

Audition

N danseurs auditionnent pour avoir la chance de participer au jeu télévisé *Dance avec les Stars*. Les danseurs sont numérotés de 1 à N selon leurs niveaux, où 1 est le moins bon danseur et N est le meilleur danseur.

La scène est divisée en D segments, numérotés de 1 à D de gauche à droite. Il y a J juges qui donnent des scores aux candidats. Le i -ème juge peut uniquement voir les danseurs des segments de l_i à r_i inclus.

Débordés et en retard, les juges ont décidé de faire danser tous les candidats en même temps. Le i -ème candidat doit apparaître sur scène exactement b_i secondes après le début de l'audition. Les danseurs arrivent sur scène à des secondes distinctes.

À chaque seconde :

1. Chaque danseur actuellement sur scène se déplace sur le segment suivant vers la droite (ou sort de scène s'il est sur le dernier segment).
2. Si un danseur arrive à cette seconde, il entre sur le segment 1.
3. Chaque juge donne un point au danseur ayant le meilleur niveau parmi les danseurs qu'il peut voir.

Notez que :

- Un danseur peut recevoir plusieurs points lors de la même seconde.
- Un danseur peut recevoir plusieurs point du même juge à des secondes différentes.
- Si un juge ne voit aucun danseur, alors il ne donne aucun point lors de cette seconde.

L'audition se termine lorsque tous les danseurs sont sortis de scène. Le *score final* d'un danseur est le nombre total de points reçus des juges lors de l'audition.

Pouvez vous aider les organisateurs à calculer le score final de chaque danseur ?

Sous-tâches et Contraintes

Pour chaque sous-tâche, il est garanti que :

- $1 \leq N \leq 100\,000$.
- $1 \leq J \leq 100\,000$.
- $1 \leq D \leq 100\,000$.
- $1 \leq b_i \leq 1\,000\,000\,000$ pour tout i .
- Les danseurs ont des b_i distincts.
- $1 \leq l_i \leq r_i \leq D$ pour tout i .

Des contraintes supplémentaires pour chaque sous-tâche sont données ci-dessous.

Sous-tâche	Points	Contrainte supplémentaire
1	7	$N, D, J \leq 100$ et $b_i \leq 100$ pour tout i .
2	9	$N, D, J \leq 100$.
3	14	$N, J \leq 1000$.
4	29	$J = 1$.
5	34	$b_i < b_{i+1}$ pour tout i .
6	7	Aucune contrainte supplémentaire

Entrée

- La première ligne de l'entrée contient les trois entiers N , J et D .
- La deuxième ligne contient N entiers b_1, b_2, \dots, b_N .
- Les J lignes suivante décrivent les juges. La i -ème ligne contient l_i et r_i .

Sortie

Affichez N lignes : la i -ème ligne doit contenir le score final du i -ème danseur.

Entrée d'Exemple

```
3 2 5
3 4 1
2 3
3 5
```

Sortie d'Exemple

```
1
5
5
```

Explications

Dans l'exemple 1, la table ci-dessous décrit chaque seconde de l'audition.

Second	Stage					Judge 1 gives a point to...	Judge 2 gives a point to...
	Judge 1		Judge 2				
1	3					-	-
2		3				3	-
3	1		3			3	3
4	2	1		3		1	3
5		2	1		3	2	3
6			2	1		2	2
7				2	1	-	2
8					2	-	2

Koh-Lanta

Un nombre impair de personnes participent à la phase finale du jeu télévisé *Koh-Lanta*. Les candidats sont numérotés de 1 à N en fonction de leur force, le candidat 1 étant le plus faible et le candidat N étant le plus fort.

Les candidats se tiennent en ligne, avec le i -ème candidat se trouvant à p_i mètres de l'extrémité gauche de la ligne. Les positions des candidats sont distinctes.

La finale consiste en $\frac{N-1}{2}$ tours. Lors de chaque tour, seuls les trois candidats avec les plus petits p_i participent (c'est à dire, les trois candidats les plus à gauche sur la ligne). Parmi eux trois, les candidats ayant la **plus grande force** et la **plus petite force** sont éliminés (les deux candidats les plus faibles font alliance pour battre le plus fort, puis le plus faible est battu par le candidat restant). Les candidats éliminés quittent la ligne et ne participent à aucun autre tour.

Lorsque tous les tours sont terminés, le candidat restant est déclaré vainqueur.

Les organisateurs du jeu n'ont pas encore finalisé les valeurs des p_i . Par conséquent, ils vous ont demandé d'effectuer Q opérations. Il existe deux types d'opération :

- U: Le candidat x_i se déplace à une nouvelle position, à v_i mètres de l'extrémité gauche de la ligne. Après chaque opération U, les positions des candidats sont distinctes.
- W: Étant données les positions actuelles, calculer qui est le vainqueur.

