

RandomLogic

Ex1 : L'énergie du futur

Vous souhaitez intégrer la Brigade du Temps, équipe d'élite composée uniquement de méta-humains capables de voir dans le futur. En rassemblant les meilleurs méta-humains, la Brigade est d'une aide précieuse pour anticiper certains événements néfastes à l'Humanité et la Planète, et les prévoyants et en proposant des solutions optimales avant même qu'ils adviennent.

La Brigade est composée de 4 Ordres : la Prévoyance, l'Innovation, la Sagesse et la Protection. Aujourd'hui c'est le grand jour : après des mois de préparation, vous allez passer l'ultime examen pour intégrer l'Ordre de la Sagesse. Vous devrez montrer que vous êtes un méta-humain, et l'un des meilleurs.

L'Ordre de la Sagesse a été missionné par la Ville pour aider les ingénieurs à démanteler les toutes dernières centrales à charbon. Votre test consistera à prédire l'avancement des travaux sur les différentes centrales, à 4 mois dans le futur.

Chaque mois, les ingénieurs effectueront une seule opération, selon la valeur de la puissance encore en place de la centrale :

Si la puissance est un multiple de 3, diviser la puissance par 3

Sinon, si la puissance est un multiple de 2, diviser la puissance par 2

Sinon, enlever 1 à la puissance

Écrivez un programme qui calcule quelle sera la puissance encore installée au bout de 4 mois de travaux.

Données

Entrée

Ligne 1 : un entier P compris entre 1 et 1000, représentant la puissance de la centrale.

Sortie

Votre programme devra renvoyer la puissance qui sera encore en place au bout de 4 mois de travaux.

Exemples

Exemple 1

46

mois 1 : 46 est divisible par 2. La puissance au bout d'un mois sera de $46 / 2 = 23$

mois 2 : 23 n'est divisible ni par 3 ni par 2. La puissance au bout de 2 mois sera donc de $23 - 1 = 22$

mois 3 : 22 est divisible par 2. Au bout de 3 mois : $22 / 2 = 11$

mois 4 : 11 n'est divisible ni par 3 ni par 2. La puissance au bout de 4 mois sera donc de $11 - 1 = 10$

Pour cette entrée, votre programme devra donc renvoyer 10

Exemple 2

108

mois 1 : $108 \rightarrow 108 / 3 = 36$

mois 2 : $36 \rightarrow 36 / 3 = 12$

mois 3 : $12 \rightarrow 12 / 3 = 4$

mois 4 : $4 \rightarrow 4 / 2 = 2$

Votre programme devra donc renvoyer 2

Ex2 : Pr vision de temp te solaire

Comme vous le savez s rement, le plus grand danger pour une infrastructure quantique est l'arriv e soudaine d'une  ruption solaire.

Gr ce au t l scope qui tra ne au fond de votre jardin, vous avez pu observer le soleil pendant de longues journ es et avez d duit un motif tr s int ressant : les taches solaires apparaissent et disparaissent d'une mani re erratique, mais   l'approche d'une  ruption vous remarquez que certaines taches apparaissent toujours dans le m me ordre !

Afin de mieux pr voir les temp tes solaires qui menacent votre infrastructure, vous souhaitez  crire un algorithme qui prend en entr e deux observations successives, et trouver les taches solaires qui sont communes aux deux. Si ces taches figurent dans le m me ordre, on risque une  ruption imminente !

Donn es

Entr e

Ligne 1 : une cha ne S1 de lettres repr sentant le premier motif d'activit  solaire. Ligne 2 : une cha ne S2 de lettres repr sentant le second motif d'activit  solaire.

Sortie

Si les points communs entre S1 et S2 figurent dans le m me ordre entre les deux motifs, affichez deux lignes :

Sur la premi re ligne, la cha ne de caract res "TEMPETE"

Sur la suivante, les caractères communs entre S1 et S2, dans l'ordre d'apparition

S'il n'y a pas de points communs ou que ceux-ci ne sont pas dans le même ordre, affichez NORMAL sur une ligne.

Contraintes

Les caractères sont uniques au sein de chaque chaîne.

Les caractères sont des lettres majuscules ou minuscules.

