Претекнување

Од аеродромот во Будимпешта до хотелот Forrás има еднонасочен пат со една лента. Патот е долг L километри.

Во текот на настанот IOI 2023, $N+1$ автобуси за трансфер го поминуваат овој пат. Автобусите се нумерирани со целите броеви од 0 до N. Автобусот i ($0 \leq i < N$) е предвидено (закажано) да го напушти аеродромот во $T[i]$-тата секунда од настанот, и може да помине 1 километар за $W[i]$ секунди. Автобусот N е резервен автобус кој може да помине 1 километар за X секунди. Времето Y кога тој ќе го напушти аеродромот не е одредено (закажано) сеуште.

Генерално не е дозволено претекнување на патот, но автобусите може да се претекнуваат меѓусебно во **станиците за сортирање**. Постојат M ($M > 1$) станици за сортирање, нумерирани со целите броеви од 0 до $M-1$, поставени на различни позиции на патот. Станицата за сортирање j ($0 \leq j < M$) е лоцирана на $S[j]$ километри од аеродромот долж патот. Станиците за сортирање се подредени во растечки редослед според растојанието од аеродромот, т.е. $S[j] < S[j+1]$ за секое $0 \leq j \leq M-2$. Прва станица за сортирање е аеродромот, а последната е хотелот, т.е. $S[0] = 0$ и $S[M-1] = L$.

Секој автобус се движи со максимална брзина, освен ако не наиде на побавен автобус пред него на патот, во кој случај тие продолжуваат да се движат како група и се принудени да патуваат со брзината на побавниот автобус, се' дури не стигнат до следната станица за сортирање. Таму, побрзите автобуси ќе ги претекнат побавните автобуси.

Формално, за секои i и j така што $0 \leq i \leq N$ и $0 \leq j < M$, времето $t_{i,j}$ (во секунди) кога автобусот i **пристигнува во** станицата за сортирање j се дефинира на следниот начин: Нека $t_{i,0} = T[i]$ за секое $0 \leq i < N$, и нека $t_{N,0} = Y$. За секое j такво што $0 < j < M$:

- Се дефинира **очекувано време на пристигнување** (во секунди) на автобусот i во станицата за сортирање j, кое се означува со $e_{i,j}$, како времето кога автобусот i би пристигнал во станицата за сортирање j ако истиот се движел со полна брзина од времето кога пристигнал во станицата за сортирање $j-1$. Односно, нека
    - $e_{i,j} = t_{i,j-1} + W[i] \cdot (S[j] - S[j-1])$ за секое $0 \leq i < N$, и
    - $e_{N,j} = t_{N,j-1} + X \cdot (S[j] - S[j-1])$.
- Автобусот i пристигнува во станицата за сортирање j во *максимумот* од очекуваните времиња на пристигнувања на автобусот i и на секој друг автобус што пристигнал во

станицата $j-1$ пред автобусот i. Формално, нека $t_{i,j}$ е максимумот од $e_{i,j}$ и од секое $e_{k,j}$ за коешто $0 \leq k \leq N$ и $t_{k,j-1} < t_{i,j-1}$.

Организаторите на IOI сакаат да го закажат резервниот автобус (автобусот N). Ваша задача е да одговорите на Q прашања од организаторите, кои се од следниот облик: ако е дадено времето Y (во секунди) кога резервниот автобус би требало да го напушти аеродромот, во кое време истиот би пристигнал кај хотелот?

Имплементациски детали

Ваша задача е да ги имплементирате следните процедури:

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

- L: должината на патот.
- N: бројот на нерезервни автобуси.
- T: низа со должина N што ги опишува времињата во кои е предвидено (закажано) нерезервните автобуси да го напуштат аеродромот.
- W: низа со должина N што ги опишува максималните брзини на движење на нерезервните автобуси.
- X: времето што му е потребно на резервниот автобус да помине 1 километар.
- M: бројот на станици за сортирање.
- S: низа со должина M што ги опишува растојанијата на секоја од станиците за сортирање од аеродромот.
- Оваа процедура се повикува точно еднаш за секој тест случај, пред кои било повици до процедурата `arrival_time`.

