

Opis:

Ze względu na panujące oboszczenia coroczny Wielki turniej ciągów musi odbyć się zdalnie. Zostałeś poproszony o napisanie programu w Javie, który rozstrzygnie wyniki tego konkursu. W turnieju bierze udział n ciągów każdy z nich mający długość m . Turniej składa się z m rund. W każdej rundzie bierze udział n reprezentantów danych ciągów, w pierwszej rundzie są to pierwsze wyrazy tych ciągów. W danej rundzie wygrywa ciąg, którego reprezentant jest największy. Dostaje on punkt, który dodawany jest do ogólnego wyniku danego ciągu (oczywiście każdy ciąg zaczyna turniej z zerową liczbą punktów). Pozostałe ciągi nie otrzymują żadnego punktu. Do kolejnej rundy brani są reprezentanci z poprzedniej rundy poza reprezentantem ciągu wygranego, który jest zastępowany przez kolejny wyraz ciągu z którego pochodził. W przypadku remisu w rundzie, który zdarza się gdy co najmniej dwóch reprezentantów osiąga tą samą największą wartość żaden ciąg nie otrzymuje punktów. Dodatkowo wszyscy reprezentanci danych ciągów zostają zastąpieni przez kolejne wyrazy ciągów z których pochodzili. By zilustrować zasady rozważmy przykład z $n = 3$, $m = 4$ i ciągami:

a_1 : 1 -1 2 3,
 a_2 : 2 0 4 5,
 a_3 : 0 0 1 0.

Turniej będzie trwał 4 rundy.

W rundzie pierwszej reprezentanci ciągów a_1 , a_2 , a_3 będą odpowiednio 1 2 0. Zatem tą rundę wygra ciąg a_2 . W drugiej rundzie reprezentantami będą 1 0 0 i tą rundę wygra ciąg a_1 . W trzeciej rundzie reprezentantami będą -1 0 0 i ta runda kończy się remisem. W czwartej rundzie reprezentantami będą 2 4 0 i tą rundę wygra ciąg a_2 . Ostatecznie ciąg a_1 otrzymał jeden punkt, ciąg a_2 dwa punkty a ciąg a_3 zero punktów.

Program poza wczytywaniem danych powinien działać w złożoności $O(m \log n + (1+r)n)$, gdzie r to liczba remisów, które wystąpiły podczas turnieju i zużywać dodatkowo co najwyżej $O(n)$ dodatkowej pamięci.

Wejście:

Dane do programu wczytywane są ze standardowego wejścia (klawiatury) zgodnie z poniższą specyfikacją:

1. Pierwszą podaną wartością będzie dodatnia liczba całkowita z ($0 < z < 101$), oznaczająca ilość zestawów danych.
2. Każdy zestaw danych ma następującą postać:
 - a. W pierwszej linii znajduje się liczba całkowita n ($1 < n < 10001$), oznaczająca liczbę ciągów oraz liczba całkowita m ($0 < m < 10001$), oznaczająca ich długość.
 - b. Na początku kolejnych n linii znajduje się *nazwa_ciągu* składająca się z co najwyżej 100 znaków (które to mogą być dużymi, małymi literami angielskiego alfabetu lub cyfrą), następnie po spacji m liczb typu int będącymi kolejnymi wyrazami danego ciągu oddzielonych spacjami.

Wyjście

Dla każdego zestawu danych program ma wypisać n linii postaci:

nazwa_ciągu - k pkt.

Gdzie ***nazwa_ciągu*** odpowiada nazwie ciągu podanego w wejściu a k oznacza liczbę punktów zdobytych w turnieju przez ten ciąg. Wyniki powinny być wypisywane zgodnie z kolejnością ciągów z wejścia.

Wymagania implementacyjne

Jedynym możliwym importem jest `java.util.Scanner`.

Dodatkowo przesłane rozwiązanie powinno spełniać wszystkie zasady zamieszczone w regulaminie zaliczania programów na BaCy.

Przykład

Wejście: 2 3 4 Batman 10 1 3 4 Listonosz 0 1 2 3 Klocuch 2 3 4 5 4 3 a1 1 0 1 a2 1 0 2 a3 1 1 -1 a4 1 0 3	Wyjście: Batman - 1 pkt. Listonosz - 0 pkt. Klocuch - 3 ptk. a1 - 0 pkt. a2 - 0 pkt. a3 - 1 pkt. a4 - 0 pkt.
--	--