КОНТРОЛНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

Група С

Анализ на задача Дестинации

Първа подзадача. Разглеждаме всички възможни преразпределения на билетите и за всяко от тях проверяваме дали е изпълнено $a_i \geq b_i$. Ако е, намираме сумата от разликите и взимаме това решение с най-малка разлика.

Сложност – **O(N! × N).**

Втора подзадача. Ще подобрим пълното изчерпване с помощта на динамично с побитови маски. Нека **dp[n][mask]** е минималната възможна сума, която можем да постигнем, ако разглеждаме първите n участника и билетите, които вече са заети са сетнати с единица в mask. Отговора ще получим като разгледаме всички билети, които може да получи n-тия участник.

Сложност – **O(2^N × N^2).**

Трета подзадача. Ще се възползваме от допълнителното ограничение, като за всяка двойка участници сметнем разликата, която би се получила, ако те разменят билетите си.

Сложност – **O(N^2).**

Четвърта подзадача. Знаейки, че всеки участник или задържа билета си, или получава този с една позиция по-напред в пермутацията му, може лесно да съобразим, че от нас се иска колкото се може по-малко участници да разменят билети помежду си. Нека да сведем задачата до малко по-стандартна. Ако **i**-тия участник вземе билета за по-добрата дестинация, то притежателя на този билет ще направи същото. Това ще се случи и с притежателя на по-добрия билет на притежателя. Така можем да построим граф, в който върхове са участниците и има насочено ребро от участник **u** към участник **v** ако притежателя на по-добрата дестинация на **u** е **v**. От върховете на участниците, получили любимата си дестинация не излизат никакви ребра, а от всички останали излиза по точно ребро. Затова може да съдим, че графът е съставен само от пръчки и цикли. Нас ни интересуват именно циклите, защото при пръчките, върха от който не излизат ребра няма как да получи по-добра дестинация, от любимата му. Решенията са всички цикли с най-малко върхове в тях.

Сложност – **O(N).**

Пета подзадача. Подобно на четвърта подзадача, отново ще строим граф. Този път обаче от връх **u** излизат ребра към всички притежатели на по-добри дестинации в пермутацията на **u**. Също така тези ребра имат тегла – съответните разлики в позициите. Тук графът няма свойствата на този от предишната подзадача, но идеята е сравнително ясна – трябва да намерим цикъла с най-малка сума на теглата. Да предположим, че реброто между **u** и **v** участва в него. Тогава остатъка от цикъла е някакъв път от **v** към **u**, и не какъв да е път, а най-късият. С помощта на алгоритъма на Дейкстра можем да намерим най-късия път от всеки връх до всеки друг, пускайки го **N** пъти, и да намерим цикъла с най-малка дължина. Как да открием и кои върхове участват в цикъла? Ако най-късото разстояние от **v** към **u** е **dist[u]**, то предпоследни върхове, лежащи на този път може да са само онези, за които сумата от най-краткото разстояние до тях плюс тежестта на реброто от тях до **u** е равна на **dist[u]**. Така рекурсивно можем да намерим някой от най-кратките пътища.

Сложност – **O(N × (N+M) × log(M)).**

Шеста подзадача. Можем да подобрим сложността от пета подзадача, като вместо алгоритъм на Дейкстра, използваме алгоритъм на Флойд за намиране на най-кратък път от всеки връх до всеки друг.

Сложност – **O(N^3).**

Автор: Александър Гатев