



Adelantamiento

Hay una carretera de un solo sentido desde el aeropuerto de Budapest hasta el Hotel Forrás. el camino es L kilómetros de largo.

Durante el evento IOI 2023, $N + 1$ los autobuses de transferencia atraviesan esta carretera. Los autobuses están numerados desde 0 to N . Autobús i ($0 \leq i < N$) tiene previsto abandonar el aeropuerto a las $T[i]$ -ésimo segundo del evento, y puede viajar 1 kilómetro en $W[i]$ segundos. Autobús N es un autobús de reserva que puede viajar 1 kilómetro en X segundos. El tiempo Y aún no se ha decidido cuándo saldrá del aeropuerto.

En general, no está permitido adelantar en la carretera, pero los autobuses pueden adelantarse entre sí en las **estaciones de clasificación**. Hay M estaciones de clasificación en la carretera, numeradas desde 0 hasta $M - 1$. Estación de clasificación j ($0 \leq j < M$) se encuentra $S[j]$ kilómetros del aeropuerto por la carretera. La primera estación de clasificación es el aeropuerto y la última es el hotel, es decir, $S[0] = 0$ and $S[M - 1] = L$.

Cada autobús viaja a máxima velocidad a menos que alcance a un autobús más lento que viaja delante de él en la carretera, en cuyo caso se amontonan y se ven obligados a viajar a la misma velocidad más baja, hasta llegar a la siguiente estación de clasificación. Formalmente, para cada i y j al que $0 \leq i \leq N$ y $0 \leq j < M$, el tiempo $t_{i,j}$ (en segundos) cuando el autobús i **llega** estación e clasificación j se define de la siguiente manera.

Si $j = 0$, entonces sea $t_{i,0} = T[i]$ para cada $i < N$, and sea $t_{N,0} = Y$. De lo contrario, para cada j tal que $0 < j < M$:

- Definir la *hora prevista de llegada* del autobús i en la estación de clasificación j como el momento en que el autobús i llegaría a la estación de clasificación j si viajaba a toda velocidad desde el momento en que llegó a la estación de clasificación $j - 1$. Es decir, deja
 - $e_{i,j} = t_{i,j-1} + W[i] \cdot (S[j] - S[j - 1])$ para cada $i < N$, and
 - $e_{N,j} = t_{N,j-1} + X \cdot (S[j] - S[j - 1])$.
- Autobús i llega a la estación de clasificación j al *máximo* de los tiempos previstos de llegada de autobús i y de todos los demás autobuses que llegaron a la estación $j - 1$ antes que el autobús i . Formalmente, dejemos $t_{i,j}$ ser el máximo de $e_{i,j}$ y cada $e_{k,j}$ para cual $0 \leq k \leq N$ y $t_{k,j-1} < t_{i,j-1}$.

Los organizadores del IOI quieren programar el autobús de traslado de reserva. Tu tarea es responder Q preguntas de los organizadores, que son de la siguiente forma:: dado el tiempo Y (en

segundos) cuando el autobús N se supone que sale del aeropuerto, ¿a qué hora llegaría al hotel?

Detalles de implementación

Su tarea es implementar los siguientes procedimientos.

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

- L : la longitud del camino.
- N : el número de autobuses de transferencia programados.
- T : una matriz de longitud N describiendo los horarios en los que los autobuses $0, \dots, N - 1$ están programados para salir desde el aeropuerto.
- W : una matriz de longitud N para describir las velocidades máximas de los autobuses $0, \dots, N - 1$.
- X : el tiempo que tarda el autobús de reserva en viajar 1 kilómetro.
- M : el número de estaciones de clasificación.
- S : una matriz de longitud M escribiendo las distancias de las estaciones de clasificación al aeropuerto.
- Este procedimiento se llama exactamente una vez para cada caso de prueba, antes de cualquier llamada a `arrival_time`.

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

- Y : hora a la que debe salir el autobús de reserva desde el aeropuerto.
- Este procedimiento debe devolver la hora a la que el autobús N llegaría al hotel.
- Este procedimiento se llama exactamente Q veces.

Ejemplo

Considere la siguiente secuencia de llamadas:

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

Ignorando el autobús 4 (que aún no ha sido programado), la siguiente tabla muestra los tiempos de llegada esperados y reales de los autobuses 0, 1, 2 y 3 en cada estación de clasificación:

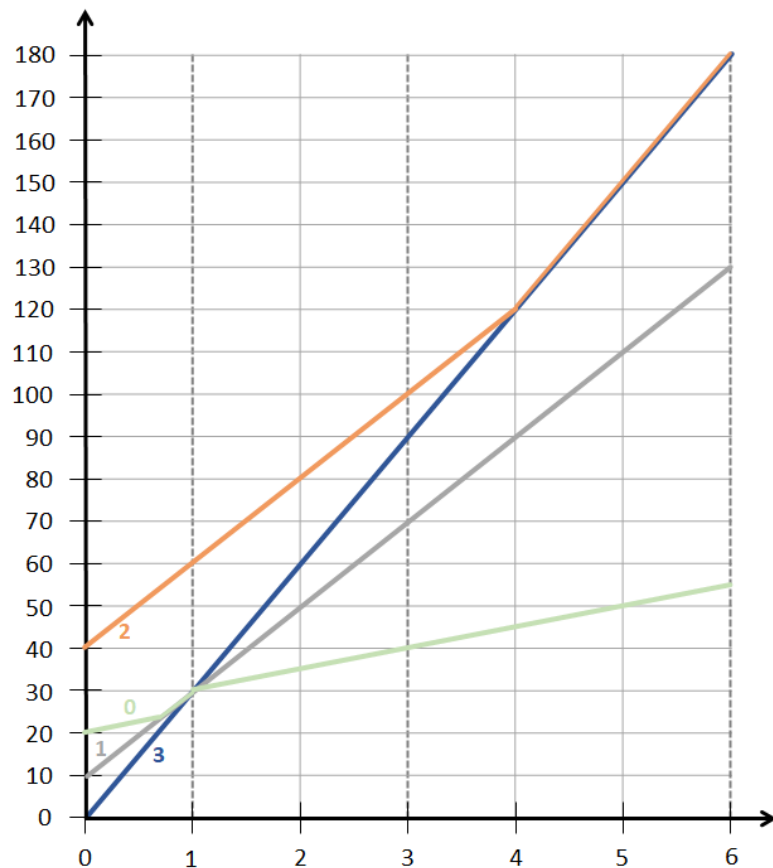
i	$t_{i,0}$	$e_{i,1}$	$t_{i,1}$	$e_{i,2}$	$t_{i,2}$	$e_{i,3}$	$t_{i,3}$
0	20	25	30	40	40	55	55
1	10	30	30	70	70	130	130
2	40	60	60	100	100	160	180
3	0	30	30	90	90	180	180

Los horarios de llegada a la estación. 0 son los horarios en los que está previsto que los autobuses salgan del aeropuerto. Eso es, $t_{i,0} = T[i]$ para $i = 0, 1, 2$ y 3 .

Los tiempos previstos y reales de llegada a la estación de clasificación. 1 se calculan de la siguiente manera:

- Los tiempos previstos de llegada a la estación 1:
 - Autobús 0: $e_{0,1} = t_{0,0} + W[0] \cdot (S[1] - S[0]) = 20 + 5 \cdot 1 = 25$.
 - Autobús 1: $e_{1,1} = t_{1,0} + W[1] \cdot (S[1] - S[0]) = 10 + 20 \cdot 1 = 30$.
 - Autobús 2: $e_{2,1} = t_{2,0} + W[2] \cdot (S[1] - S[0]) = 40 + 20 \cdot 1 = 60$.
 - Autobús 3: $e_{3,1} = t_{3,0} + W[3] \cdot (S[1] - S[0]) = 0 + 30 \cdot 1 = 30$.
- Los horarios de llegada a la estación. 1:
 - Autobuses 1 y 3 llegan a la estación 0 antes que el autobús 0, so $t_{0,1} = \max(e_{0,1}, e_{1,1}, e_{3,1}) = 30$.
 - Autobús 3 llega a la estación 0 antes que el autobús 1, entonces $t_{1,1} = \max(e_{1,1}, e_{3,1}) = 30$.
 - Autobús 0, bus 1 y el autobús 3 llegan a la estación de clasificación 0 antes que el autobús 2, entonces $t_{2,1} = \max(e_{0,1}, e_{1,1}, e_{2,1}, e_{3,1}) = 60$.
 - Ningún autobús llega a la estación. 0 antes que el autobús 3, so $t_{3,1} = \max(e_{3,1}) = 30$.

Podemos trazar las distancias de los autobuses desde el aeropuerto (en kilómetros) frente al tiempo (en segundos) como se muestra en la siguiente figura. Las líneas discontinuas verticales marcan las posiciones de las estaciones de clasificación. Diferentes líneas discontinuas (acompañadas de los índices de autobuses) representan los cuatro autobuses de traslado programados.



```
arrival_time(0)
```

Autobús 4 acepta 10 segundos para viajar 1 kilómetro y ahora está programado para salir del aeropuerto en el 0-ésimo segundo. En este caso, la siguiente tabla muestra los horarios de llegada de cada autobús. Los cambios respecto a la tabla inicial están subrayados.

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

Vemos ese autobús 4 llega al hotel a la 60-ésimo segundo. Por tanto, el procedimiento debería volver 60.