Exemple

Pour l'entrée :

SmXwpKz

meXZKi

Trois lettres apparaissent en commun : m, X et K. Comme elles apparaissent dans l'ordre mXK dans S1 et S2, une tempête est imminente et vous devez afficher :

TEMPETE

mXK

Notez que les taches solaires z et Z sont considérées comme différentes.

Exemple 2

Pour l'entrée :

LMepmWsAO

kMrTpOmzIf

Ici, les éléments communs sont M, p, m et O. Cependant, ils apparaissent dans S1 dans l'ordre MpmO et dans S2 dans un autre ordre MpOm. Ainsi, il n'y a pas d'activité anormale et vous devez afficher :

NORMAL

Ex3 : La meilleure protection

Vous vous faites vite remarquer au sein de la Brigade. Depuis vos exploits sur le précédent challenge, vous avez été un des éléments les plus prometteurs de l'Ordre. Mais durant vos missions, vous avez remarqué qu'un méta-humain très puissant va plus vite que vous, et anticipe encore mieux que la Brigade certaines catastrophes...

Vous n'en parlez pas encore à votre direction, mais vous lancez vos recherches sur ce mystérieux bienfaiteur inconnu de la Brigade. Dans votre quête, vous remarquez un motif étrange : il semblerait que celui-ci se déplace en portant secours à ceux qui en ont le plus besoin.

Pour commencer vos recherches, vous décidez de vous concentrer uniquement sur le projet de protection des villes spatiales. Ces villes sont composées de 2 types de modules : les modules d'habitation et les modules de contrôle, et chaque module peut en protéger un ou plusieurs autres. Il n'y a jamais de cycles dans la relation de protection (par exemple, si A protège B et B protège C, alors C ne protège pas A).

Les modules protégeant au moins un autre module sont des modules de contrôle, et ceux n'en protégeant aucun autre sont des modules d'habitation. Trouvez, dans la ville, le contrôleur contrôlant le plus de modules d'habitation pour connaître le quartier pouvant protéger le plus d'habitants à la fois.

Données

Entrée

Ligne 1 : Deux entiers N et V indiquant combien il y a de relations de protection et combien de modules (numérotés de 0 à V-1)

N lignes suivantes : Deux entiers a et b indiquant que le module a protège le module b

$$1 \leq N < V \leq 11\,000$$

Sortie

Ligne 1 Un entier indiquant le contrôleur gérant le plus de modules d'habitation.

Exemple

Pour l'entrée :

8	10
8	4
7	2
6	8
6	1
7	9
6	7
2	0
5 3	

La solution est :

6

La première ligne indique 8 relations et 10 modules numérotés de 0 à 9. Ici le module 6 contrôle 4 modules d'habitations (0, 1, 4, 9), alors que le contrôleur 5 n'en contrôle qu'un seul (3). C'est donc 6 qu'il faut répondre.

<https://freeimage.host/i/graphviz.HCkDTZX>

Notons qu'il existe toujours une unique solution possible dans les entrées proposées.

Ex4 : ChronoCoin

Vous avez très rapidement monté les échelons de la Brigade du Temps et vous vous retrouvez vite au commandement central. Vous avez été affecté à la lutte contre le ChronoGang, un groupe de méta humains qui utilisent leurs méta-sens criminellement.

Pour analyser leurs actions, vous vous penchez sur les mouvements financiers des comptes en banque liés au groupe. Depuis vos visions, vous voyez que les transactions sont plus ou moins fréquentes : vous vous intéressez bien évidemment aux transactions les plus fréquentes, mais vous pressentez que les actions les moins fréquentes cachent aussi quelque chose.

A partir d'une liste de transactions, trouvez la transaction la plus fréquente, et la moins fréquente. En cas d'égalité, prendre la plus grande des plus fréquentes et la plus petite des moins fréquentes.

Données

Entrée

Ligne 1 : N entiers (représentant les transactions) compris entre 1 et 99 séparés par des espaces, avec $2 \leq N \leq 99$.

Sortie

Votre programme devra renvoyer l'entier le plus fréquent moins l'entier le moins fréquent dans la liste. En cas d'égalité, prendre le plus grand des nombres apparaissant le plus de fois et le plus petit de ceux apparaissant le moins.

