АНАЛИЗ на Двоични нули

Обхождането на интервала [*s*, *t*] и проверката за брой на двоичните нули във всяко число от него може да се прави по различни начини:

* Чрез превръщане (с деление на две) на всяко от числата в интервала;
* Чрез побитова проверка за четност и побитово отместване надясно;
* Чрез модификация на известния алгоритъм за преброяване на единиците в двоичен запис (предхождано от инвертиране на значещите битове).

Колкото по-умно е реализирано такова сканиране на [*s*, *t*], толкова по-добре (и надежди за до 30% от предвидените точки). Отделно може да се съобрази, че за всеки брой двоични цифри в интервала сканирането може да започне от и за завърши на (ако, разбира се, те не са извън интервала). Възможни са и други идеи за „прескачане“ на ненужни проверки. Естествено, наивната сканираща идея, колкото и умно да е реализирана, дори и да се замени с генерираща, има сложност най-малко колкото е броят на числата с желаното свойство (тъй като на всяка стъпка се добавя най-много 1), а този брой може да бъде голям.

Специално зададените ограничения „между две степени на двойката“ (разбирай – измежду всички двоични записи с брой цифри, започващ от броя двоични цифри на *s* и завършващ с този на *t*-1) целят да подскажат друг, комбинаторен поглед към проблема. Наистина, ако гледаме на значещите битове като на *n* кутийки, в които може да има 0 или 1, положението на нулите може да се разглежда като извадка без повторение от *n*-1 елемента (старшият бит винаги е 1) от *z*-ти клас. Както е известно, този брой е и, следователно, търсеният брой в този случай е сумата от съответните биномни коефициенти.

За пълно решение на задачата можем да направим още една крачка: да подредим *n*-цифрените записи със *z* нули (или, все едно, извадките без повторение) в нарастващ ред (в случая на еднакъв брой цифри – все едно в азбучен ред). Известен е алгоритъмът, чрез който бързо може да бъде намерен номерът на една извадка в такава подредба. Тази идея използваме, за да преброим търсените записи в началото и края на интервала [*s*, *t*], които, евентуално, съдържат само част от числата с исканото свойство. Остава да създадем една характерна извадка за дадено число *a*, например тази, която описва първото число с желаното свойство, което не е по-малко от *a*.

Да запишем двоично числото *a*. Ако в записа вече има поне *z* нули, то местата на старшите *z* от тях образуват търсената извадка. В противен случай, тръгвайки от най-младшата нула, търсим тази, която можем да превърнем в 1, като плътно надясно от нея има място да допълним записа с нули до общ брой *z*. Първата намерена дава нужната извадка. Ако такава няма, то след *a* няма записи, които ни интересуват. Добре е да знаем и това, дали самото *a* е от търсените записи.

Този алгоритъм използваме за началото *s* и края *t* на зададения интервал. Трябва да се внимава при случаите, когато *s* и *t* са с еднакъв брой двоични цифри.

*Автор: Павлин Пеев*