



Изпреварване

Между Летище Будапеща и Хотел Форас има еднолентов и еднопосочен път с дължина L километра.

През IOI 2023, $N + 1$ автобуса превозват участници по този път. Автобусите са номерирани с числата от 0 до N . По разписание автобус i ($0 \leq i < N$) тръгва от летището в секунда $T[i]$ и може да измине 1 километър за $W[i]$ секунди. Автобус N е резервен автобус, който може да измине 1 километър за X секунди. Времето Y , в което автобус N ще тръгне от летището, все още не е определено.

Не е позволено изпреварването на пътя, но автобусите могат да се изпреварват на специални **станции за сортиране**. Има M ($M > 1$) станции за сортиране по пътя, номерирани с числата от 0 до $M - 1$, и намиращи се на различни позиции. Станция за сортиране j ($0 \leq j < M$) се намира на километър $S[j]$ по пътя от летището. Станциите за сортиране са подредени в нарастващ ред спрямо разстоянието си от летището, тоест $S[j] < S[j + 1]$ за всяко $0 \leq j \leq M - 2$. Първата станция за сортиране е самото летище, а последната е хотелът - т.е. $S[0] = 0$ и $S[M - 1] = L$.

Всеки автобус пътува с максималната си скорост освен ако не е настигнал друг по-бавен автобус, пътуващ пред него. При такова настигане автобусите се движат заедно със скоростта на по-бавния от тях, докато не стигнат следващата станция за сортиране, където по-бързите автобуси изпреварват по-бавните.

Формално, за всяко i и j , такива че $0 \leq i \leq N$ и $0 \leq j < M$, времето $t_{i,j}$ (в секунди), в което автобус i **пристига на** станция за сортиране j е дефинирано по следния начин. Нека $t_{i,0} = T[i]$ за всяко $0 \leq i < N$, и нека $t_{N,0} = Y$. За всяко j , такова че $0 < j < M$:

- Дефинираме **очаквано време на пристигане** (в секунди) на автобус i в станция за сортиране j , означено като $e_{i,j}$, като времето, в което автобус i би пристигнал в станция за сортиране j ако се е движил с максималната си скорост от моментът, в който е пристигнал в станция за сортиране $j - 1$. Тоест, нека
 - $e_{i,j} = t_{i,j-1} + W[i] \cdot (S[j] - S[j - 1])$ за всяко $0 \leq i < N$, и
 - $e_{N,j} = t_{N,j-1} + X \cdot (S[j] - S[j - 1])$.
- Автобус i пристига в станция за сортиране j в **максималното** от очакваните времена на пристигане за автобус i и за всеки друг автобус, пристигнал в станция $j - 1$ по-рано от автобус i . Формално, нека $t_{i,j}$ е максимумът от $e_{i,j}$ и всяко $e_{k,j}$, за които $0 \leq k \leq N$ и $t_{k,j-1} < t_{i,j-1}$

Организаторите на IOI искат да изберат кога да тръгне резервният автобус (автобус N). Вашата задача е да отговорите на Q заявки от следния вид: за дадено време Y (в секунди), в което резервният автобус тръгва от летището, кога той ще пристигне в хотела?

Детайли по имплементацията

Трябва да имплементирате следната функция.

```
void init(int L, int N, int64[] T, int[] W, int X, int M, int[] S)
```

- L : дължината на пътя.
- N : броят нерезервни автобуси.
- T : масив с дължина N , описващ времената, в които нерезервните автобуси напускат летището.
- W : масив с дължина N , описващ максималните скорости на нерезервните автобуси.
- X : времето, за което резервният автобус изминава 1 километър.
- M : броят станции за сортиране.
- S : масив с дължина M , описващ разстоянията на станциите за сортиране от летището.
- Тази функция ще се извика точно веднъж за всеки тест, преди каквито и да е извиквания на `arrival_time`.

```
int64 arrival_time(int64 Y)
```

- Y : времето, в което резервният автобус (автобус N) ще тръгне от летището.
- Тази функция трябва да върне времето, в което резервният автобус ще пристигне в хотела.
- Тази функция ще бъде извикана точно Q пъти.

Пример

Да разгледаме следната поредица от извиквания:

```
init(6, 4, [20, 10, 40, 0], [5, 20, 20, 30], 10, 4, [0, 1, 3, 6])
```

Пренебрегвайки автобус 4 (чието разписание още не е фиксирано), следната таблица показва очакваните и истинските времена на пристигане за нерезервните автобуси за всяка станция за сортиране:

i	$t_{i,0}$	$e_{i,1}$	$t_{i,1}$	$e_{i,2}$	$t_{i,2}$	$e_{i,3}$	$t_{i,3}$
0	20	25	30	40	40	55	55
1	10	30	30	70	70	130	130
2	40	60	60	100	100	160	180
3	0	30	30	90	90	180	180

Времената на пристигане в станция 0 са времената, в които автобусите напускат летището. Тоест, $t_{i,0} = T[i]$ за $0 \leq i \leq 3$.