Sous-tâches et Contraintes

Pour chaque sous-tâche, il est garanti que :

- $3 \leq N \leq 300\,000$, et N est impair.
- $1 \leq p_i \leq 1\,000\,000$ pour tout i .
- $1 \leq Q \leq 300\,000$.
- $1 \leq x_i \leq N$ pour tout i .
- $1 \leq v_i \leq 1\,000\,000$ pour tout i .
- Avant et après chaque opération, les positions p_i des candidats sont distinctes.

Des contraintes supplémentaires pour chaque sous-tâche sont données ci-dessous.

Sous-tâche	Points	Contrainte supplémentaire
1	5	$N, Q \leq 100$
2	7	$N, Q \leq 5000$
3	15	Voir * ci-dessous.
4	11	Voir ** ci-dessous.
5	20	$x_i = x_j$ pour tous i et j : chaque opération U déplace le même candidat.
6	26	Lors de chaque opération W, les positions des candidats sont inférieures ou égales à N .
7	16	Aucune contrainte supplémentaire.

*Dans la sous-tâche 3: Seuls les candidats ayant une position initiale ≤ 100 bougent, et ne se déplacent que vers d'autres positions ≤ 100 (c'est à dire, pour chaque opération U, $p_{x_i}, v_i \leq 100$ pour tout i). Tous les candidats ayant une position > 100 ne se déplacent pas.

**Dans la sous-tâche 4: Seuls les candidats ayant une position initiale ≤ 999900 bougent, et ne se déplacent que vers d'autres positions ≤ 999900 (c'est à dire, pour chaque opération U, $p_{x_i}, v_i \leq 999900$ pour tout i). Tous les candidats ayant une position > 999900 ne se déplacent pas.

Entrée

- La première ligne de l'entrée contient les deux entiers N et Q .
- La deuxième ligne de l'entrée contient N entiers p_1, p_2, \dots, p_N .
- Suivent Q lignes décrivant les opérations. La i -ème ligne commence soit par U ou W décrivant le type de l'opération :
 - S'il s'agit d'une opération U, les deux entiers x_i et v_i suivent.
 - S'il s'agit d'une opération W, rien ne suit.

Sortie

Pour chaque opération W, affichez une ligne contenant le vainqueur étant données les positions des candidats à cet instant.

Entrée d'Exemple 1

```
5 8
2 1 4 8 6
W
U 1 7
W
U 5 2
U 2 9
U 4 3
U 4 1
W
```

Sortie d'Exemple 1

```
4
3
2
```

Entrée d'Exemple 2

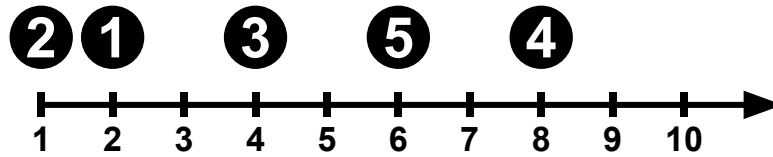
```
11 7
10 3 9 7 2 5 4 8 1 11 6
W
W
U 9 313
U 6 1
U 10 5
U 9 11
W
```

Sortie d'Exemple 2

```
6
6
7
```

Explications

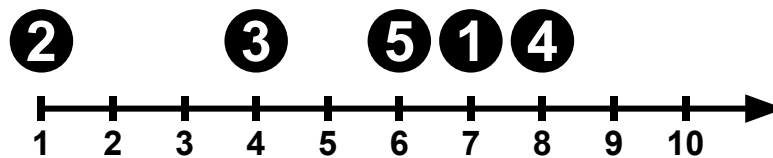
Dans l'exemple 1, il y a $N = 5$ candidats et $Q = 8$ opérations. Lors de la première opération W, les candidats sont positionnés comme suit :



Le vainqueur est alors le candidat 4, car :

- Lors du premier tour, les candidats 2, 1 et 3 participent. 1 et 3 sont éliminés.
- Lors du second tour, les candidats 2, 5 et 4 participent. 2 et 5 sont éliminés.

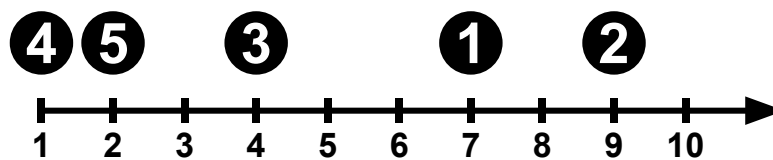
Lors de la seconde opération W, les candidats sont positionnés comme suit :



Le vainqueur est alors le candidat 3, car :

- Lors du premier tour, les candidats 2, 3 et 5 participent. 2 et 5 sont éliminés.
- Lors du second tour, les candidats 3, 1 et 4 participent. 1 et 4 sont éliminés.

