

Задача 1[†] Масивът $A[1, \dots, n]$ съдържа всички цели числа от 0 до n без едно. Елементите на A са записани в двоична позиционна бройна система и единственият разрешен достъп до елементите на A е побитов. Това значи, че **не е** допустимо да се прави достъп от вида $A[i]$ в константно време, какъвто е нормален за масив от цели числа; единствено допустимият достъп в константно време е от вида „дай ми j -ия бит на i -тото число на A “. Казано по друг начин, A е матрица от нули и единици с размер m реда на n колони, където m е броят на битовете, необходими и достатъчни за представяне на числото n , и в константно време имаме достъп до елементите на тази матрица. Пояснение: първото число в масива се разполага в първата колона, второто, във втората, и т.н. Допуснете, че най-младшият бит на всяко число е на ред m , и т.н., най-старшият бит е на ред 1. A не е сортиран.

Предложете алгоритъм със сложност по време $O(n)$, който връща липсващото число. Ограничението за побитов достъп е в сила само за A . Алгоритъмът може да ползва помощни променливи, които са цели числа или масиви от цели числа или други типове данни, такива че достъпът до тях е нормалният достъп, с който сме работили досега.

Колко бързо можете да откриете липсващото число, ако A е сортиран, но същите ограничения за достъпа остават в сила?

Задача 2[‡] Нека $G = (V, E)$ е ориентиран граф без примки и $n = |V|$. *Сифон* (sink) се нарича всеки връх $u \in V$, такъв че $\forall v \in V, v \neq u : (v, u) \in E$, но $\forall w \in V : (u, w) \notin E$. Нека G е представен с матрица на съседствата (adjacency matrix). Конструирайте алгоритъм, който има сложност по време $O(n)$ и връща „ДА“, ако G има сифон, и „НЕ“, в противен случай.

Домашното да бъде предадено най-късно на 12 април 2010 г. до края на лекциите. Оценяването е с точки, като първата задача е 20 точки, а втората, 30. Втората задача е по-сложна от нормалното за домашно и ще се счита за „бонус“, тоест за 100% е достатъчно да се реши първата задача.

В първата задача искам съвсем подробно решение – много детайлен псевдокод или, още по-добре, програма. Ако решавате втората, нека решението не е много дълго. Опишете идеята недвусмислено на псевдокод. Особено важно е да се направи адекватен анализ на коректността и на сложността по време на предложените алгоритми. Решения, в които липсва какъвто и да е анализ, ще бъдат игнорирани.

Напишете имената си и факултетните номера върху работата си. Ако ползвате няколко листа, моля ви, защитете ги с телбод.

[†]Това е Задача 4-2 от учебника на Cormen, Leiserson, Rivest. В първото издание е на стр. 73.

[‡]Това е Задача 23.1-6 от учебника на Cormen, Leiserson, Rivest. В първото издание е на стр. 468.