Apsteigšana

Starp Budapeštas lidostu un Forrás viesnīcu ir vienjoslas vienvirziena ceļš. Ceļš ir L kilometru garš.

IOI 2023 pasākuma laikā $N+1$ autobusi brauks pa šo ceļu. Autobusi ir sanumurēti no 0 līdz N. Autobuss ar numuru i ($0 \le i < N$) izbrauks no lidostas pasākuma $T[i]$-tajā sekundē, un var nobraukt 1 kilometru $W[i]$ sekundēs. Autobuss ar numuru N ir rezerves autobuss, kurš var nobraukt 1 kilometru X sekundēs. Laiks Y, kad tas izbrauks no lidostas, vēl nav izlemts.

Vispārīgi, apsteigšana uz ceļa nav atļauta, bet autobusiem ir atļauts apsteigt vienam otru **šķirošanas stacijās**. Uz ceļa ir M ($M > 1$) šķirošanas stacijas dažādās pozīcijās, kas sanumurētas no 0 līdz $M-1$. Šķirošanas stacija ar numuru j ($0 \le j < M$) atrodas uz ceļa $S[j]$ kilometru attālumā no lidostas. Šķirošanas stacijas ir numurētas secībā pēc augoša attāluma no lidostas, tas ir, $S[j] < S[j+1]$ katram $0 \le j \le M-2$. Pirmā šķirošanas stacija ir lidosta un pēdējā ir viesnīca, tas ir, $S[0] = 0$ un $S[M-1] = L$.

Katrs autobuss ceļo ar maksimālu ātrumu, izņemot, ja tas panāk lēnāku autobusu sev priekšā, kurā gadījumā tie izveido kolonnu un abi ceļo ar lēnākā autobusa ātrumu, kamēr tie nesasniedz nākāmo šķirošanas staciju. Tur ātrāki autobusi apsteidz lēnākus autobusus.

Precīzāk, katram i un j, kur $0 \le i \le N$ un $0 \le j < M$, laiks $t_{i,j}$ (sekundēs), kad autobuss i **atbrauks** šķirošanas stacijā j, ir definēts šādi. Lai $t_{i,0} = T[i]$ katram $0 \le i < N$ un lai $t_{N,0} = Y$. Katram j, kur $0 < j < M$:

- Definēsim autobusa i **sagaidāmo atbraukšanas laiku** (sekundēs) šķirošanas stacijā j, un apzīmēsim ar $e_{i,j}$, kā laiku, kad autobuss i atbrauks šķirošanas stacijā j, ja tas ceļotu ar maksimālo ātrumu no momenta, kad tas atbrauca šķirošanas stacijā $j-1$.
    - $e_{i,j} = t_{i,j-1} + W[i] \cdot (S[j] - S[j-1])$ katram $0 \le i < N$, un
    - $e_{N,j} = t_{N,j-1} + X \cdot (S[j] - S[j-1])$.
- Autobuss i atbrauc šķirošanas stacijā j *maksimālā* no autobusa i un katra cita autobusa, kas atbrauca šķirošanas stacijā $j-1$ āgrāk, nekā autobuss i, sagaidāmajiem atbraukšanas laikiem. Precīzāk, lai $t_{i,j}$ ir maksimums no $e_{i,j}$ un visiem $e_{k,j}$, kuriem $0 \le k \le N$ un $t_{k,j-1} < t_{i,j-1}$.

IOI organizatori grib ieplānot rezerves autobusu N. Jūsu uzdevums ir atbildēt uz organizatoru Q jautājumiem, kas ir šādā formā: ja ir dots laiks Y (sekundēs), kad rezerves autobuss izbrauks no lidostas, kāds ir laiks, kad tas atbrauks uz viesnīcu?

Implementācijas detaļas

Jums ir jāimplementē šādas procedūras:

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

- L: ceļa garums.
- N: nerezerves autobusu skaits.
- T: masīvs garumā N, kas apraksta laikus, kuros nerezerves autobusi izbrauks no lidostas.
- W: masīvs garumā N, kas apraksta nerezerves autobusu maksimālus ātrumus.
- X: laiks, kurā rezerves autobuss nobrauc 1 kilometru.
- M: šķirošanas staciju skaits.
- S: masīvs garumā M, kas apraksta šķirošanas staciju attālumus no lidostas.
- Šī procedūra tiks izsaukta vienu reizi katrā testā, pirms jebkuriem `arrival_time` izsaukumiem.

