سبقت گرفتن

یک جاده یکطرفه و تک لاین از فرودگاه بوداپست به هتل فورس وجود دارد. طول این جاده L کیلومتر است.

در زمان برگزاری رویداد آی‌اوآی ۲۰۲۳، $N+1$ اتوبوس این جاده را طی می‌کنند. اتوبوس‌ها از 0 تا N شماره‌گذاری شده‌اند. اتوبوس iام ($0 \le i < N$) قرار است در زمان $T[i]$ام این رویداد فرودگاه را به مقصد هتل ترک کند. این اتوبوس می‌تواند یک کیلومتر را در $W[i]$ ثانیه طی کند. اتوبوس Nام، اتوبوس ذخیره است که می‌تواند یک کیلومتر را در X ثانیه طی کند. زمان حرکت این اتوبوس از فرودگاه که با Y نمایش داده می‌شود فعلا نامشخص است.

در حالت کلی سبقت گرفتن در جاده امکان‌پذیر نیست، اما اتوبوس‌ها می‌توانند در یکسری ایستگاه که به نام ایستگاه مرتب‌سازی می‌شناسیم از یکدیگر سبقت بگیرند. در جاده M ایستگاه مرتب‌سازی وجود دارد که با شماره‌های 0 تا $M-1$ شماره‌گذاری شده‌اند. این ایستگاه‌ها متمایز هستند. ایستگاه مرتب‌سازی jام در فاصله $S[j]$ کیلومتری از فرودگاه قرار گرفته است. ایستگاه‌ها به ترتیب فاصله از فرودگاه شماره‌گذاری شده‌اند، یعنی $S[j] < S[j+1]$ برای هر $0 \le j \le M-2$. اولین ایستگاه مرتب‌سازی فرودگاه و اخرین ایستگاه مرتب‌سازی هتل است. در واقع $S[0]=0$ و $S[M-1]=L$ است.

هر اتوبوس با ماکزیمم سرعت خود حرکت می‌کند مگر انکه به اتوبوسی برسد که سرعتش کمتر است. در این حال هر دو با سرعت اتوبوس کندتر حرکت می‌کنند تا اینکه به ایستگاه مرتب‌سازی بعدی برسند. در ایستگاه اتوبوس‌های با سرعت بیشتر از اتوبوس‌های با سرعت کم‌تر سبقت می‌گیرند.

بطور رسمی، برای هر i و j که $0 \le i \le N$ و $0 \le j < M$ زمان $t_{i,j}$ (برحسب ثانیه) که زمان رسیدن اتوبوس iام به ایستگاه مرتب‌سازی jام است بدین شکل تعریف می‌شود. فرض کنید $t_{i,0} = T[i]$ برای هر $0 \le i < N$ و فرض کنید $t_{N,0} = Y$. برای هر j که $0 < j < M$:

- زمان رسیدن اتوبوس iام به ایستگاه jام که با $e_{i,j}$ نمایش داده می‌شود برابر زمان رسیدن اتوبوس iام به ایستگاه j می‌باشد اگر از زمانی که به ایستگاه $j-1$ ام می‌رسد با ماکزیمم سرعت خود حرکت می‌کرد. به عبارت دیگر
    - $e_{i,j} = t_{i,j-1} + W[i] \cdot (S[j] - S[j-1])$ برای هر $0 \le i < N$ و
    - $e_{N,j} = t_{N,j-1} + X \cdot (S[j] - S[j-1])$
- زمان رسیدن اتوبوس iام به ایستگاه jام برابر با ماکزیمم زمان رسیدن اتوبوس iام و هر اتوبوسی دیگری است که زودتر از اتوبوس iام به ایستگاه $j-1$ برسد. بطور رسمی، فرض کنید $t_{i,j}$ برابر ماکزیمم $e_{i,j}$ و هر $e_{k,j}$ است که برای آن $0 \le k \le N$ و $t_{k,j-1} < t_{i,j-1}$.

