**Persistence**

(Решение)

Задачата “Persistence” беше донякъде подобна на “Closest“ от пролетния турнир 2015-та година. Реално, решението отново е рекурсивно изчерпване (backtrack) като отново трябва да направим сходно наблюдение (там, че можем да ползваме множимите в нарастващ ред, тук, че цифрите на числото ще са в ненамаляващ ред (освен за P = 1, когато отговорът е 10)).

Това е известен проблем, и при наличие на интернет състезателите биха могли веднага да намерят всички отговори (<https://en.wikipedia.org/wiki/Persistence_of_a_number>). За щастие, на състезанието (поне на теория) състезателите нямат достъп до интернет.

**Наивен брутфорс**

За малки P-та минималното число е относително малко (за P < 10 най-големият отговор е под 30 милиона). Тъй като проверката каква е устойчивостта на дадено число е относително бързa, тестовете с P < 10 можеха да бъдат хванати с проста итерация от 0 нагоре, докато стигнем до число с търсената устойчивост. За това решение бяха предвидени 50 точки.

**Наивен брутфорс + запаметяване на отговорите**

Ако сме малко по-хитри, можем да забележим, че възможните входове са много малко – едва 12. Можем да пробваме да изчислим на нашия собствен компютър отговорите за тях, да ги вкараме в кода, и после да отговаряме константно. Така бихме могли да намерим и отговора за P = 10, който е малко под 4 милиарда (и бихме имали нужда малко под минута за да го намерим с наивното решение). Това би ни донесло още 20 точки.

**Умен брутфорс**

За последните 30 точки (тестовете с P = 11) трябва да измислим малко по-умно решение. Наблюдението, което ни трябва, всъщност не е никак сложно (всъщност, вглеждайки се в отговорите за P ≤ 10 бихме видели почти веднага). Числа с еднакви цифри в различен ред имат еднакво P, но минималното е това, в което цифрите са сортирани. Изключение правят числата, които имат нула, тъй като тя не може да е водеща. Както се оказва, обаче, ако числото има нула, то още на втората стъпка става едноцифрено (0), съответно единствените случаи, в които имаме нула, са за P = 0 (0) и за P = 1 (10). За останалите числа (тези без нула), наистина можем да видим, че, реда на цифрите няма значение за „устойчивостта“, тъй като умножението на цифрите е асоциативна операция. Например, 679, 697, 769, 796, 967, и 976 всички имат еднакво P = 5.

Така можем да генерираме само числа, при които цифрите са в ненамаляващ ред. Това можем да направим с прост бектрек (рекурсия). Лесен начин, по който можем да сметнем горна граница за това колко операции ще направим при това решение, е да забележим, че във всяка позиция на числото можем или да имаме цифрата от предходната позиция, или нова (по-голяма) цифра. Тъй като е гарантирано, че отговорът ще е по-малък от 1018, тоест има до 18 цифри, това са 2^18 възможности за смените (реално много по-малко, тъй като не можем да сменим 12 пъти цифрата – можем максимум 8 (1->2, 2->3, …, 8->9). Но дори 2^18 е относително малко число – едва четвърт милион числа, чиято устойчивост трябва да сметнем.

Така, за всяко число с ненамаляващи цифри, смятаме устойчивостта и го записваме като потенциален отговор, ако досегашният за тази устойчивост е бил по-голям.

*Автор: Александър Георгиев*