Attention : le nombre en sortie peut être négatif.

Exemples

3 2 1 2 1 1

Votre programme devra renvoyer pour cet exemple -2 : 1 est le nombre le plus fréquent et 3 le moins fréquent, d'où $1 - 3 = -2$

1 2 2 3 3 4

1 et 4 apparaissent une fois, 2 et 3 apparaissent 2 fois. On prend donc le minimum de 1 et 4 et le maximum de 2 et 3, d'où la sortie attendue $3 - 1 = 2$

Ex5 : Opération discrétion

En préparation pour votre raid sur la planque d'un gang de chrono-criminels, vous devez faire appel à des équipes de mercenaires pour vous assurer d'être en effectif suffisant.

Mais attention : impossible de recruter tout le monde, cela risquerait d'attirer l'attention des criminels, et vous risqueriez aussi d'exploser le budget de la brigade ! Il va donc falloir jouer finement, en choisissant bien les équipes de mercenaires que vous allez contacter.

Pour éviter que vos adversaires ne puissent vous surprendre en faisant appel à des mercenaires, vous souhaitez avoir strictement plus de la moitié des combattants disponibles. Ainsi, même si les criminels recrutent tous les mercenaires disponibles restants, vous serez toujours en supériorité numérique. Les mercenaires sont organisés en groupes, et vous cherchez à savoir quels groupes vous devrez recruter.

Votre brigade dispose déjà de quelques officiers pour le raid, ce qui vous permet de partir avec une base solide d'effectifs. Combien de mercenaires au minimum devrez-vous embaucher pour vous assurer d'avoir l'avantage pendant le raid ?

Données

Entrée

Ligne 1 : un entier N compris entre 0 et 1 000 000 représentant le nombre d'effectifs disponibles de votre brigade.

Ligne 2 : un entier M compris entre 0 et 100 représentant le nombre d'équipes de mercenaires disponibles.

M lignes suivantes : M entiers compris entre 0 et 1 000 représentant la taille de chaque groupe de mercenaires.

Sortie

Votre programme devra afficher le nombre minimum de mercenaires à recruter pour vous assurer d'avoir l'avantage durant le raid.

Exemple

247

7

178

168

66

158

60

113

227

Vous disposez de 247 officiers, auxquels vous pourrez ajouter 385 mercenaires ($227 + 158$) pour un total de 632 personnes de votre côté.

Dans ce cas, les criminels pourront recruter au maximum les 585 mercenaires restants ($178 + 168 + 66 + 60 + 113$), ce qui vous place en supériorité numérique.

385 est le nombre minimum de recrutements qui vous permettra d'être en majorité, votre code devra donc afficher 385.

Ex6 : Indices temporels

Tout le monde le sait à la Brigade, vous êtes un des plus brillants agents des rangs. A tel point que l'on vous a confié l'enquête sur la Chrono-Alliance : ce gang de méta-humains terrorise la Cité et arrive à brouiller les visions des agents de la Brigade en ajoutant des traces temporelles.

Jusque-là, personne ne semblait pouvoir trouver le moindre indice sur eux, mais vous pensez avoir une piste : les traces qu'ils laissent semblent créer des états temporels impossibles. A partir des indices que vous trouvez sur les lieux de leurs crimes, vous remarquez effectivement qu'à partir d'un moment la liste d'indices ne peut plus correspondre à une réalité possible.

Les états sont représentés par 5 voyants, pouvant être de 3 couleurs (rouge R, vert V ou bleu B). Les indices que vous trouvez sont flous : ils vous donnent 3 conditions dont au moins une est vraie mais pas forcément toute. Par exemple, l'indice R1 B2 V2 vous indique que le premier voyant est rouge, ou que le deuxième voyant est bleu, ou que le deuxième voyant est vert.

Vous examinez les indices dans l'ordre : vous cherchez d'abord si un état des 5 voyants est possible avec le premier indice. Si oui, vous cherchez si un état est possible avec le premier ET le second indice, et ainsi de suite jusqu'à trouver le premier indice qui, s'il est pris en compte, rend impossible

Écrivez un programme qui prend en entrée une liste d'indices, et renvoie l'indice maximum à partir duquel il existe toujours un état possible.

