**Анализ на решението на задача Експеримент**

**НАЦИОНАЛНА ЛАГЕР-ШКОЛА ПО ИНФОРМАТИКА  
Смолян, 31 август – 5 септември 2019г.  
Контролно състезание, 8 клас**

**Задача. Експеримент**

Международният експериментален термоядрен реактор ITER разположен в югоизточна Франция се счита за най-сложно устроеното съоръжение в човешката история. Една причина за това са неговите огромни системи за диагностика и анализ. Те използват множество сензори разположени на различни места в реактора, отчитащи температура, плътност на плазмата и други.

Сензорите са направени така, че да няма два сензора подаващи една и съща стойност. Също така сензорите подават сигнали само в диапазона [1, K], където K е броят на сензорите. Тоест когато подредим стойностите отчетени от всеки сензор в редица, ще получим пермутация на числата от 1 до K.

Сигналите се предават чрез междинни компютри, докато стигнат до главния. Сензорите и междинните компютри могат да се свържат към точно един компютър. Към сензорите не може да се свърже нищо. Главният компютър не се свързва към нищо друго. Стойността във всеки компютър се образува като се вземе или най-малката или най-голямата измежду стойностите във всяко устройство свързано към него.

Задачата ви е при дадена конфигурация на компютрите и сензорите да определите максималната стойност, която може да пъде получена в главния компютър.

**Вход**

На първия ред е дадено числото N – общия брой устройства, вкл. главния и междинните компютри и сензорите, номерирани от 1 до N. Главния компютър е с номер 1.

Следва ред с N числа f1, f2, ..., fN като fi =1, ако устройството с номер i взима **максималната** стойност или 0, ако взима минималната. Ако устройството е сензор просто игнорирайте това число.

На последния ред са дадени N-1 числа p2, p3, ..., pN – те показват номера на компютъра към който е свързано съответното устройство.

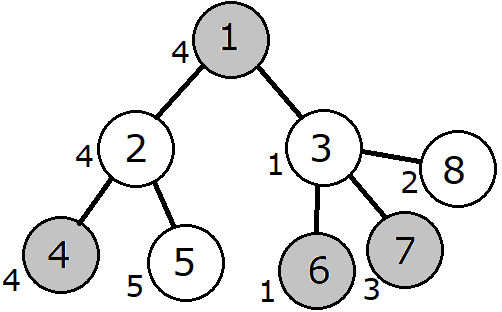
**Изход**  
 На един ред на изхода изведете максималната стойност която може да се получи в главния компютър.

**Ограничения**  
 2 ≤ N ≤ 200000,

1 ≤ pi ≤ i – 1, за всяко 2 ≤ i ≤ N

**Примери**

|  |  |
| --- | --- |
| Вход | Изход |
| 8  1 0 0 1 0 1 1 0  1 1 2 2 3 3 3 | 4 |
| 9  1 1 0 0 1 0 1 0 1  1 1 2 2 3 3 4 4 | 5 |

 **Пояснение**:  
В първия пример сензорите са с номера 4,5,6,7,8. Междинните компютри са 2 и 3. В сиво са оцветени върховете, които взимат максимума, а в бяло – минимума. Вляво от върховете са дадени техните стойности. При дадените стойности на сензорите компютър 2 взима по-малкото от числата 4 и 5, т.е. стойността му става 4. Компютър 3 взима 1, защото е май-малкото измежду 1, 2 и 3. Накрая Главния компютър ще вземе стойността в компютър 2, тъй като е по-голяма от тази в номер 3.

От условието следва, че разглеждаме кореново дърво.

Ще разгледаме две решения на задачата, базирани на различни методи.

Първото решение се базира на динамично оптимиране. За да проверим дали дадено число x е решение, можем да заместим числата в листата по-малки от него с 0, а тези по-големи или равни с 1. Тогава ако стойността в корена е 1, то имаме възможно решение. Това обаче не може да се реши с двоично търсене, тъй като не всички отговори са постижими – зависи от дървото. Ще направим следното: с dp[i] ще означим минималния брой единици в листата на поддървото с корен i, които са нужни за постигане на единица във върха i. Като при операция с максимум ще взимаме най-малкото от dp[v], където v е дете на i. А при операция с минимум ще взимаме сумата на dp[v]. Крайният отговор е K+1–dp[1].

Другото решение, може би по-лесното, е да се приложи алчна стратегия. Нека с leaf[u] означим броя на листата в поддърво с корен u, а с ans[u] – максималната стойност, която можем да получим във връх u, ако го разгледаме самостоятелно.

Разглеждаме двата варианта за операция в u:

1. Максимум(f[u]=1) – намираме минималната разлика mindiff=min{leaf[v]-ans[v]} измежду всички деца v на u. Отговорът за u е ans[u]=leaf[u]-mindiff. Доказва се, че това е оптималното разпределение.
2. Минимум(f[u]=0) – ще получим оптималното разпределение, ако във всяко поддърво с корен v номерираме първите ans[v]-1 листа, както сме намерили отговора за v. Тогава както и да разместим останалите листа, винаги отговорът в u ще е: ans[u]=1+∑(ans[v]-1) за всички деца v на върха u.

Следвайки тази стратегия ще намерим крайния отговор ans[1].