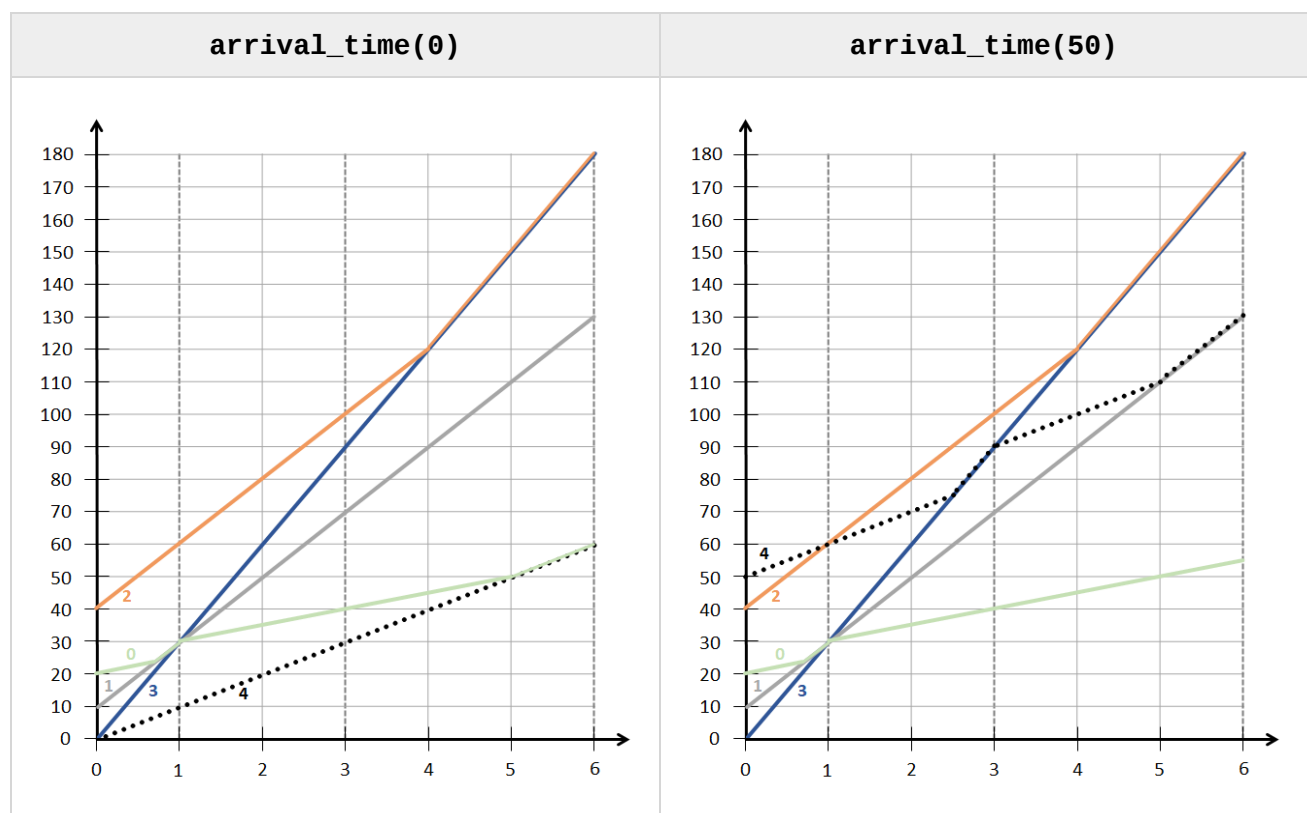
```
arrival_time(50)
```

Autobús 4 ahora está previsto que abandone el aeropuerto a las 50-ésimo segundo. En este caso no hay cambios en los horarios de llegada de los autobuses. 0, 1, 2 y 3. Los horarios de llegada se muestran en la siguiente tabla.

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

Autobús 4 adelanta al autobús más lento 2 en la estación de clasificación 1 ya que llegan al mismo tiempo. Siguiendo, autobús 4 se amontona con el autobús 3 entre estación 1 y estación 2, haciendo autobús 4 llegar a la estación 2 en el 90-ésimo segundo en lugar del 80-th. Después de salir de la estación 2, autobús 4 se amontona con el autobús 1 hasta llegar al hotel. Autobús 4 llega al hotel a la 130-ésimo segundo. Por tanto, el procedimiento debería volver 130.

En las siguientes figuras, trazamos las distancias de los autobuses desde el aeropuerto en función del tiempo. La línea negra punteada representa el autobús de reserva.



Restricciones

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

Subtareas

1. (9 puntos) $N = 1, Q \leq 1\,000$
2. (10 puntos) $M = 2, Q \leq 1\,000$
3. (20 puntos) $N, M, Q \leq 100$
4. (26 puntos) $Q \leq 5\,000$
5. (35 puntos) Sin restricciones adicionales.

Calificador de ejemplo Grader

El calificador de muestra lee la entrada en el siguiente formato:

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

El calificador de ejemplo imprime sus respuestas en el siguiente formato:

- línea $1 + k$ ($0 \leq k < Q$): el valor de retorno de `arrival_time` para la pregunta k