Очакваните и истинските времена за пристигане в станция 1 се пресмятат по следния начин:

- Очакваните времена на пристигане в станция 1:
 - Автобус 0: $e_{0,1} = t_{0,0} + W[0] \cdot (S[1] - S[0]) = 20 + 5 \cdot 1 = 25$.
 - Автобус 1: $e_{1,1} = t_{1,0} + W[1] \cdot (S[1] - S[0]) = 10 + 20 \cdot 1 = 30$.
 - Автобус 2: $e_{2,1} = t_{2,0} + W[2] \cdot (S[1] - S[0]) = 40 + 20 \cdot 1 = 60$.
 - Автобус 3: $e_{3,1} = t_{3,0} + W[3] \cdot (S[1] - S[0]) = 0 + 30 \cdot 1 = 30$.
- Истинските времена на пристигане в станция 1:
 - Автобуси 1 и 3 пристигат в станция 0 по-рано от автобус 0, така че $t_{0,1} = \max([e_{0,1}, e_{1,1}, e_{3,1}]) = 30$.
 - Автобус 3 пристига в станция 0 по-рано от автобус 1, така че $t_{1,1} = \max([e_{1,1}, e_{3,1}]) = 30$.
 - Автобус 0, автобус 1 и автобус 3 пристигат в станция 0 по-рано от автобус 2, така че $t_{2,1} = \max([e_{0,1}, e_{1,1}, e_{2,1}, e_{3,1}]) = 60$.
 - Никой автобус не пристига в станция 0 по-рано от автобус 3, така че $t_{3,1} = \max([e_{3,1}]) = 30$.

```
arrival_time(0)
```

На автобус 4 му отнема 10 секунди да измине 1 километър и в тази заявка е избрано да тръгне от летището в секунда 0. В такъв случай, следната таблица показва времената на пристигане за всеки автобус. Единствената промяна на очаквано или реално време на пристигане за нерезервен автобус е подчертана.

i	$t_{i,0}$	$e_{i,1}$	$t_{i,1}$	$e_{i,2}$	$t_{i,2}$	$e_{i,3}$	$t_{i,3}$
0	20	25	30	40	40	55	<u>60</u>
1	10	30	30	70	70	130	130
2	40	60	60	100	100	160	180
3	0	30	30	90	90	180	180
4	0	10	10	30	30	60	60

Виждаме, че автобус 4 пристига в хотела на 60-тата секунда. Съответно, функцията трябва да върне 60.