Données

Entrée

Ligne 1 : le nombre N d'indices, avec $8 \leq N \leq 30$ Lignes 2 à N+1 : la liste d'indices au format avec 3 conditions séparées par des espaces, chaque

condition représentée par une couleur sur une lettre (R, V ou B), et un entier entre 1 et 5 donnant le numéro du voyant.

Par exemple, l'indice R1 B2 V2 vous indique que le premier voyant est rouge, ou que le deuxième voyant est bleu, ou que le deuxième voyant est vert.

Sortie

Un entier représentant le nombre maximum d'indices d'affilés qui peuvent représenter un état des 5 voyants.

Note : si une combinaison de voyants existe qui satisfait tous les indices, votre programme devra renvoyer N.

Exemples

8

R5	R3	R1
B1	V5	V5
B3	R5	B3
V1	V1	B4
V4	V1	R4
V1	R5	V3
R2	B1	R2
R3 B3 R4		

La combinaison BVVBR satisfait les 4 premières conditions. En effet :

R5 est satisfait pour le premier indice

B1 pour le second

R5 pour le troisième

B4 pour le quatrième

En revanche, il n'existe aucune combinaison satisfaisant les 5 premiers indices. Votre programme devra donc renvoyer 4

Ex7 : Espionnage du futur.

Pour espionner les faits et gestes des chrono-criminels vous lancez dans le passé des sondes spatiales automatisées. Chaque sonde suit un parcours infini répétitif, et attendra peut-être votre objectif (un chrono-parrain) connu dans un lointain futur. À vous de trouver à quel moment, mais attention, cela peut être dans très longtemps !

Données

Entrée

Ligne 1 : 2 entiers séparés par des espaces indiquant les coordonnées à atteindre X et Y, sachant que les coordonnées de départ de la sonde sont 0,0

Ligne 2 : Une chaîne de caractères composées uniquement des lettres URDL. C'est la suite des étapes du parcours de la sonde. R correspond à X+1 (déplacement vers la droite), L à X-1 (vers la gauche), U à Y+1 (vers le haut) et D à Y-1 (vers le bas).

Les valeurs numériques sont limitées à un entier signé sur 50 bits.

La suite des étapes est au plus de taille 5000, quand la sonde arrive au bout elle recommence sans fin au début de la suite.

Sortie

Votre programme devra renvoyer sur la sortie standard un entier correspondant au numéro de la première étape dans le futur à laquelle elle arrive à la destination. Ainsi on renverra 0 si l'objectif est égal au point de départ, 1 si on l'atteint après le premier déplacement, etc... S'il n'y a pas de solution, on utilise la chaîne not possible en sortie.

Note : le résultat est un entier positif ou nul limité à 62 bits, et peut donc potentiellement être très grand.

Exemples

Entrée

21

43

UUURRRDLUU

Sortie

104

Le motif répété est le suivant: 3 étapes vers le haut, 3 étapes vers la droite, 1 vers le bas, 1 vers la gauche puis 2 vers le haut. En partant de (0,0), en répétant 10 fois le motif de 10 étapes et en faisant 4 étapes sur le onzième motif on arrive pour la première fois en (21, 43), à l'étape 104.

Ex8 : Terrain glissant

Vous avez trouvé la planque de la Chrono-Alliance !! Vous aviez vu juste avec les indices, ils vous ont directement mené à eux. Vous mobilisez votre équipe et foncez les arrêter. Vous plongez dans l'Ether, espace que seuls les méta-humains peuvent pénétrer. L'Alliance s'y cache !

L'Ether a une particularité : vous pouvez vous déplacer dans toutes les 4 directions de cet espace à 2 dimensions, mais une fois une direction choisie il n'y a que 2 moyens de s'arrêter. Si votre chemin passe par un Grip temporel, votre chemin s'arrêtera dessus et vous pouvez repartir dans les 4 directions. Il existe également des Obstacles temporels, qui sont impénétrables et qui arrêteront votre chemin.

A partir de la carte de l'Ether, trouvez le chemin le plus court entre vous et la planque de l'Alliance !

