แซง

ถนนจากสนามบินบูดาเปสต์ไปที่โรงแรมฟอร์ราชเดินทางได้ทางเดียวและมีเลนเดียว ถนนนี้มีความยาว L กิโลเมตร

ในช่วงงานแข่งขัน IOI 2023 มีรถบัส $N+1$ คัน วิ่งบนถนนนี้ รถบัสแต่ละคันกำหนดด้วยหมายเลข 0 ถึง N รถบัสหมายเลข i ($0 \leq i < N$) ถูกกำหนดไว้ให้ออกจากสนามบิน ณ วินาทีที่ $T[i]$ ของงาน และสามารถเดินทางได้ 1 กิโลเมตรในระยะเวลา $W[i]$ วินาที รถบัสหมายเลข N เป็นรถบัสสำรอง ซึ่งสามารถเดินทางได้ 1 กิโลเมตรในระยะเวลา X วินาที รถบัสสำรองจะออกจากสนามบิน ณ วินาทีที่ Y ซึ่งยังไม่ได้ถูกกำหนดไว้

การแซงของรถบัสบนถนนไม่สามารถทำได้ แต่รถบัสสามารถที่จะแซงกันได้ที่ **สถานีเรียงรถ** โดยมีสถานีเรียงรถทั้งหมด M ($M > 1$) สถานี ซึ่งกำหนดด้วยหมายเลข 0 ถึง $M-1$ อยู่ในตำแหน่งที่แตกต่างกันบนถนน สถานีเรียงรถ j ($0 \leq j < M$) จะอยู่ตามเส้นทางบนถนนในตำแหน่งกิโลเมตรที่ $S[j]$ โดยเริ่มต้นจากสนามบิน สถานีเรียงรถนั้นเรียงตามระยะทางจากสนามบินจากน้อยไปมาก ซึ่งแปลว่า $S[j] < S[j+1]$ สำหรับ $0 \leq j \leq M-2$ โดยสถานีเรียงรถแรกคือสนามบินและสถานีเรียงรถสุดท้ายคือโรงแรม ซึ่งแปลว่า $S[0] = 0$ และ $S[M-1] = L$

รถบัสแต่ละคันจะวิ่งด้วยความเร็วสูงสุดจนกระทั่งตามทันรถบัสคันที่อยู่ด้านหน้าที่วิ่งช้ากว่าบนถนน ซึ่งในกรณีนั้นรถบัสจะต้องวิ่งต่อกันและถูกบังคับให้เคลื่อนที่ด้วยความเร็วของรถบัสคันที่ช้ากว่าจนกระทั่งถึงสถานีเรียงรถถัดไป รถบัสที่เร็วกว่าจึงจะแซงรถบัสที่ช้ากว่าได้

กล่าวคือ สำหรับ i กับ j ซึ่ง $0 \leq i \leq N$ และ $0 \leq j < M$ นั้น วินาทีที่ $t_{i,j}$ เป็นเวลาที่รถบัส i **ถึงที่** สถานีเรียงรถ j ซึ่งสามารถนิยามได้ดังนี้ ให้ $t_{i,0} = T[i]$ สำหรับแต่ละ $0 \leq i < N$, และให้ $t_{N,0} = Y$

สำหรับแต่ละ j ซึ่ง $0 < j < M$

- กำหนดให้ $e_{i,j}$ คือเวลาที่ **คาดว่าจะถึง** (ในหน่วยวินาทีที่) ของรถบัส i ที่สถานีเรียงรถ j ซึ่งเป็นเวลาที่รถบัส i จะถึงสถานีเรียงรถ j ถ้ารถบัสคันดังกล่าวเคลื่อนที่ด้วยความเร็วสูงสุดตั้งแต่เวลาที่รถบัสนั้นถึงสถานีเรียงรถ $j-1$ นั่นคือ ให้
  - $e_{i,j} = t_{i,j-1} + W[i] \cdot (S[j] - S[j-1])$ สำหรับ $0 \leq i < N$ และ
  - $e_{N,j} = t_{N,j-1} + X \cdot (S[j] - S[j-1])$
- รถบัส i ถึงสถานีเรียงรถ j ณ เวลาที่*มากที่สุด* ระหว่างเวลาที่คาดว่าจะถึงของรถบัส i และของรถบัสคันอื่นที่ถึงที่สถานี $j-1$ ก่อนรถบัส i กล่าวคือให้ $t_{i,j}$ เป็นค่ามากที่สุดของ $e_{i,j}$ และทุก ๆ $e_{k,j}$ ซึ่ง $0 \leq k \leq N$ และ $t_{k,j-1} < t_{i,j-1}$

ทางผู้จัดงาน IOI ต้องการที่จะกำหนดเวลาออกจากสนามบินของรถบัสสำรอง งานของคุณคือตอบคำถาม Q คำถามจากผู้จัดงาน ซึ่งอยู่ในรูปแบบต่อไปนี้ หากกำหนดให้วินาทีที่ Y เป็นเวลาที่รถบัสสำรองจะออกจากสนามบิน แล้วถามว่า ณ วินาทีที่เท่าไร รถบัสสำรองจึงจะมาถึงโรงแรม

