

# C. Paliwo

---

Dostępna pamięć: 96 MB

Od miasta A do miasta B prowadzi tylko jedna droga o długości  $\ell$  kilometrów. Znany haker Limak mieszka w mieście A i postanowił odwiedzić swoją ciotkę mieszkającą w mieście B. Limak ma auto (nawiasem mówiąc straszliwego rzęcha), które potrafi przejechać na jednym baku odległość  $b$ .

W  $n$  miejscach przy drodze rozmieszczone są stacje benzynowe. Za tankowanie na stacji  $i$  płaci się konkretną ryczałtową kwotę  $c_i$ . Oznacza to, że jeśli Limak zatrzyma się, żeby zatankować na danej stacji  $i$ , to zawsze doleją mu paliwa do pełna, ale zapłaci  $c_i$  niezależnie od tego, ile miał jeszcze paliwa w baku.

Limak startuje z pełnym bakiem z miasta A i interesuje go, ile pieniędzy musi wydać na tankowania, żeby dojechać do miasta B.

## Specyfikacja danych wejściowych

W pierwszym wierszu danych wejściowych znajdują się trzy opisane wyżej liczby naturalne  $\ell \in [1, 10^9]$ ,  $n \in [0, 10^6]$  i  $b \in [1, 10^9]$  oddzielone pojedynczymi odstępami.

W każdym z  $n$  kolejnych wierszy danych wejściowych znajdują się dwie liczby naturalne  $d_i$  oraz  $c_i \in [0, 10^9]$  oddzielone pojedynczym odstępem, opisujące  $i$ -tą stację benzynową:  $d_i$  jest odległością w kilometrach od miasta A, zaś  $c_i$  kosztem tankowania. Stacje benzynowe są posortowane, żadne dwie stacje nie leżą w tym samym miejscu i nie ma stacji benzynowych w miastach A ani B. Innymi słowy jeśli  $1 \leq i < j \leq n$ , to  $0 < d_i < d_j < \ell$ .

## Specyfikacja danych wyjściowych

W pierwszym i jedynym wierszu wyjścia Twój program powinien wypisać jedną liczbę naturalną będącą minimalną kwotą, którą Limak musi wydać na tankowania. Jeśli przejazd jest niemożliwy, Twój program powinien wypisać liczbę  $-1$

### Przykład A

**Wejście:**

6 2 3  
1 2  
5 3

**Wyjście:**

-1

### Przykład B

**Wejście:**

8 3 4  
2 1  
4 2  
6 3

**Wyjście:**

2

### Przykład C

**Wejście:**

8 4 3  
1 1  
3 3  
4 1  
5 1

**Wyjście:**

3