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

- Y: laiks, kad rezerves autobuss N izbrauks no lidostas.
- Šai procedūrai ir jāatgriež laiks, kad rezerves autobuss atbrauks viesnīcā.
- Šī procedūra tiks izsaukta tieši Q reizes.

Piemērs

Apskatīsim šādu procedūru izsaukumu virkni:

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

Ignorējot 4. autobusu (kura izbraukšanas laiks vēl nav izlemts), tabula zemāk parāda sagaidāmo un faktisko atbraukšanas laiku nerezerves autobusiem katrā šķirošanas stacijā:

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

Atbraukšanas laiki 0. šķirošanas stacijā ir laiki, kuros autobusi izbrauks no lidostas. Tas ir, $t_{i,0} = T[i]$ visiem $0 \le i \le 3$.

Sagaidāmie un faktiskie atbraukšanas laiki 1. šķirošanas stacijā tiek aprēķināti šādi:

- Sagaidāmie atbraukšanas laiki 1. šķirošanas stacijā:
  - 0. autobuss: $e_{0,1} = t_{0,0} + W[0] \cdot (S[1] - S[0]) = 20 + 5 \cdot 1 = 25$.
  - 1. autobuss: $e_{1,1} = t_{1,0} + W[1] \cdot (S[1] - S[0]) = 10 + 20 \cdot 1 = 30$.
  - 2. autobuss: $e_{2,1} = t_{2,0} + W[2] \cdot (S[1] - S[0]) = 40 + 20 \cdot 1 = 60$.
  - 3. autobuss: $e_{3,1} = t_{3,0} + W[3] \cdot (S[1] - S[0]) = 0 + 30 \cdot 1 = 30$.
- Faktiskie atbraukšanas laiki 1. šķirošanas stacijā:
  - 1. un 3. autobusi atbrauc 0. šķirošanas stacijā āgrāk par 0. autobusu, līdz ar to $t_{0,1} = \max([e_{0,1}, e_{1,1}, e_{3,1}]) = 30$.
  - 3. autobuss atbrauc 0. šķirošanas stacijā āgrāk par 1. autobusu, līdz ar to $t_{1,1} = \max([e_{1,1}, e_{3,1}]) = 30$.
  - 0. autobuss, 1. autobuss un 3. autobuss atbrauc 0. šķirošanas stacijā āgrāk par 2. autobusu, līdz ar to $t_{2,1} = \max([e_{0,1}, e_{1,1}, e_{2,1}, e_{3,1}]) = 60$.
  - Nav autobusu, kas atbrauc 0. šķirošanas stacijā āgrāk par 3. autobusu, līdz ar to $t_{3,1} = \max([e_{3,1}]) = 30$.

```
arrival_time(0)
```

4. autobuss nobrauc 1 kilometru 10 sekundēs un tagad izbrauks no lidostas 0. sekundē. Šajā gadījumā tabula zemāk parāda autobusu sagaidāmo un faktisko atbraukšanas laikus katrā šķirošanas stacijā. Vienīgā izmaiņa nerezerves autobusu sagaidāmajiem un faktiskajiem atbraukšanas laikiem ir pasvītrota.

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

Var redzēt, kā 4. autobuss atbrauc viesnīcā 60. sekundē. Tātad, procedūrai ir jāatgriež 60.

