



Écoulement

Le temps s'améliore et la souris Binna a l'intention de remplir sa piscine. Cette dernière est particulière, car elle n'a pas qu'un seul robinet et un seul siphon, mais un total de n robinets et siphons que Binna peut ouvrir et fermer à sa guise. Pour chaque robinet et chaque siphon, Binna connaît la vitesse à laquelle l'eau s'écoule (elle est exprimée en litres par minute). Notez qu'un nombre positif indique que de l'eau s'écoule d'un robinet, tandis qu'un nombre négatif indique que de l'eau s'écoule par un siphon.

La souris Binna ouvre l'un des robinets et rentre à l'intérieur. La souris Stofl arrive et décide de faire une farce à Binna en jouant avec sa piscine. Il commence à ouvrir et à fermer les robinets et les siphons de façon arbitraire. Lorsqu'il sent que Binna revient, il ferme rapidement tous les robinets et tous les siphons et fait comme s'il venait de passer. Binna se demande alors pourquoi il y a tant d'éclaboussures d'eau autour d'elle et pourquoi son robinet est fermé.

Stofl veut avouer à Binna ce qu'il a fait. Il veut également connaître la quantité maximale d'eau qui se trouvait dans la piscine à un moment donné pendant que Binna était dans la maison. Pouvez-vous l'aider à trouver cette information ?

Pour ce problème, nous ne nous préoccupons pas de la capacité maximale de la piscine de Binna, donc vous pouvez supposer que sa capacité est infinie. Bien entendu, il ne peut y avoir de quantité négative d'eau dans la piscine, c'est-à-dire que si la piscine est vide et qu'une bonde est ouverte, il n'y a tout simplement pas d'eau qui s'écoule.

Entrée

L'entrée commence par une ligne qui contient un seul entier m représentant le nombre de fois que Stofl manipule les robinets et les siphons.

Cette ligne est suivie de m lignes, chaque ligne contenant trois entiers s, e, c . Ce qui indique qu'un robinet ou un drain avec un débit d'eau de c litres par minute a été ouvert à la minute s et fermé avant la minute e (l'eau coule à la minute s mais pas à la minute e). Il est garanti que le premier d'entre eux commence à l'instant $s = 0$ et est un robinet, c'est-à-dire $c > 0$.

Sortie

Affichez une ligne contenant un seul nombre entier représentant la quantité maximale d'eau dans la piscine à tout moment.

Limites

Il y a 4 groupes de test. Pour chaque groupe de test, il y a $1 \leq m \leq 10^5$, $0 \leq s_i < e_i \leq 10^8$ et $-10^4 \leq c_i \leq 10^4$.

- Dans la sous-tâche 1, qui vaut 20 points, il y a $c_i \geq 0$, c'est-à-dire qu'il n'y a pas de siphons, il y a $1 \leq m \leq 1000$ et $0 \leq s_i < e_i \leq 10^4$.
- Dans la sous-tâche 2, qui vaut 20 points, il y a $1 \leq m \leq 1000$ et $0 \leq s_i < e_i \leq 10^4$.
- Dans la sous-tâche 3, qui vaut 20 points, il y a $0 \leq s_i < e_i \leq 10^6$.
- Dans la sous-tâche 4, qui vaut 40 points, il n'y a pas de restrictions supplémentaires.



Exemples

Entrée	Sortie
4 0 3 2 1 2 1 3 4 4 1 4 1	14

À la minute 0, on a 2 litres qui coulent dans la piscine donc on a 2 litres dans la piscine. Ensuite, dans la minute 1, deux robinets supplémentaires s'ouvrent, conduisant à 4 litres s'écoulant dans la piscine, soit un total de 6 litres dans la piscine. Dans la minute 2 un des robinets se referme, donc 3 litres s'y écoulent par minute donnant 9 litres dans la piscine. Puis le premier robinet est fermé et un nouveau s'ouvre, menant à un afflux de 5 litres en minute 3 donc la piscine est remplie de 14 litres après cela. À la minute 4, tous les robinets sont à nouveau fermés. Le maximum est obtenu en minute 3 avec 14 litres.

Entrée	Sortie
6 0 2 1 1 3 -3 2 5 3 2 3 -4 0 4 1 3 5 2	11

À la minute 0, deux robinets s'ouvrent, avec un débit total de 2 litres/min. Ainsi, après la minute 0, nous avons 2 litres dans la piscine. À la minute 1, un siphon est ouvert, son débit de -3 étant supérieur au flux d'eau entrant, nous avons 1 litre par minute qui quitte la piscine à ce moment-là. Après la minute 1, il ne reste donc plus qu'un litre. À la minute 2, Stofl ouvre un robinet et un siphon supplémentaires et un robinet se ferme, ce qui donne un débit total de -3 litres par minute. Après la minute 2, la piscine est vide. À la minute 3, les drains sont fermés et un nouveau robinet est ouvert. À ce stade, 6 litres par minute s'écoulent dans la piscine, de sorte qu'après la minute 3, il y a 6 litres dans la piscine. À la minute 4, l'un des robinets a été fermé, de sorte que 5 litres/minute s'écoulent dans la piscine, ce qui donne un total de 11 litres. Ensuite, tous les robinets sont fermés. Le maximum est obtenu à la minute 4 avec 11 litres.