کمیته اجرایی آی‌اوآی می‌خواهد اتوبوس ذخیره (اتوبوس Nام) را زمانبندی کند. وظیفه شما این است که به Q پرسش کمیته اجرایی که به فرمت زیر است پاسخ دهید: زمان Y (برحسب ثانیه)، زمان ترک کردن اتوبوس Nام از فرودگاه، داده می‌شود، زمان رسیدن این اتوبوس به هتل را گزارش کنید.

Implementation Details

.Your task is to implement the following procedures

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

the length of the road. * N: the number of scheduled buses. * T: an array of length N :L * describing the times at which non-reserve buses are scheduled to leave from the airport. * W: an array of length N describing the maximum speeds of non-reserve buses. * X: the time it takes for the reserve bus to travel 1 kilometre. * M: the number of sorting stations. * S: an array of length M describing the distances of the sorting stations from the airport. * This procedure is called .exactly once for each test case, before any calls to `arrival_time`

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

the time at which the reserve bus is supposed to leave from the airport. * This procedure :Y * should return the time at which bus N would arrive at the hotel. * This procedure is called exactly Q .times

Example

:Consider the following sequence of calls

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

Ignoring bus 4 (that has not yet been scheduled), the following table shows the expected and :actual times of arrivals for non-reserve buses at each sorting station

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

The times of arrivals at station 0 are the times at which buses are scheduled to leave the airport. .$0 \le i \le 3$ That is, $t_{i,0} = T[i]$ for

The expected and actual times of arrivals at sorting station 1 are computed as follows: * The :1 expected times of arrivals at station

- .$e_{0,1} = t_{0,0} + W[0] \cdot (S[1] - S[0]) = 20 + 5 \cdot 1 = 25$:0 Bus
- .$e_{1,1} = t_{1,0} + W[1] \cdot (S[1] - S[0]) = 10 + 20 \cdot 1 = 30$:1 Bus

- $.e_{2,1} = t_{2,0} + W[2] \cdot (S[1] - S[0]) = 40 + 20 \cdot 1 = 60$:2 Bus
- 1 Bus 3: $e_{3,1} = t_{3,0} + W[3] \cdot (S[1] - S[0]) = 0 + 30 \cdot 1 = 30$. * The times of arrivals at station
:
- $.t_{0,1} = \max([e_{0,1}, e_{1,1}, e_{3,1}]) = 30$ Buses 1 and 3 arrive at station 0 earlier than bus 0, so
- $.t_{1,1} = \max([e_{1,1}, e_{3,1}]) = 30$ Bus 3 arrives at station 0 earlier than bus 1, so
- Bus 0, bus 1 and bus 3 arrive at sorting station 0 earlier than bus 2, so $.t_{2,1} = \max([e_{0,1}, e_{1,1}, e_{2,1}, e_{3,1}]) = 60$
- $.t_{3,1} = \max([e_{3,1}]) = 30$ No bus arrives at station 0 earlier than bus 3, so

```
arrival_time(0)
```

Bus 4 takes 10 seconds to travel 1 kilometre and is now scheduled to leave the airport at the 0-th second. In this case, the following table shows the times of arrivals for each bus. The only change .regarding the expected and actual arrival times of the non-reserve buses is 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

.60 We see that bus 4 arrives at the hotel at the 60-th second. Thus, the procedure should return

