

Szał zakupowy

Zdalne Warsztaty Olimpijskie dla Juniorów, drugi sparing
29 marca 2020

Kod zadania: **sza**
Limit czasu: **10 s**
Limit pamięci: **256 MB**



Bajtocy po comiesięcznych oględzinach mieszkania zauważył, że musi pojechać na zakupy. Zainteresowany ostatnimi obniżkami cen chciałby zaplanować dokładnie datę zakupów.

W galerii znajduje się N produktów. Każdy z produktów będzie miał własny okres czasu, w którym będzie na promocji, ale w Bitocji rabaty udzielane są w wyjątkowo oryginalny sposób. Produkt o numerze i będzie na promocji od dnia A_i do dnia B_i włącznie, przed rabatem kosztuje on P_i bitodolarów, a podczas rabatu jego cena obniża się stopniowo. Dokładniej, w pierwszym dniu rabatu (A_i) cena produktu jest niższa od pierwotnej o O_i , następnego dnia o $2 \cdot O_i$, aż do dnia B_i , kiedy jest najniższa i równa $P_i - (B_i - A_i + 1) \cdot O_i$. Po zakończeniu obowiązywania rabatu, czyli w dniu $B_i + 1$, produkt wraca do swojej pierwotnej ceny P_i . Promocyjne ceny są tak dobrane, żeby cena nie mogła w ten sposób spaść poniżej zera (niestety, nawet w Bitocji nikt nie dopłaca za kupienie jego towaru).

Kiedy Bajtocy wejdzie do galerii, wpadnie w szal zakupowy i kupi po jednej sztuce każdego produktu. Bajtocy jest jednak doświadczonym zakupoholikiem i wie dla każdego produktu, kiedy ten jest na przecenie, oraz zna parametry P_i oraz O_i . Bajtocy lubi też oszczędzać, więc chciałby wydać jak najmniej. Na zakupy może jednak pójść tylko raz – chciałby zatem wiedzieć, w którym dniu powinien to zrobić, aby kupując wszystkie produkty wydać możliwie najmniej. W przypadku kilku możliwości Bajtazar oczywiście wybierze najwcześniejszą. Pomóż mu!

Wejście

W pierwszym wierszu standardowego wejścia znajduje się jedna liczba całkowita N ($1 \leq N \leq 300\,000$) oznaczająca liczbę produktów. W kolejnych N wierszach znajdują się opisy każdego produktu. Każdy opis składa się z czterech liczb całkowitych A_i, B_i, P_i, O_i ($1 \leq A_i \leq B_i \leq 10^9, 1 \leq P_i, O_i \leq 10^9, P_i - (B_i - A_i + 1) \cdot O_i \geq 0$) oznaczające odpowiednio pierwszy i ostatni dzień przeceny, początkową cenę produktu, oraz spadek ceny podczas każdego dnia rabatu.

Wyjście

Twój program powinien wypisać dwie liczby całkowite oddzielone spacją, oznaczające odpowiednio numer dnia, w którym Bajtocy powinien iść na zakupy, aby wydać jak najmniej, oraz kwotę jaką wyda. Pamiętaj, że jeśli jest kilka możliwych dni z najmniejszym wydatkiem, musisz wypisać najwcześniejszy z nich.

Ocenianie

Możesz rozwiązać zadanie w kilku prostszych wariantach – niektóre grupy testów spełniają pewne dodatkowe ograniczenia. Poniższa tabela pokazuje, ile punktów otrzyma Twój program, jeśli przejdzie testy z takim ograniczeniem.

Dodatkowe ograniczenia	Liczba punktów
$N \leq 1000$ oraz dla każdego $i = 1, 2, \dots, N$: $B_i \leq 1000$	41
dla każdego $i = 1, 2, \dots, N$: $B_i \leq 10^6$	83

Przykłady

Wejście dla testu sza0a:

```
1
3 8 40 2
```

Wyjście dla testu sza0a:

```
8 28
```

Wyjaśnienie do przykładu: Cena jednego produktu będzie wynosiła w kolejnych dniach [40, 40, 38, 36, 34, 32, 30, 28, 40, 40, ...]. Naturalnie warto poczekać do ostatniego dnia promocji, czyli 8. dnia.



Wejście dla testu sza0b:

```
3
1 2 5 1
2 2 10 2
6 6 100 1
```

Wyjście dla testu sza0b:

```
2 111
```

Wyjaśnienie do przykładu: Bajtocy robiąc zakupy w drugim dniu wyda: $3 + 8 + 100 = 111$, jest to minimalna kwota jaką może wydać.

Wejście dla testu sza0c:

```
7
1 10 30 1
11 20 30 1
21 30 30 1
31 40 30 1
41 50 30 1
51 60 30 1
61 70 30 1
```

Wyjście dla testu sza0c:

```
10 200
```

Wejście dla testu sza0d:

```
5
1 1000 1000000000 1
2 2000 1000000000 2
3 3000 1000000000 3
4 4000 1000000000 4
5 5000 1000000000 5
```

Wyjście dla testu sza0d:

```
4000 4999964032
```

Pozostałe testy przykładowe

- test sza0e: $N = 300\,000$, a dla każdego przedmiotu $A_i = 1, B_i = 10^9, P_i = 10^9, O_i = 1$.