Broderies

Lorsque Stofl et Binna ont trouvé qu'ils s'étaient qualifiées pour l'IOI en Hongrie, elles ont décidé qu'ils allaient non seulement se préparer incroyablement bien pour le concours, mais aussi apprendre quelque chose sur la culture hongroise à l'avance. Au cours de leurs recherches, ils sont tombés sur des photos de costumes traditionnels hongrois. Les costumes féminins et masculins sont richement brodés de fleurs colorées et d'autres motifs.

Après avoir passé des heures à regarder des photos de ces broderies, Stofl et Binna s'accordent à dire qu'ils veulent en porter sur leurs T-shirts pour l'IOI. Sur un costume traditionnel, ils trouvent plusieurs motifs sur le col qui leur plaisent particulièrement. En fait, chacun de ces motifs ressemble un peu à une séquence de parenthèses ouvertes et fermées. En bons programmeurs qu'ils sont, ils se rendent bien sûr immédiatement compte que ces motifs de parenthèses ne sont pas tous réguliers. Ils conviennent qu'ils ne peuvent pas adopter un tel modèle sur leurs T-shirts (il serait un peu embarrassant d'avoir des parenthèses non régulières sur le T-shirt de l'IOI).

Pour cette raison, ils veulent maintenant savoir combien de sous-motifs différents ils peuvent trouver dans un modèle pour que la séquence de parenthèses montrée soit régulière. (Une séquence de parenthèses est régulière si elle peut être obtenue en écrivant une expression mathématique avec seulement les lettres '(' , ')' , '+' et '1', et en supprimant ensuite tous les caractères '+' et '1'). Take the pattern "((()())(" as an example. This pattern has 8 subpatterns :

"((()())", "(()()", "((())", "(()()", "(())", "()()", "()", et "".

Notez que chaque sous-motif n'est compté qu'une seule fois, même s'il apparaît à plusieurs endroits, et que le sous-motif vide est compté comme régulier.

Comme le nombre de ces sous-motifs peut être très important, les souris Stofl et Binna veulent que le résultat soit modulo $10^9 + 7$.

Entrée

La première ligne de l'entrée contient n , la longueur du motif. La deuxième ligne contient le motif.

Sortie

Affichez le nombre de différents sous-modèles qui sont des séquences de parenthèses régulières modulo $10^9 + 7$.

Limites

Il y a quatre sous-tâches, chacune valant 25 points.

- Dans la sous-tâche 1, il y a $1 \leq n \leq 16$.
- Dans la sous-tâche 2, il y a $1 \leq n \leq 36$.
- Dans la sous-tâche 3, il y a $1 \leq n \leq 200$.
- Dans la sous-tâche 4, il y a $1 \leq n \leq 3\,000$.



Exemple

Entrée	Sortie
9 ((()())(8



Méga-saut

La souris Stofl se trouve à l'origine d'un système de coordonnées rectangulaires. Stofl peut effectuer un certain nombre de mouvements : Il peut marcher d'une unité horizontalement, verticalement ou diagonalement. En d'autres termes, il peut atteindre les 8 positions entourant sa position actuelle en un seul mouvement. De plus, il peut faire un *méga-saut*, où il saute de k unités horizontalement et de l unités verticalement.

Pouvez-vous trouver le nombre minimum de mouvements nécessaires pour que Stofl atteigne une position particulière ?

Entrée

Sur la première ligne, il y a trois entiers k , l et q . Les lignes q suivent, la i ème ligne contient les entiers x_i et y_i .

Sortie

Affichez q lignes. La i ème ligne doit contenir un entier d_i .

d_i doit être le nombre minimal de déplacements pour aller de $(0,0)$ à (x_i, y_i) . Lors de chaque déplacement, Stofl peut aller de (x, y) à $(x \pm 1, y)$ ou $(x, y \pm 1)$ ou $(x \pm 1, y \pm 1)$ ou $(x \pm k, y \pm l)$.

Limites

Il y a 4 sous-tâches, dans chacune d'elles $q = 100$, $0 \leq k \leq 1000$, $0 \leq l \leq 1000$, $0 \leq x \leq 10^9$, $0 \leq y \leq 10^9$.

- Dans la sous-tâche 1, qui vaut 10 points, il y a $k = 1$, $l = 1$.
- Dans la sous-tâche 2, qui vaut 20 points, il y a $1 \leq k \leq 1000$, $l = 0$.
- Dans la sous-tâche 3, qui vaut 20 points, il y a $0 \leq k \leq 100$, $0 \leq l \leq 100$, $0 \leq x \leq 100$, $0 \leq y \leq 100$.
- Dans la sous-tâche 4, qui vaut 50 points, il n'y a pas de contraintes supplémentaires.