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

- Y: времето во кое резервниот автобус (автобусот N) треба да го напушти аеродромот.
- Оваа процедура треба да го врати времето во кое резервниот автобус би пристигнал кај хотелот.
- Оваа процедура се повикува точно Q пати.

Пример

Да ја разгледаме следната секвенца од повици:

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

Игнорирајќи го автобусот 4 (кој сеуште не е закажан), следната табела ги прикажува очекуваните и вистинските времиња на пристигнувања за нерезервните автобуси во секоја од станиците за сортирање:

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

Времињата на пристигнувања во станицата 0 се времињата во кои автобусите се закажани да го напуштат аеродромот. Односно, $t_{i,0} = T[i]$ за $0 \leq i \leq 3$.

Очекуваните и вистинските времиња за пристигнувања во станицата за сортирање 1 се пресметуваат на следниот начин:

- Очекуваните времиња на пристигнувања во станицата 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: $e_{3,1} = t_{3,0} + W[3] \cdot (S[1] - S[0]) = 0 + 30 \cdot 1 = 30$.
- Времињата на пристигнувања во станицата 1:
  - Автобусите 1 и 3 пристигнуваат во станицата 0 пред автобусот 0, па $t_{0,1} = \max([e_{0,1}, e_{1,1}, e_{3,1}]) = 30$.
  - Автобусот 3 пристигнува во станицата 0 пред автобусот 1, па $t_{1,1} = \max([e_{1,1}, e_{3,1}]) = 30$.
  - Автобусот 0, автобусот 1 и автобусот 3 пристигнуваат во станицата 0 пред автобусот 2, па $t_{2,1} = \max([e_{0,1}, e_{1,1}, e_{2,1}, e_{3,1}]) = 60$.
  - Ниту еден автобус не пристигнува во станицата 0 пред автобусот 3, па $t_{3,1} = \max([e_{3,1}]) = 30$.

```
arrival_time(0)
```

На автобусот 4 му требаат 10 секунди за да помине 1 километар и сега е закажан да го напушти аеродромот во 0-тата секунда. Во овој случај, следната табела ги прикажува времињата на пристигнувања за секој автобус. Единствената промена во однос на очекуваните и вистинските времиња на пристигнување на нерезервните автобуси е означена со подвлечено (анг. underlined).

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	$\underline{60}$
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

Гледаме дека автобусот 4 пристигнува кај хотелот во 60-тата секунда. Според тоа, процедурата треба да врати 60.

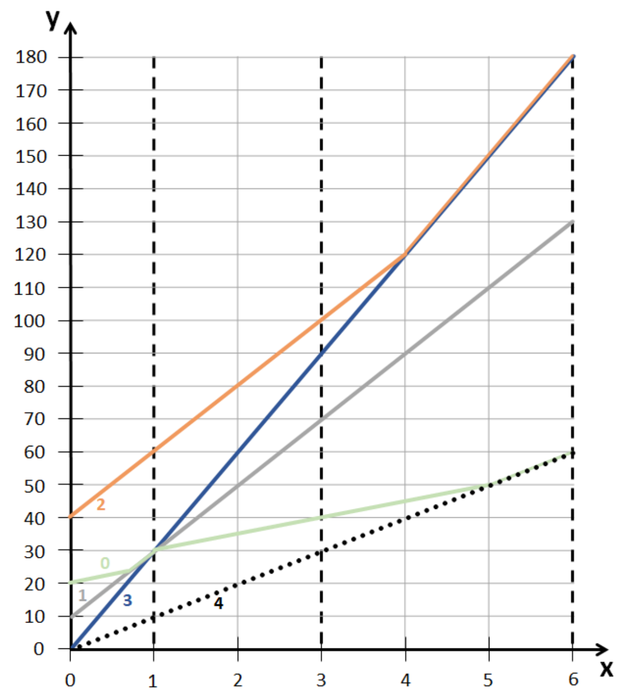
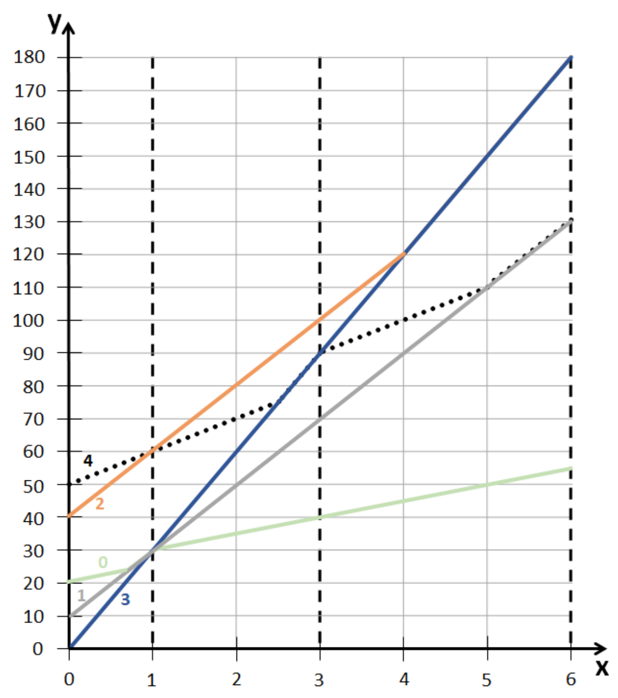
```
arrival_time(50)
```