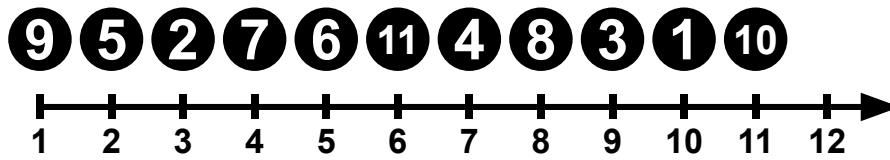
Lors de la troisième opération W, les candidats sont positionnés comme suit :



Le vainqueur est alors le candidat 2, car :

- Lors du premier tour, les candidats 4, 5 et 3 participent. 3 et 5 sont éliminés.
- Lors du second tour, les candidats 4, 1 et 2 participent. 1 et 4 sont éliminés.

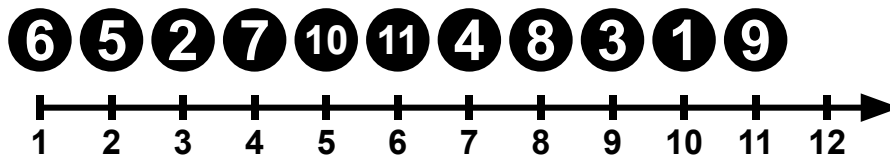
Dans l'exemple 2, il y a $N = 11$ candidats et $Q = 2$ opérations. Lors de la première opération W, les candidats sont positionnés comme suit :



Le vainqueur est alors le candidat 6, car :

- Lors du premier tour, les candidats 9, 5 et 2 participent. 2 et 9 sont éliminés.
- Lors du deuxième tour, les candidats 5, 7 et 6 participent. 5 et 7 sont éliminés.
- Lors du troisième tour, les candidats 6, 11 et 4 participent. 4 et 11 sont éliminés.
- Lors du quatrième tour, les candidats 6, 8 et 3 participent. 3 et 8 sont éliminés.
- Lors du cinquième tour, les candidats 6, 1 et 10 participent. 1 et 10 sont éliminés.

Lors de la troisième opération W, les candidats sont positionnés comme suit :



Le vainqueur est alors le candidat 7, car :

- Lors du premier tour, les candidats 6, 5 et 2 participent. 2 et 6 sont éliminés.
- Lors du deuxième tour, les candidats 5, 7 et 10 participent. 5 et 10 sont éliminés.
- Lors du troisième tour, les candidats 7, 11 et 4 participent. 4 et 11 sont éliminés.
- Lors du quatrième tour, les candidats 7, 8 et 3 participent. 3 et 8 sont éliminés.
- Lors du cinquième tour, les candidats 7, 1 et 9 participent. 1 et 9 sont éliminés.

Crop Circles

Des touristes extra-terrestres adorent visiter la terre pour voir des artistes humains dessiner des cercles dans des champs. Vous êtes responsable de produire le plus joli motif de cercles jamais dessiné ! Le champ dans lequel vous travaillez peut-être représenté par un plan 2D infini. Des centaines d'années d'études sur l'esthétique extra-terrestre vous donnent des contraintes à respecter : Vous devez dessiner exactement N cercles, le centre du i -ème d'entre eux devant se trouver au point de coordonnées entières (x_i, y_i) .

Votre tâche est de sélectionner un rayon positif ou nul r_i pour chaque cercle, de manière à ce que les cercles ne se superposent pas. Deux cercles peuvent cependant se toucher sur leurs bords. Notez que les rayons sélectionnés ne sont pas nécessairement entiers. Formellement, les cercles i et j se superposent si et seulement si :

$$(x_i - x_j)^2 + (y_i - y_j)^2 < (r_i + r_j)^2$$

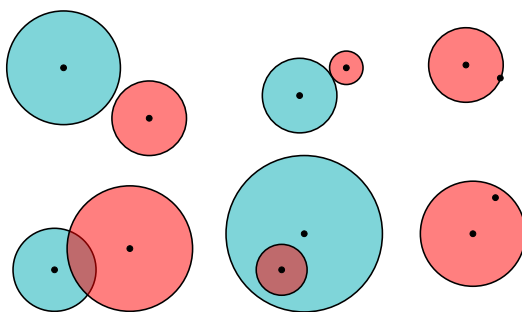


Figure 1: Les trois exemples du haut montrent des cercles qui ne se superposent pas. Les trois exemples du bas montrent des cercles qui se superposent. Notez que les deux exemples sur la droite ont un cercle de rayon 0.

