

## ClosestNumber (Решение)

Задачата ClosestNumber е като че ли най-простата, която съм давал в С група, като очаквам голям голям брой хора (може би двуцифрен) да се справят с нея за 100 точки.

Задачата се решава с техниката "търсене с връщане" (backtrack), като пробва всички възможности за умножения на числа от интервала [1, 20]. Все пак, просто така задачата би била твърде тривиална за тази група. Затова ограниченията са достатъчно големи, че в най-простия си вид бектрекът да върви твърде бавно.

За да хванат 100 точки, участниците трябваше да направят две прости наблюдения, които водят до огромни оптимизации на скоростта.

Първото наблюдение беше, че може да се намали броят числа, по които умножаваме. Наистина, ако сме пробвали да умножим по 3 и по 5 (потенциално по няколко път), нужно ли е да пробваме да умножаваме и по 15? Не, защото бихме могли да умножим още веднъж по 3 и още веднъж по 5 като постигнем същото нещо ( $15 = 3 * 5$ ). Аналогично не е нужно да умножаваме по 14, тъй като  $14 = 2 * 7$ . Елиминирайки и останалите числа, които могат да се представят като произведение на други, в крайна сметка оставаме само с простите числа между [1, 20]. Така вместо да пробваме да умножаваме по двадесет числа (окей де, деветнадесет, защото 1 най-вероятно бихте изключили автоматично), ще умножаваме само по осем: {2, 3, 5, 7, 11, 13, 17, 19}. Това значително забързва бектрека и без тази стъпка решението би било твърде бавно.

Дори ползвайки това наблюдение, обаче, решението би било относително бавно, ако просто на всяка стъпка пробваме да умножаваме по някое от тези числа. В много случаи бихме стигали до едно и също число. Защо?  $2 * 5 * 3 * 2 = 60 = 5 * 3 * 2 * 2 = 2 * 2 * 3 * 5$ . Тъй като умножението е комутативно, можем да разменяме местата на множителите без резултатът да се променя. Дори за малко число като 60 имаме цели 12 начина, по които можем да наредим неговите множители {2, 2, 3, 5}. Това е крайно неефективно, тъй като по никакъв начин пробването на различни наредби не помага за намиране на по-добър отговор. За да върви решението ни достатъчно бързо е нужно да го изградим по такъв начин, че да не повтаря едно и също число.

За да направим това, трябва освен текущото число, да подаваме на рекурсията и индекса на последното от простите числа, по което сме умножавали. Вътре в рекурсията пробваме да умножим най-много по две числа: това, по което до сега сме умножавали, и най-близкото по-голямо. Така всяко възможно число се генерира само по веднъж и вече решението е достатъчно бързо.

Автор: Александър Георгиев