Автобусот 4 сега е закажан да го напушти аеродромот во 50-тата секунда. Во овој случај, нема промени во времињата на пристигнувања за нерезервните автобуси во однос на почетната табела. Времињата на пристигнувања се прикажани во следната табела.

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

Автобусот 4 го претекнува побавниот автобус 2 во станицата за сортирање 1 бидејќи тие пристигнуваат таму во исто време. Следно, автобусот 4 влегува во група со автобусот 3 помеѓу станицата 1 и станицата 2, што предизвикува автобусот 4 да пристигне во станицата 2 во 90-тата секунда, наместо во 80-тата. Откако ќе ја напушти станицата 2, автобусот 4 влегува во група со автобусот 1 се' додека тие не пристигнат кај хотелот. Автобусот 4 пристигнува кај хотелот во 130-тата секунда. Според тоа, процедурата треба да врати 130.

Можеме графички да го претставиме времето што му е потребно на секој автобус за да пристигне на секое растојание од аеродромот. X-оската на овој график го претставува растојанието од аеродромот (во километри), а y-оската го претставува времето (во секунди). Вертикалните испрекинати линии ги означуваат позициите на станиците за сортирање. Различните полни линии (придружени со индексите на автобусите) ги претставуваат четирите нерезервни автобуси. Точкестата црна линија го претставува резервниот автобус.

`arrival_time(0)`	`arrival_time(50)`
	

Ограничувања

- $1 \le L \le 10^9$
- $1 \le N \le 1\,000$
- $0 \le T[i] \le 10^{18}$ (за секое i такво што $0 \le i < N$)
- $1 \le W[i] \le 10^9$ (за секое i такво што $0 \le i < N$)
- $1 \le X \le 10^9$
- $2 \le M \le 1\,000$
- $0 = S[0] < S[1] < \cdots < S[M-1] = L$
- $1 \le Q \le 10^6$
- $0 \le Y \le 10^{18}$

Подзадачи

1. (9 поени) $N = 1, Q \le 1\,000$
2. (10 поени) $M = 2, Q \le 1\,000$
3. (20 поени) $N, M, Q \le 100$
4. (26 поени) $Q \le 5\,000$
5. (35 поени) Без дополнителни ограничувања.

Пример-оценувач

Пример-оценувачот го чита влезот во следниот формат:

- линија 1: $L\ N\ X\ M\ Q$
- линија 2: $T[0]\ T[1]\ \ldots\ T[N-1]$
- линија 3: $W[0]\ W[1]\ \ldots\ W[N-1]$
- линија 4: $S[0]\ S[1]\ \ldots\ S[M-1]$
- линија $5+k$ ($0 \le k < Q$): Y за прашањето k

Пример-оценувачот ги печати вашите одговори во следниот формат:

- линија $1+k$ ($0 \le k < Q$): повратната вредност на процедурата `arrival_time` за прашањето k