Données

Entrée

Ligne 1 : deux entiers séparés par un espace, H et W la hauteur et la largeur de la grille

H lignes suivantes : Une ligne de la grille de l'Ether comportant exactement W caractères.

est une obstacle temporel (zone infranchissable)

. est une zone libre (zone permettant le mouvement)

o est un grip temporel (on doit s'y arrêter lorsqu'on passe dessus)

P est votre position initiale (la case est une zone libre)

X est la planque de l'Alliance (la case est une zone libre)

H et W sont compris entre 5 et 100

L'Ether est toujours entouré par un bord d'obstacles infranchissables.

Sortie

Votre programme devra afficher le nombre minimum de mouvements nécessaires pour parvenir à la planque. Cette valeur existe toujours.

Exemple

Entrée

```
5
#####
#..#.#
#..Xo.#
#P#o.##
#####
```

Sortie

7

Ici le chemin le plus court pour parvenir à la planque consiste à aller d'abord vers le haut (on glisse de 2 cases) puis vers la droite (1 case), puis vers le bas (1 case). On a déjà fait 3 mouvements. On doit alors se diriger vers la droite. On passe sur la planque mais attention, c'est une zone libre donc on ne s'y arrête pas (raté !) mais on s'arrête sur le grip temporel sur la case suivante. On va ensuite vers le bas (1 case), vers la gauche (1 case sur le grip), et enfin on va vers le haut pour s'arrêter sur la planque. Nous avons eu donc besoin de 7 mouvements pour arriver à destination, il n'existe pas de chemin plus court depuis la position initiale.

Ex9 : On avait dit 3 parts

Après la très fructueuse opération de démantèlement de la Chrono-Alliance, votre brigade mérite bien une célébration !

À l'occasion de votre remise du prix de bravoure et de service de la brigade, votre chef a préparé un magnifique gâteau et c'est à vous de faire le service.

Comme le veut la tradition, la première découpe doit se faire à la verticale et découper le gâteau en trois parts égales. Cette découpe est facile à accomplir sur une forme simple, mais le chef a pris quelques libertés créatives sur la forme du gâteau, et vous avez devant vous un amas informe recouvert de glaçage...

Pour ne pas décevoir votre équipe et impressionner le chef avec vos capacités de découpe hors norme (après tout, c'est vous qui avez été désigné), vous souhaitez écrire un programme qui calcule l'emplacement précis des deux découpes, à partir de la forme du gâteau.

<https://iili.io/H0EK30u.md.png>

Vous remarquez que le contour du gâteau peut être modélisé selon deux lignes brisées (bord supérieur et bord inférieur) dont les abscisses sont strictement monotones. Le bord supérieur comporte uniquement des ordonnées positives, et le bord inférieur seulement des ordonnées négatives. Les deux points extrêmes à gauche et à droite sont donc sur l'axe horizontal.

À quel endroit devez-vous effectuer les découpes pour séparer le gâteau en trois parts de superficie identique ?

Données

Entrée

Ligne 1 : Un entier A : le nombre de points sur le bord supérieur.

Les A lignes suivantes : Deux entiers X et Y représentant les coordonnées d'un point sur le bord supérieur du gâteau.

Ligne 1 : Un entier B : le nombre de points sur le bord inférieur.

Les B lignes suivantes : Deux entiers X et Y représentant les coordonnées d'un point sur le bord inférieur du gâteau.

Les coordonnées des deux bords sont données de gauche à droite (coordonnée X croissante), et leurs deux extrémités seront identiques.

Les coordonnées sont comprises entre -10^6 et 10^6 . Il y a au maximum 1000 points sur chaque bord.

Sortie

Votre programme devra afficher l'abscisse des deux découpes, une sur chaque ligne.

Les deux nombres sont à afficher sous forme décimale, une tolérance absolue de 10^{-6} sera appliquée.

Exemple

L'exemple ci-dessous correspond à l'image présentée en début d'énoncé.

Entrée

5	
2	0
6	4
7	3
10	5
12	0
4	
2	0
4	-4
11	-4
12 0	

Sortie

6.10485381178

8.85920709431

Note : les résultats ci-dessus sont donnés avec une précision plus élevée que nécessaire. Votre code peut afficher autant de décimales que vous le souhaitez, tant que votre réponse se situe dans la tolérance de 10^{-6} .