

C. Team Coding

Nom du problème	Team Coding
Limite de temps	4 secondes
Limite mémoire	1 gigaoctet

L'entreprise "Eindhoven Gigantic Open-Source Institute" (EGOI) est structurée de manière très hiérarchique. A part la chef d'entreprise Anneke, chacun des autres $N - 1$ employés de l'entreprise a un unique chef, et la hiérarchie est acyclique. Vous pouvez assimiler la hiérarchie de l'entreprise à un arbre enraciné en un sommet correspondant à Anneke. Comme c'est une entreprise diversifiée, les employés codent dans K langages de programmation différents, mais chaque employé a exactement un langage de programmation favori.

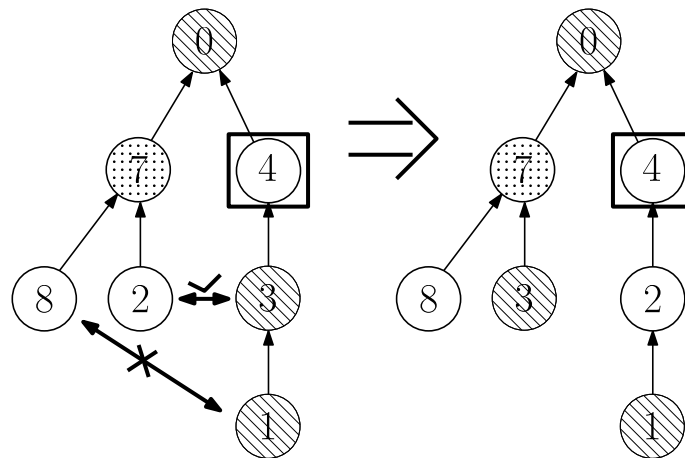
Anneke a un grand nouveau projet de travail pour une équipe de son entreprise. Elle veut investir autant de ressources que possible dans ce projet. Pour décider quelle équipe travaillera sur ce projet, elle va :

1. Choisir une personne pour diriger l'équipe. Cela définira également le langage de programmation dans lequel sera codé le projet. Chaque employé qui est dans le sous-arbre en-dessous de la chef d'équipe, et qui a le même langage de programmation favori qu'elle, travaillera sur le problème.
2. Augmenter le nombre d'employés qui travaillent sur le projet en mettant dans l'équipe les employés qui ont le même langage de programmation favori que la chef d'équipe, en faisant des échanges.

Pour maximiser le nombre d'employés qui travailleront sur le projet, elle peut effectuer l'opération d'échange suivante un nombre quelconque de fois :

1. Elle choisit deux employés :
 - Un employé qui est actuellement dans le sous-arbre de la chef d'équipe et n'a pas le même langage de programmation favori que la chef d'équipe.
 - Un employé qui n'est pas actuellement dans le sous-arbre de la chef d'équipe, et qui a le même langage de programmation favori que la chef d'équipe. Additionnellement, cet employé doit être au même niveau que l'autre employé choisi ; ce qui signifie qu'ils ont le même nombre de chefs dans la chaîne de rapports jusqu'à Anneke. Si vous assimilez la hiérarchie de l'entreprise à un arbre, alors les deux employés sont à la même profondeur dans l'arbre.

2. Ces deux employés (et *seulement* eux – pas d'autres employés) échangent leurs positions dans la hiérarchie de l'entreprise. Notez que les employés qui ont pour chef direct un des deux employés affectés restent en place et changent juste de chef. Dans l'exemple ci-dessus, avec l'employé 4 choisi comme chef d'équipe, nous pouvons échanger les employés 3 et 2 mais pas les employés 1 et 8.



Trouvez le nombre maximum d'employés travaillant sur le nouveau projet que vous pouvez atteindre et le nombre minimum d'opérations d'échanges nécessaires pour atteindre cela.

Entrée

La première ligne de l'entrée contient deux entiers, N et K , le nombre d'employés de EGOI et le nombre de langages de programmation que les employés utilisent éventuellement.

Les employés de EGOI sont numérotés de 0 à $N - 1$, et Anneke la chef d'entreprise a le numéro 0. La ligne suivante contient N entiers ℓ_i avec $0 \leq \ell_i < K$, le langage de programmation favori des employés.