La *beauté* de votre motif est la somme des **circonférences** des cercles. Il n'est pas nécessaire de produire le motif de beauté maximale, votre score sera calculé en fonction de la beauté que vous avez réussi à atteindre. Veuillez lire la section Score ci-dessous.

Sous-tâches et Contraintes

Dans chaque sous-tâche, il est garanti que :

- $1 \leq x_i, y_i \leq 1\,000\,000\,000$ pour tout i .
- Les cercles ont des centres distincts. C'est à dire, $(x_i, y_i) \neq (x_j, y_j)$ pour tout $i \neq j$.

Dans tous les fichiers tests (à l'exception des exemples), les valeurs de x_i et y_i sont choisies aléatoirement, de manière uniforme en respectant les contraintes ci-dessus.

Des contraintes supplémentaires pour chaque sous-tâche sont données ci-dessous. Chaque sous-tâche a exactement 5 fichiers tests.

Sous-tâche	Points	N
1	10	10
2	10	20
3	10	50
4	10	100
5	15	200

Sous-tâche	Points	N
6	15	500
7	15	1000
8	15	2000

Entrée

- La première ligne de l'entrée contient N .
- Suivent N lignes décrivant les centres des cercles. La i -ème ligne contient x_i et y_i .

Sortie

Affichez N lignes : la i -ème ligne doit contenir r_i , le rayon du i -ème cercle.

Score

Si deux cercles se superposent, ou si un cercle a un rayon négatif, votre score sera de 0%.

Sinon, notons OPT la beauté maximale atteignable pour le fichier test, et notons SOL la beauté de votre solution. Si $SOL = OPT$, votre score sera de 100%.

Sinon, votre score sera de $-20 \times \log_{10}(1 - \frac{SOL}{OPT})\%$ pour le fichier test (jusqu'à un maximum de 100%). En particulier:

ratio SOL/OPT	points (%)
0.5	6.02
0.6	7.96
0.7	10.46
0.8	13.98
0.9	20
0.99	40
0.999	60
0.9999	80
0.99999	100

Pour s'assurer que la sortie de votre programme est suffisamment précise, nous vous recommandons d'utiliser le type 'double' en C++ et d'afficher le rayon de chaque cercle avec 9 chiffres de précision.

Pour afficher une variable définie par double x ; sur la sortie standard avec `printf`, utilisez `printf("%.9f");`

Pour afficher la sortie avec `cout`, commencez par inclure `#include <iomanip>`. Puis, depuis la fonction `main`, et avant tout appel à `cout`, écrivez :

```
std::cout << std::fixed << std::setprecision(9);
```


Entrée d'exemple

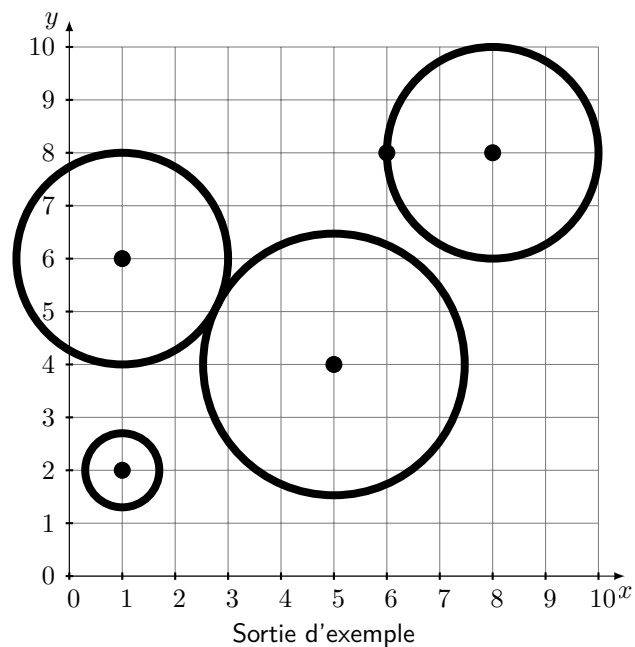
5
1 6
5 4
1 2
8 8
6 8

Sortie d'exemple

2.000000000
2.472135955
0.700000000
2.000000000
0.000000000

Explications

La beauté totale de la sortie est 45.064... , tandis que la beauté maximale pour le fichier test est de 53.232... , le score obtenu pour ce fichier test est donc de 16.28% des points.



Une solution optimale pour un fichier généré aléatoirement avec $N = 100$ est affichée ci-dessous.