รายละเอียดการเขียนโปรแกรม

คุณจะต้องเขียนสองฟังก์ชันต่อไปนี้

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

- L: ความยาวของถนน
- N: จำนวนของรถบัสที่ไม่ใช่รถบัสสำรอง
- T: อาร์เรย์ขนาด N ที่ระบุเวลาที่รถบัสไม่สำรองออกจากสนามบิน
- W: อาร์เรย์ขนาด N ที่กำหนดความเร็วสูงสุดของรถบัสไม่สำรอง
- X: ระยะเวลาที่รถบัสสำรองใช้ในการเดินทาง 1 กิโลเมตร
- M: จำนวนสถานีเรียงรถ
- S: อาร์เรย์ขนาด M ที่ระบุระยะทางจากสนามบินไปยังสถานีเรียงรถ
- ฟังก์ชันนี้จะถูกเรียกเพียงครั้งเดียวเท่านั้น ก่อนการเรียก `arrival_time` ใด ๆ

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

- Y: เวลาที่รถบัสสำรอง (รถบัสหมายเลข N) จะออกจากสนามบิน
- ฟังก์ชันนี้ต้องคืนค่าเวลาที่รถบัสสำรองมาถึงที่โรงแรม
- ฟังก์ชันนี้จะถูกเรียกจำนวน Q ครั้งพอดี

ตัวอย่าง

พิจารณาลำดับการเรียกฟังก์ชันต่อไปนี้

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

ตารางต่อไปนี้แสดงเวลาที่คาดว่าจะถึงและเวลาที่ถึงจริงของรถบัสไม่สำรองที่ไปถึงสถานีเรียงรถต่าง ๆ โดยไม่คำนึงถึงรถบัสหมายเลข 4 (ที่ยังไม่ได้ระบุเวลา)

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

เวลาที่ถึงสถานีเรียงรถ 0 เป็นเวลาที่รถบัสถูกกำหนดให้ออกจากสนามบิน นั่นคือ $t_{i,0} = T[i]$ สำหรับ $0 \le i \le 3$

เวลาที่คาดว่าจะถึงและเวลาที่ถึงจริงของสถานีเรียงรถ 1 สามารถคำนวณได้ดังต่อไปนี้:

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

    - รถบัส 2: $e_{2,1} = t_{2,0} + W[2] \cdot (S[1] - S[0]) = 40 + 20 \cdot 1 = 60$
    - รถบัส 3: $e_{3,1} = t_{3,0} + W[3] \cdot (S[1] - S[0]) = 0 + 30 \cdot 1 = 30$
- เวลาจริงที่ถึงสถานีเรียงรถ 1
    - รถบัส 1 กับ รถบัส 3 ถึงสถานี 0 ก่อนรถบัส 0 ดังนั้น $t_{0,1} = \max([e_{0,1}, e_{1,1}, e_{3,1}]) = 30$
    - รถบัส 3 ถึงสถานี 0 ก่อนรถบัส 1 ดังนั้น $t_{1,1} = \max([e_{1,1}, e_{3,1}]) = 30$
    - รถบัส 0 รถบัส 1 และ รถบัส 3 ถึงสถานีเรียงรถ 0 ก่อนรถบัส 2 ดังนั้น $t_{2,1} = \max([e_{0,1}, e_{1,1}, e_{2,1}, e_{3,1}]) = 60$
    - ไม่มีรถบัสอื่นถึงสถานี 0 ก่อนรถบัส 3 ดังนั้น $t_{3,1} = \max([e_{3,1}]) = 30$

```
arrival_time(0)
```

รถบัส 4 ใช้เวลา 10 วินาทีที่จะเดินทาง 1 กิโลเมตรและถูกกำหนดให้ออกจากสนามบินที่วินาทีที่ 0 ในกรณีนี้ ตารางต่อไปนี้แสดงเวลาที่ถึงของรถบัสแต่ละคัน โดยเวลาที่คาดว่าจะถึงและเวลาที่ถึงจริงของรถบัสไม่สำรอง จุดที่เปลี่ยนแปลงทั้งหมดได้ขีดเส้นใต้ไว้

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

เราจะเห็นว่ารถบัส 4 ถึงโรงแรม ณ วินาทีที่ 60 ดังนั้นฟังก์ชันต้องคืนค่า 60

```
arrival_time(50)
```