Les $N - 1$ lignes suivantes contiennent la hiérarchie de l'entreprise. La i ème ligne contient un entier b_i avec $0 \leq b_i < N$, le chef direct du i ème employé. Notez que i va de 1 à $N - 1$ (inclus), comme Anneke, la chef d'entreprise, n'a pas de chef.

Sortie

Renvoyez une unique ligne avec deux entiers, P et S , le nombre maximum d'employés (en incluant le chef d'équipe) travaillant sur le nouveau projet que vous pouvez atteindre avec n'importe quel nombre d'échanges, et le nombre *minimum* d'échanges nécessaires pour atteindre cela.

Contraintes et Répartition des points

- $1 \leq N \leq 10^5$.
- $1 \leq K \leq N$.

Votre solution sera testée sur un ensemble de sous-tâches, chacune rapportant un certain nombre de points. Chaque sous-tâche contient un ensemble de tests. Pour avoir tous les points d'une sous-tâche, vous devez valider tous les tests de la sous-tâche.

Sous-tâche	Points	Limites
1	12	Le chef direct de l'employé i est $i - 1$ pour tout $1 \leq i < N$.
2	19	$K \leq 2$
3	27	Pour chaque langage de programmation, il y a au plus 10 employés qui le préfèrent.
4	23	$N \leq 2\,000$
5	19	Pas de contraintes supplémentaires

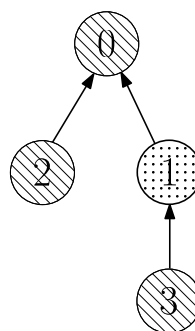
Exemples

Dans les deux premiers exemples, la hiérarchie de l'entreprise est comme suit, où chaque motif représente un langage de programmation (0="à rayures", 1="à points", 2="uni") :

Graph for example 1

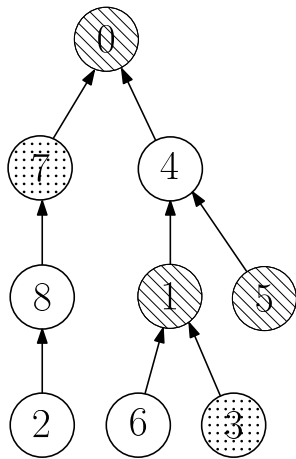


Graph for example 2

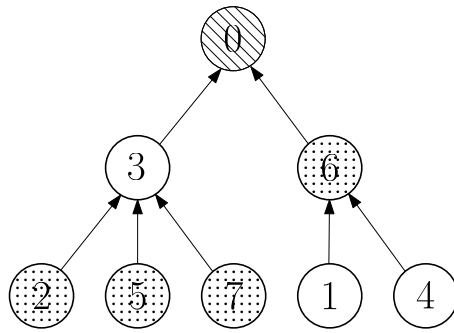


Dans l'exemple 1, nous pouvons choisir l'employé 1 comme chef d'équipe avec l'employé 4 préférant le même langage de programmation, et il n'y a pas d'échanges possibles pour améliorer cela. Dans l'exemple 2, l'entreprise entière a 3 employés préférant le langage 0, qui est aussi le langage favori d'Anneke, donc choisir Anneke comme chef d'équipe donne une équipe de taille 3 où aucun échange n'est nécessaire.

Graph for example 3



Graph for example 4



Dans l'exemple 3, nous choisissons l'employé 4 comme chef d'équipe et après nous pouvons faire changer d'équipe les employés 1 & 8 et 2 & 3 pour avoir un total de 4 employés ayant le même langage favori que 4, c'est-à-dire le langage 2 (uni). Dans l'exemple 4, le score maximal peut être obtenu en choisissant l'employé 6 comme chef d'équipe et en échangeant les employés 4 & 7 et 1 & 5. Notez que l'on ne peut pas échanger les employés 6 & 3 avant de choisir le chef d'équipe, pour obtenir un score de 4, parce que nous devons choisir le chef d'équipe d'abord.

Entrée	Sortie
<pre> 5 3 0 1 2 2 1 0 1 2 3 </pre>	<pre> 2 0 </pre>
<pre> 4 2 0 1 0 0 0 0 1 </pre>	<pre> 3 0 </pre>
<pre> 9 3 0 0 2 1 2 0 2 1 2 4 8 1 0 4 1 0 7 </pre>	<pre> 4 2 </pre>
<pre> 8 3 0 2 1 2 2 1 1 1 6 3 0 6 3 0 3 </pre>	<pre> 3 2 </pre>