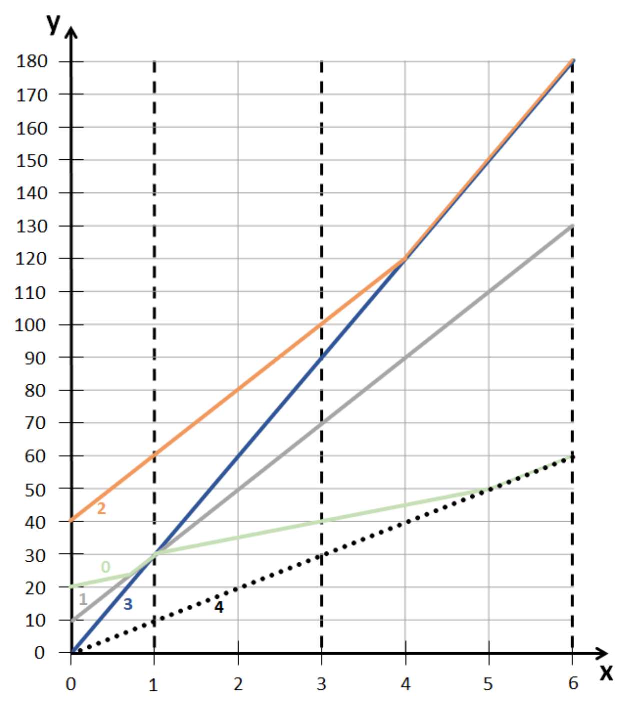
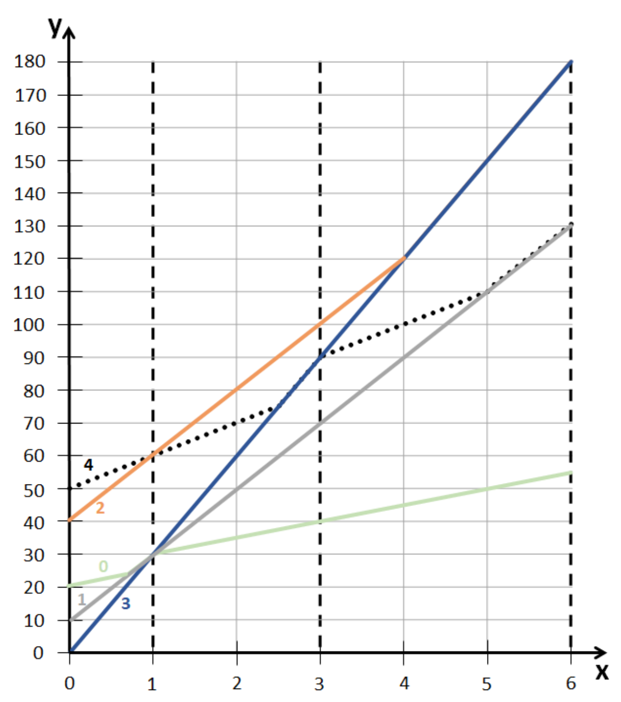
```
arrival_time(50)
```

4. autobuss tagad izbrauks no lidostas 50. sekundē. Šajā gadījumā, nav izmaiņu nerezerves autobusu sagaidāmajiem un faktiskajiem atbraukšanas laikiem, salīdzinot ar sākotnējo tabulu. Nākamā tabula parāda sagaidāmo un faktisko atbraukšanas laikus nerezerves autobusiem katrā šķirošanas stacijā:

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

4. autobuss apsteidz 2. autobusu, kas ir lēnaks, 1. šķirošanas stacijā, jo tie atbrauc vienlaicīgi. Pēc tam 4. autobuss nokļūst kolonnā ar 3. autobusu starp 1. un 2. šķirošanas stacijām, kas liek 4. autobusam atbraukt 2. šķirošanas stacijā 90. sekundē (nevis 80.). Pēc aizbraukšanas no 2. šķirošanas stacijas, 4. autobuss nokļūst kolonnā ar 1. autobusu līdz tie atbrauc viesnīcā. 4. autobuss atbrauc viesnīcā 130. sekundē. Tātad, procedūrai ir jāatgriež 130.

Mēs varam uzzīmēt grafiku laikam, kurā katrs autobuss atbrauks katrā attālumā no lidostas. Grafika x-ass apzīmē attālumu no lidostas (kilometros) un grafika y-ass apzīmē laiku (sekundēs). Vertikālas pārtrauktās līnijas apzīmē šķirošanas staciju pozīcijas. Dažādas nepārtrauktās taisnes (kopā ar autobusu numuriem) apzīmē četrus nerezerves autobusus. Melnā punktētā taisne apzīmē rezerves autobusu.

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

Ierobežojumi

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

Apakšuzdevumi

1. (9 punkti) $N = 1, Q \le 1\,000$
2. (10 punkti) $M = 2, Q \le 1\,000$
3. (20 punkti) $N, M, Q \le 100$
4. (26 punkti) $Q \le 5\,000$
5. (35 punkti) Bez papildus ierobežojumiem.

Piemēra vērtētājprogramma

Piemēra vērtētājprogramma ielasa ievaddatus šādā formātā:

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

Piemēra vērtētājprogramma izvada jūsu atbildes šādā formātā:

- $(1+k)$-tā ($0 \le k < Q$) rinda: procedūras `arrival_time` atgrieztā vērtība jautājumam k