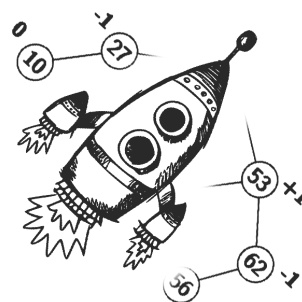


Úloha č. 2

Posádka



Odpověz Sfinze!

10 b

Tato úloha je vyhodnocována automaticky. Je potřeba, aby výstup programu **přesně** korespondoval se specifikací výstupu níže. Jak odevzdávat tento typ úloh se můžeš dočíst na webových stránkách FIKSu pod záložkou „Jak řešit FIKS“.

Po úspěšném testovacím letu, se kterým jste se jako tým tak tak vešli do časového limitu (a málem při tom shodili plexisklový obal rektorátu), bylo třeba začít řešit další z velmi ožehavých problémů, a to sestavení možných posádek pro soutěž. První pokusy o sestavení posádek již proběhly, ale vždycky s nimi bylo něco špatně. Jednou se to někdo snažil rozhodnout takříkajíc v ruce, podruhé to spočítal stroj pamatující dost možná Alana Turinga. Vždy se tam ale vloupal nějaký zlotřilý bug a udělal v tvorbě posádky zmatek.

Člen posádky má nějakou počáteční kompetentnost a může mít přímé nadřízené. Každý člen má také právě jednoho přímého nadřízeného, pokud avšak není velitelem posádky (velitel posádky je právě jeden). Dále se můžete spolehnout, že žádný člen není přímým ani nepřímým nadřízeným sám sobě. Máte k dispozici určitý počet dovednostních bodů, které můžete přerozdělit mezi jednotlivé členy posádky. Při použití jednoho dovednostního bodu zvýšíte dotyčnému členovi kompetentnost o 1 a spotřebujete tím 1 dovednostní bod. Vaším úkolem je využít dovednostní body tak, že posádky budou stabilní (nebo detekovat, že to nelze) a při tom maximalizovat kompetentnosti těchto posádek (kompetentnost posádky je rovna kompetentnosti nejméně kompetentního člena posádky). Člen posádky zpochybňuje nějakého svého přímého nebo nepřímého nadřízeného, pokud není kompetentnost nadřízeného ostře větší. Posádka, kde nějaký člen zpochybňuje nadřízeného, by nebyla stabilní.

Vstup

Na prvním řádku vstupu se nachází přirozené číslo t ($1 \leq t \leq 1000$) – počet posádek, pro které máte najít řešení. Následují informace o jednotlivých posádkách. Každá informace o posádce má tři řádky v následujícím formátu:

Na prvním řádku jsou dvě celá čísla n a k ($1 \leq n \leq 2 \cdot 10^4$, $0 \leq k \leq 10^{18}$) – počet členů posádky a počet dovednostních bodů, které máte k dispozici.

Na druhém řádku se nachází n mezerou oddělených celých čísel a_i ($0 \leq a_i \leq 10^{18}$, $1 \leq i \leq n$) – i -té číslo reprezentuje počáteční kompetentnost i -tého člena posádky.

Na třetím řádku se nachází n mezerou oddělených čísel p_i ($1 \leq p_i \leq n$, $1 \leq i \leq n$) – i -té číslo reprezentuje přímého nadřízeného i -tého člena posádky (nebo $p_i = -1$ pokud je i -tý člen posádky velitelem posádky).

Výstup

Pro každou z t posádek vypište na samostatný řádek následující: Pokud lze dovednostní body přerozdělit tak, že lze vytvořit stabilní posádku, pak vypište největší kompetentnost posádky, které lze dosáhnout. Pokud stabilní posádku vytvořit s danými prostředky nelze, pak vypište ajajaj.

Vstup

Výstup

3
ajajaj
7
2
1
2
1
6
3
1000000002147483648

Pokud váš program tedy nestihne dopočítat všechny vstupy, tak odevzdejte (ve stejném pořadí jako ve vstupním souboru) alespoň část výstupů, které dopočítat stihl.

Varování

Pokud programujete v jazyku, který neumí automaticky počítat s neomezeně velkými celými čísly, dejte pozor, že některá čísla, která se mohou ve vstupních a výstupních datech vyskytnout, mohou být na datový typ `int` moc velká. V takových případech je (například v C a C++) možné využít datový typ `long long`. Naopak třeba Python s velkými celými čísly pracuje bez problému a není potřeba tento problém řešit.