

## Analizator logów

<https://szkopul.edu.pl/problemset/problem/alo/site>

### Luka

Staszek, Franek i Hania zauważyli lukę na rynku. Coraz więcej telefonów, tabletów, kamer podłącza się do sieci w domach i firmach. Może któreś z urządzeń ma wirusa? Może ktoś chce wykorzystać sieć do spamowania? Albo cyfrowego ataku?

### Wchodzimy w to!

Licealiści postanowili napisać własną aplikację. Będzie analizować logi komputerów, routerów by wykryć potencjalne zagrożenia. To temat, w którym wiele jest do zrobienia potrzeba wielu pomysłów, powstanie wiele nowych firm. Dlaczego? Każda firma ma wiele routerów, mnóstwo GB / TB logów. Jak odsiać szum informacyjny od prawdziwego zagrożenia? I pokazać wyniki w sensownym czasie?

### Zespół

Na szczęście Staszek jest mistrzem programowania, Hania zna algorytmy (startowała w olimpiadzie) a Franek wie, jak trafić do firm by spróbowali nowej genialnej, aplikacji!

### Wszystko się zaczyna od v. 1.0

Pierwsza wersja analizatora logów ma po prostu wypisywać ilość połączeń do poszczególnych adresów IP z lokalnej podsieci.

### Wyjaśnienie

Adres IP składa się z 4 oktetów – u nas numerowanych od 0 do 3. Na przykład:

0 1 2 3 - numery oktetów

**192 168 32 46** - adres IP

Każdy oktet to wartość od 0 do 255 (zapisany jest na 8 bitach).

Nasze adres są klasy C. Oznacza to, że pierwsze 3 oktety mówią o adresie sieci, zaś ostatni 4 jest adresem serwera, laptopa, ... - urządzenia w tej sieci.

### Wejście

Pierwsza linia wejścia składa się z jednej liczby naturalnej  $n$  ( $1 \leq n \leq 10^5$ ) oznaczającej ilość poleceń dla aplikacji analizującej logi.

W kolejnych  $n$  liniach znajduje się jedno z 3 możliwych poleceń:

**k** *oktet0 oktet1 oktet2 oktet3* – Wszystkie dane oddzielone spacją. Oznacza, że nastąpiła komunikacja do adresu IP: *oktet0.oktet1.oktet2.oktet3*

**z** *oktet0 oktet1 oktet2 oktet3* – Wszystkie dane oddzielone spacją. **Z**najdź adres IP (*oktet0.oktet1.oktet2.oktet3*) i wypisz:

adres IP oraz ilość połączeń do niego – wszystkie elementy oddzielone spacją

Jeśli nie było połączeń do tego adresu IP, wypisz 0.

**w** – **W**yypisz wszystkie adresy IP do których były połączenia do danego momentu logu wraz z liczbą połączeń do każdego adresu IP:

adres IP oraz ilość połączeń do niego – wszystkie elementy oddzielone spacją

Adresy powinny być uporządkowane następująco:

- a. Najpierw według rosnących adresów hostów (*oktet3*)
- b. W przypadku gdy dwa hosty mają ten sam adres (*oktet3*) należy wypisać adresy według rosnącej części sieciowej (*oktet0 oktet1 oktet2*) – jeśli *oktet0* jest większy adres powinien być później, następnie jeśli *oktet1* jest większy adres powinien być później i w końcu jeśli *oktet3* jest większy adres powinien być później,

## Wyjście

Program powinien wypisać w oddzielnych liniach wyjścia dla poleceń **z** oraz **w** zgodnie z powyższym

## Przykład

### Wejście

9

k 10 20 30 40

k 30 40 10 20

k 15 60 70 20

w

k 10 20 30 40

k 50 10 10 25

z 10 20 30 40

z 10 20 30 80

w

### Wyjście

15 60 70 20 1

30 40 10 20 1

10 20 30 40 1  
10 20 30 40 2  
10 20 30 80 0  
15 60 70 20 1  
30 40 10 20 1  
50 10 10 25 1  
10 20 30 40 2

#### Wyjaśnienie

A. Dla pierwszego polecenia **w (wypisania adresów IP z którymi był kontakt)** mamy 3 adresy IP – do każdego był dostęp raz. Wypisujemy adresy IP wraz liczbą połączeń do nich. Adresy sortujemy rosnąco po ostatnim oktecie (wartość 20 najmniejsza na ostatnim oktecie) a następnie rosnąco po pierwszych trzech oktetach:

15 60 70 20 1  
30 40 10 20 1  
10 20 30 40 1

B. Dla polecenia **z 10 20 30 40** mamy, że do tej pory adres IP wystąpił do tej dwukrotnie -> wypisujemy 2

C. Dla polecenia **z 10 20 30 80** mamy, że do tej pory adres IP nie wystąpił -> wypisujemy 0

D. Dla ostatniego polecenia **w (wypisania adresów IP z którymi był kontakt)** mamy 4 adresy IP. Wypisujemy je w kolejności jak w A. Adres 10 20 30 80 nie wystąpił do tej pory w komunikacji – jedynie w zapytaniu o liczbę połączeń – nie wypisujemy:

15 60 70 20 1  
30 40 10 20 1  
50 10 10 25 1  
10 20 30 40 2