Exemples

Entrée	Sortie
4 3 2 3 3 10 1	2 4

Ici, il y a $k = 4$, $l = 3$. Dans la première question, Stofl peut faire un méga-saut de $(0,0)$ à $(4,3)$, et ensuite marcher un pas jusqu'à $(3,3)$, pour un total de 2 déplacements. Dans la seconde question, Stofl peut faire deux méga-sauts $(0,0) - (4,3) - (8,0)$, puis faire deux pas $(8,0) - (9,1) - (10,1)$.



Entrée	Sortie
5 0 2	2
6 0	4
4 4	

Ici, il y a $k = 5, l = 0$. Dans la première question, Stofl peut marcher jusqu'à $(1, 0)$ et faire un méga-saut jusqu'à $(6, 0)$. Dans la seconde question, Stofl peut faire 4 pas $(0, 0) - (1, 1) - \dots - (4, 4)$. Il ne peut pas aller plus vite avec des méga-sauts.

Ennamis

Souris Binna et Souris Stofl viennent d'arriver dans leur nouvelle école d'informatique. Leur classe est composée de N autres étudiants et comme elles ne connaissent pas encore tous leurs noms, elles décident de les numéroter de 0 à $N - 1$. D'une part, la souris Stofl est très timide et ne veut pas prendre le risque de parler à deux souris qui ne s'aiment pas. D'autre part, la souris Binna veut rencontrer le plus possible ses nouveaux camarades de classe.

Néanmoins, une chose qu'elles veulent toutes deux par-dessus tout, est rester ensemble, car elles sont des Superbes Souris Sociables depuis un certain temps. Elles doivent donc décider avec qui elles seront amies dans les années à venir. Sur la base des critères précédents, le groupe d'amis doit respecter les règles suivantes : Aucune souris ne doit se détester au sein du groupe d'amis et le groupe doit être aussi grand que possible.

Malheureusement, elles ne savent pas encore qui aime et qui n'aime pas qui. Pour y remédier, elles commencent à écouter les différents potins qui circulent dans l'école. Comme ce sont des souris intelligentes, elles savent que l'ennemi de leur ennemi est leur ami et que l'ami de leur ennemi est aussi leur ennemi. De plus, l'ami de leur ami est leur ami (et leurs amis sont aussi leurs amis. . .). Par exemple, si la souris 0 est amie avec la souris 1 et que la souris 1 n'aime pas la souris 2 , alors la souris 0 n'aimera pas non plus la souris 2 . Notez que toutes les relations sont symétriques, cela signifie que si la souris 0 aime la souris 1 , alors la souris 1 aime la souris 0 .

Enfin, les souris Binna et Stofl ne prennent en compte que les ragots cohérents. Elles écoutent les rumeurs une par une et ne croient que les rumeurs qui sont cohérentes avec leurs connaissances actuelles. (Par exemple, si elles savent que la Souris 1 aime la Souris 2 mais qu'elles entendent ensuite que la Souris 2 n'aime pas la Souris 1 , elles ne tiendront pas compte de cette dernière affirmation).

Entrée

Sur la première ligne, il y a deux nombres N et Q — le nombre de souris dans la classe et le nombre de ragots qu'elles entendent.

Sur chacune des lignes Q suivantes, il y a trois nombres s, a, b ($0 \leq s \leq 1, 0 \leq a \neq b \leq N - 1$) représentant un potin. Le premier étant un 0 signifie que les deux souris sont amies et étant un 1 signifie qu'elles sont ennemies. Les deux nombres suivants représentent les deux souris qui sont soit amies, soit ennemies.

Sortie

Tout d'abord, affichez Q lignes. Dans la i ème ligne, affichez "YES" si le i ème ragot de l'entrée est cohérent avec les ragots précédents et "NO" sinon.

Ensuite, il faut sortir un seul entier positif, le plus grand groupe possible tel que deux souris ne se détestent pas l'une l'autre. Notez que dans ce groupe, il n'est pas nécessaire que tout le monde se connaisse.

Limites

Il y a 3 sous-tâches. Dans toutes les sous-tâches, il y a $2 \leq N \leq 100\,000$ et $1 \leq Q \leq 350\,000$.

- Dans la sous-tâche 1, qui vaut 30 points, il est garanti que $2 \leq N \leq 1000$, $1 \leq Q \leq 2000$ et que, pour chaque souris, après avoir traité tous les cohérents potins, elles ne détestent pas



plus d'une autre souris.

- Dans la sous-tâche 2, qui vaut 30 points, il est garanti que $2 \leq N \leq 1000$ et $1 \leq Q \leq 2000$
- Dans la sous-tâche 3, qui vaut 40 points, il n'y a pas de contrainte supplémentaire.

Pour chaque sous-tâche, vous pouvez obtenir **50%** des points alloués si toutes les Q premières lignes sont correctes mais que la taille du plus grand groupe est incorrecte.

Examples

Entrée	Sortie
4 4	YES
0 0 1	YES
1 2 3	NO
1 0 1	YES
1 2 3	3

Entrée	Sortie
5 6	YES
0 0 1	YES
1 0 2	YES
0 2 3	NO
0 1 2	YES
1 4 3	NO
1 4 1	3