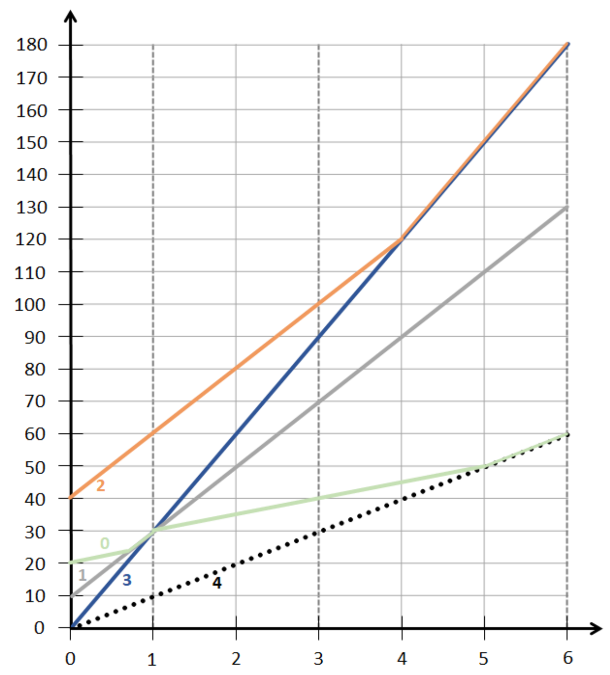
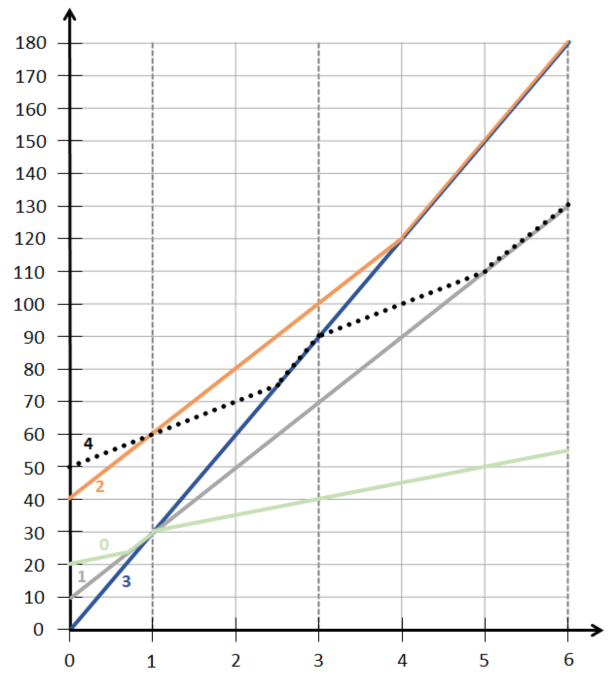
รถบัส 4 ถูกกำหนดให้ออกจากสนามบินที่วินาทีที่ 50 ในกรณีนี้ ไม่มีการเปลี่ยนแปลงเวลาที่ถึงของรถบัสไม่สำรองจากตารางแรก ตารางต่อไปนี้แสดงเวลาที่ถึงของรถบัสแต่ละคัน

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

รถบัส 4 แซงรถบัส 2 ที่ช้ากว่าที่สถานีเรียงรถ 1 โดยที่ทั้งสองคันถึงที่เวลาเดียวกัน หลังจากนั้น รถบัส 4 วิ่งติดกับรถบัส 3 ระหว่างสถานี 1 และ สถานี 2 ทำให้รถบัส 4 ถึงสถานีที่ 2 ณ วินาทีที่ 90 แทนที่จะเป็นวินาทีที่ 80 และหลังจากออก

จากสถานี 2 นั้น รถบัส 4 วิ่งติดไปกับรถบัส 1 จนกระทั่งถึงโรงแรม รถบัส 4 ถึงโรงแรม ณ วินาทีที่ 130 ดังนั้นฟังก์ชันต้องคืนค่า 130

เราสามารถพล็อตกราฟเวลาที่รถบัสแต่ละคันต้องใช้เพื่อไปยังระยะทางต่าง ๆ จากสนามบินได้ แกน x ของกราฟคือระยะทางจากสนามบิน (หน่วยกิโลเมตร) และแกน y ของกราฟคือเวลา (หน่วยวินาที) เส้นประแนวตั้งระบุตำแหน่งของสถานีเรียงรถ เส้นทึบต่าง ๆ (พร้อมด้วยหมายเลขของรถบัส) แสดงถึงรถบัสไม่สำรองทั้งสี่คัน เส้นประสีดำแสดงถึงรถบัสสำรอง

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

ข้อจำกัด

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

ปัญหาย่อย

1. (9 คะแนน) $N = 1, Q \le 1\,000$
2. (10 คะแนน) $M = 2, Q \le 1\,000$

3. (20 คะแนน) $N, M, Q \le 100$
4. (26 คะแนน) $Q \le 5\,000$
5. (35 คะแนน) ไม่มีข้อจำกัดใดเพิ่มเติม

เกรดเดอร์ตัวอย่าง

เกรดเดอร์ตัวอย่างจะอ่านค่าอินพุตในรูปแบบดังนี้

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

เกรดเดอร์ตัวอย่างจะพิมพ์ผลลัพธ์ของคุณตามรูปแบบต่อไปนี้

- บรรทัดที่ $1 + k$ ($0 \le k < Q$): คืนค่าผลลัพธ์ของ `arrival_time` ในคำถามที่ k