumaczenia.
- **Wiadomość do zaszyfrowania w języku pythońskim** – składa się z polecenia `sss`, po którym w kolejnej linii następują dwie liczby  $w$  i  $p$ . Opisują one odpowiednio wysokość tekstu i początkową pozycję, z której zaczynamy pisać. Kolejna linia zawierać będzie wiadomość do przetłumaczenia.

Wejście kończy się poleceniem `cii`, które ma uciszyć (zakończyć pracę) automatycznego tłumacza.

W każdym z przypadków  $1 \leq w \leq 4000$ ,  $1 \leq p \leq w$ . Pojedyncza wiadomość ma nie więcej niż 1000 znaków i składa się wyłącznie z małych liter alfabetu łacińskiego.

Przypominamy, że liczba zapytań nie jest z góry znana, ale nie przekracza ona 25.

## Opis wyjścia

Na wyjściu standardowym dla każdego zapytania powinna pojawić się stosowna odpowiedź. W przypadku prośby o przetłumaczenie wiadomości podanej w jednym z języków węzowych, poprawną odpowiedź stanowi pojedyncza linia z odczytanym tekstem.

W przypadku próśb o zapisanie zadanego tekstu w języku kobrzym lub pythońskim należy na standardowym wyjściu wypisać  $w$  linii tekstu (gdzie  $w$  jest elementem prośby o tłumaczenie określającym wysokość tekstu) zgodnie z formatem danego języka.

Po każdej odpowiedzi powinna zostać wypisana dodatkowa pusta linia (tak jak w drugim z poniższych przykładów).

## Przykład

Dla przykładowego, podanego poniżej wejścia:

```
odsycz
3 2
..j...a.
.u.a.i.k
k...w...
cii
```

prawidłową odpowiedzią jest:

```
kujawiak
```

Z kolei dla przykładowego wejścia:

```
odsycz
4 2
....a....
...r.c...
k.b...z.k
.u.....e.
sss
5 2
mariuszek
SSS
4 1
kolanko
cii
```

prawidłową odpowiedzią jest:

kubraczek

```
.....Z..  
.....S.e.  
m...u...k  
.a.i.....  
..r.....  
  
.o.....  
k.l...o  
...a.k.  
...n..
```

## Wyjaśnienie przykładów

W pierwszym przypadku ciężko stwierdzić z jakim językiem mamy do czynienia (z uwagi na wysokość tekstu i pozycję początkową, ten tekst w obu językach wyglądałby tak samo). Niemniej, idąc kolumna po kolumnie wzdłuż odczytywanego tekstu uzyskujemy kolejno litery **k u j a w i a k**, co tworzy poszukiwane słowo. Następujące po zapytaniu **cii** kończy pierwszy przykład.

W drugim przykładowym wejściu mamy więcej próśb o tłumaczenie.

Pierwsza z nich dotyczy przetłumaczenia wiadomości zapisanej w języku pythońskim – wiadomość odczytujemy w analogiczny sposób jak w poprzednim przykładzie.

Druga prośba dotyczy zapisania słowa **mariuszek** w języku pythońskim. Zadana wysokość tekstu to 5, zaś początkowa pozycja to 2. Pierwszy znak musimy więc postawić w środkowym rzędzie (przypominamy o numerowaniu wierszy od 0). Zgodnie z zasadami języka, kolejna litera musi być postawiona jeden rząd niżej, po prawej, podobnie w przypadku trzeciej litery. Następnie, z uwagi na narzuconą wysokość bloku tekstu, należy *odbić* kierunek zapisu w górę, więc kolejna litera (**i**) będzie znajdować się wiersz wyżej. Druga zmiana kierunku zapisu nastąpi dopiero przy literze **z**.

Ostatnią w tym zestawie jest prośba o przetłumaczenie tekstu na język kobrzy. Zasady są podobne jak w języku pythońskim, jedyną różnicą jest początkowy kierunek zapisywania kolejnych liter.

## Punktacja

Oczywiście jeżeli Twój algorytm podoła jedynie części przypadków testowych, to zostaniesz nagrodzony częściowymi punktami. Poniższa tabela opisuje poszczególne grupy testów obłożone dodatkowymi założeniami.

| Dodatkowe założenia:                                                       | Punkty za grupę testów: |
|----------------------------------------------------------------------------|-------------------------|
| Zapytania dotyczą jedynie tłumaczenia słów z bajtockiego na języki węzowe. | 10                      |
| Rozmiar <sup>1</sup> tłumaczonych słów nie przekracza 1200.                | 25                      |
| Brak dodatkowych ograniczeń.                                               | 65                      |

<sup>1</sup> Rozmiar tutaj rozumiany jest jako iloczyn długości i wysokości słowa.

Biuro Projektu Partnera Wiodącego: Politechnika Łódzka | Wydział Elektrotechniki, Elektroniki, Informatyki i Automatyki |  
ul. Stefanowskiego 22, pokój 14, 90-924 Łódź | tel. (42) 631-28-89, | e-mail: [biuro@cmi.edu.pl](mailto:biuro@cmi.edu.pl) | [www.cmi.edu.pl](http://www.cmi.edu.pl)

Partner Wiodący Projektu



Politechnika Łódzka

Partnerzy Projektu



AGH



POLITECHNIKA  
GDAŃSKA

Politechnika  
Warszawska



Politechnika  
Wrocławska

I<sup>2</sup>math

Cyfrowy  
DIALOG



Fundusze  
Europejskie  
Polska Cyfrowa



Rzeczpospolita  
Polska



CENTRUM  
PROJEKTÓW  
POLSKA  
CYFROWA

Unia Europejska  
Europejski Fundusz  
Rozwoju Regionalnego