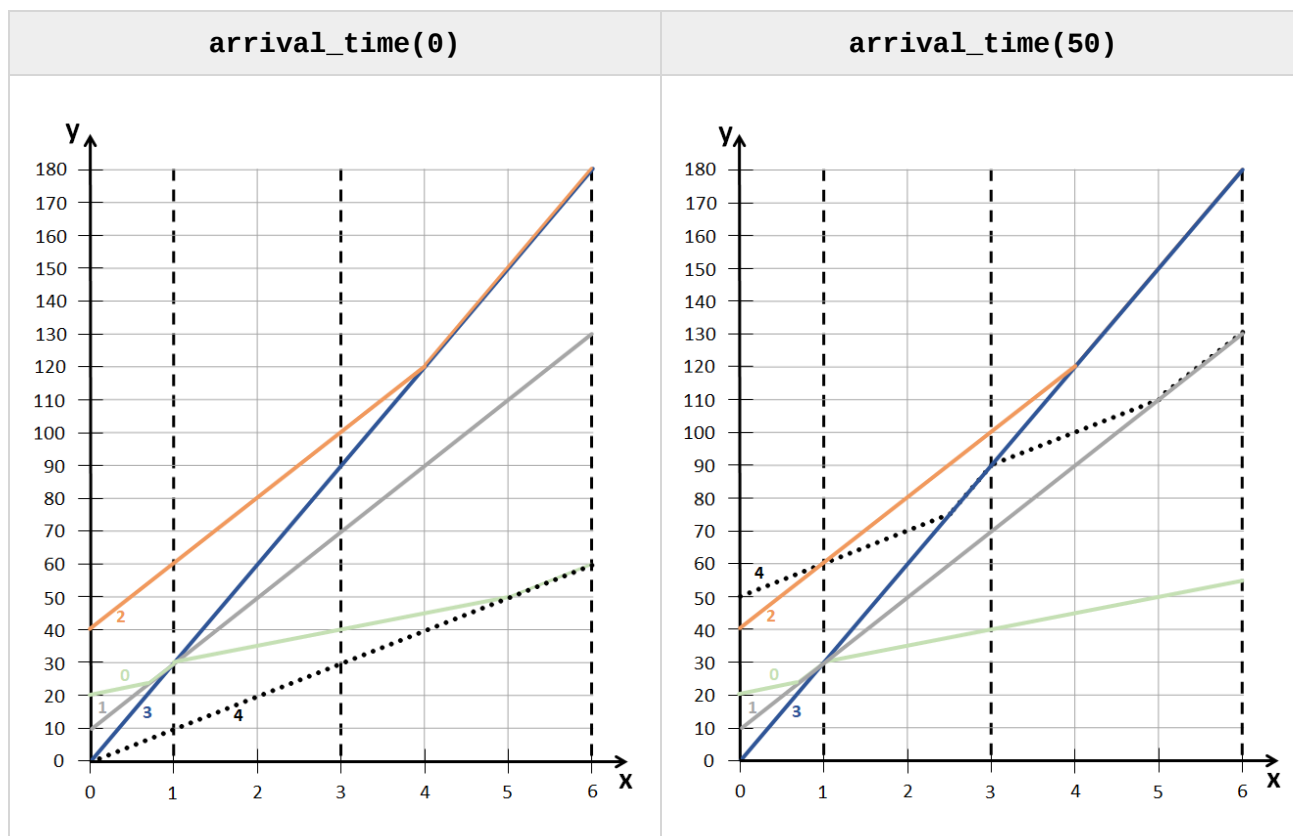
```
arrival_time(50)
```

В тази заявка автобус 4 тръгва от летището на 50-тата секунда. В такъв случай, няма промени във времената на пристигане на нерезервните автобуси спрямо първоначалната таблица. Времената на пристигане са показани в следната таблица.

i	$t_{i,0}$	$e_{i,1}$	$t_{i,1}$	$e_{i,2}$	$t_{i,2}$	$e_{i,3}$	$t_{i,3}$
0	20	25	30	40	40	55	55
1	10	30	30	70	70	130	130
2	40	60	60	100	100	160	180
3	0	30	30	90	90	180	180
4	50	60	60	80	90	120	130

Автобус 4 изпреварва по-бавния автобус 2 на станция за сортиране 1, тъй като пристигат по едно и също време. След това автобус 4 настига автобус 3 между станции 1 и 2 и се движи с него, съответно пристигайки в станция 2 на 90-тата секунда, вместо на 80-тата. След като напуска станция 2, автобус 4 настига автобус 1 и се движи с него, докато стигнат до хотела. Автобус 4 стига в хотела на 130-тата секунда. Съответно, функцията трябва да върне 130.

Можем да начертаем времето, което е нужно на всеки автобус да пристигне на всяко разстояние от летището. X-координатата на графиката обозначава разстоянието от летището (в километри), а Y-координатата обозначава времето (в секунди). Вертикални пунктирани линии обозначават позициите на станциите за сортиране. Различните непрекъснати линии (и индексите на автобусите към тях) обозначават четирите нерезервни автобуса. Черната пунктирана линия обозначава резервния автобус.



Ограничения

- $1 \leq L \leq 10^9$
- $1 \leq N \leq 1\,000$
- $0 \leq T[i] \leq 10^{18}$ (за всяко i , такова че $0 \leq i < N$)
- $1 \leq W[i] \leq 10^9$ (за всяко i , такова че $0 \leq i < N$)
- $1 \leq X \leq 10^9$
- $2 \leq M \leq 1\,000$
- $0 = S[0] < S[1] < \dots < S[M-1] = L$
- $1 \leq Q \leq 10^6$
- $0 \leq Y \leq 10^{18}$

Подзадачи

1. (9 точки) $N = 1, Q \leq 1\,000$
2. (10 точки) $M = 2, Q \leq 1\,000$
3. (20 точки) $N, M, Q \leq 100$
4. (26 точки) $Q \leq 5\,000$
5. (35 точки) Няма допълнителни ограничения.

Локално тестване

Локалният грейдър чете входа в следния формат:

- ред 1: $L \ N \ X \ M \ Q$
- ред 2: $T[0] \ T[1] \ \dots \ T[N - 1]$
- ред 3: $W[0] \ W[1] \ \dots \ W[N - 1]$
- ред 4: $S[0] \ S[1] \ \dots \ S[M - 1]$
- ред $5 + k$ ($0 \leq k < Q$): Y за заявка k

Локалният грейдър извежда отговорите Ви в следния формат:

- ред $1 + k$ ($0 \leq k < Q$): стойността върната от `arrival_time` за заявка k