

АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА ТЕЛЕФОННИ СМЕТКИ

В тази задача, броят на числата може да е много голям и решения със сложност $O(N^2)$ и няма да получат висока оценка.

Тук изглежда естествено такова решение: да се сортират числата от втория масив (платените сметки) и за всяко число от първия масив (сметките за плащане) да се провери с двоично търсене дали го има в сортирания масив. Такова решение щеше да е много добро, ако в масивите няма повтарящи се числа. Ако такива числа има, а условието не изключва такава възможност, решението даже може да е грешно, защото е достатъчно за няколко еднакви сметки за плащане да има една фактура на съответната сума и двоичното търсене без допълнителни проверки заблуждаващо ще реши, че всичките сметки са платени. Поправянето на този алгоритъм така, че да може да отчита повторенията се сблъсква с проблеми.

Предлагаме следното, по-доброто решение за този случай. Сортираме и двата масива в нарастващ ред. Аналогично на алгоритъма за сливане на два сортирани масива, поставяме по един индекс в началото на двата масива и започваме сравняване на сочените от двата индекса числа:

- Ако двете числа са равни това съответства на платена сметка и увеличаваме и двата индекса;
- Ако числото в първия масив е по-голямо от числото във втория масив, това съответства на неплатена сметка, програмата извежда числото от първия масив и увеличава индекса му;
- Не е възможно числото в първия масив да е по-малко от числото във втория! Това би означавало, че измежду плащанията е попаднало такова, за което не съществува сметка за плащане, което по условие е невъзможно.

След като вторият индекс напусне границите на масива, тогава всички останали в първия масив числа, ако изобщо има такива), съответстват на неплатени сметки и трябва да бъдат изведени в резултата.

Автор: Красимир Манев