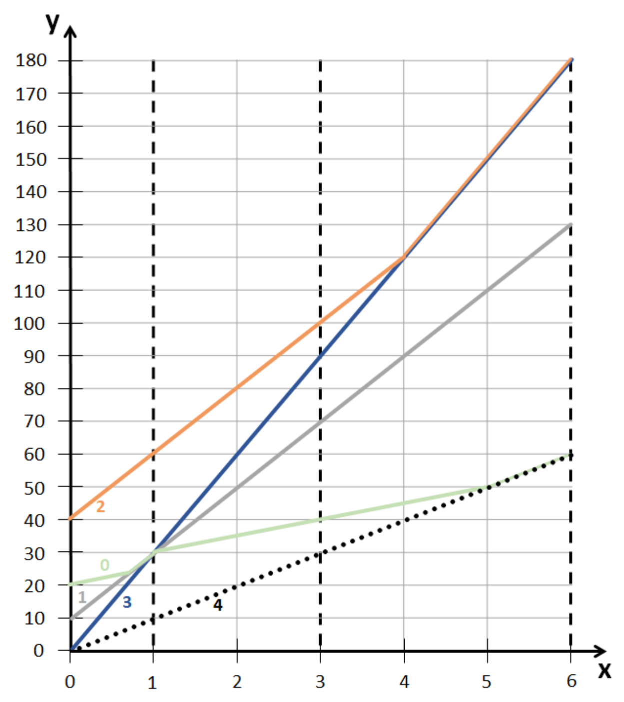
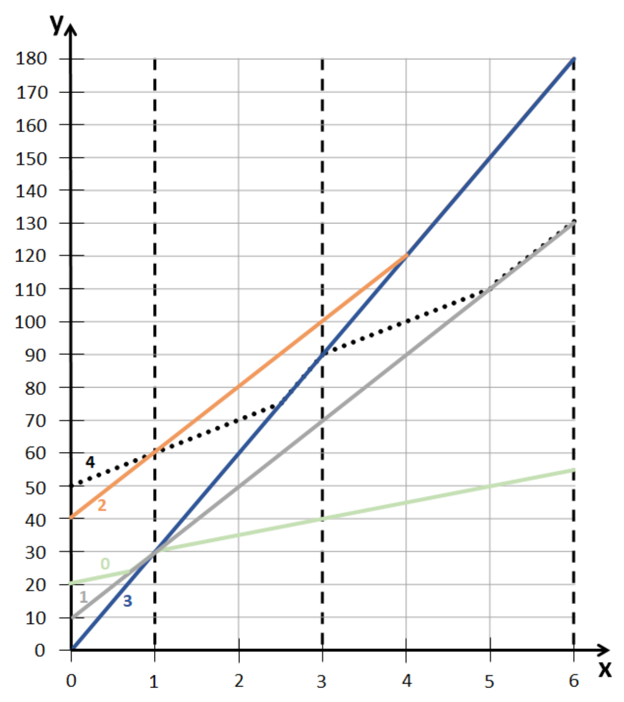
```
arrival_time(50)
```

Bus 4 is now scheduled to leave the airport at the 50-th second. In this case, there are no changes in the times of arrivals for the non-reserve buses compared to the initial table. The times of .arrivals are shown in the following table

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

Bus 4 overtakes the slower bus 2 at sorting station 1 as they arrive at the same time. Next, bus 4 gets bunched with bus 3 between station 1 and station 2, making bus 4 arrive at station 2 at the 90-th second instead of the 80-th. After leaving station 2, bus 4 gets bunched with bus 1 up until they arrive at the hotel. Bus 4 arrives at the hotel at the 130-th second. Thus, the procedure should .130 return

We can plot the time it takes for each bus to arrive at each distance from the airport. The x-axis of the plot represents the distance from the airport (in kilometres) and the y-axis of the plot represents the time (in seconds). Vertical broken lines mark the positions of the sorting stations. Different solid lines (accompanied by the bus indices) represent the four scheduled non-reserve .buses. The dotted black line represents the reserve bus

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

Constraints

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

Subtasks

$N = 1, Q \le 1\,000$ (points 9) .1
$M = 2, Q \le 1\,000$ (points 10) .2
$N, M, Q \le 100$ (points 20) .3

$Q \leq 5\,000$ (points 26) .4

.points) No additional constraints 35) .5

Sample Grader

:The sample grader reads the input in the following format

line 1: $L\ N\ X\ M\ Q$ * line 2: $T[0]\ T[1]\ \ldots\ T[N-1]$ * line 3: $W[0]\ W[1]\ \ldots\ W[N-1]$ * line 4: *
k $S[0]\ S[1]\ \ldots\ S[M-1]$ * line $5+k$ $(0 \leq k < Q)$: Y for question

:The sample grader prints your answers in the following format

k line $1+k$ $(0 \leq k < Q)$: the return value of `arrival_time` for question *