

## ИЗПИТ ПО “ДИЗАЙН И АНАЛИЗ НА АЛГОРИТМИ” — 10.02.2018 Г.

**Задача 1.** Алгоритмичната задача `SubsetSum` остава ли NP-трудна в частния случай, когато всички дадени числа са четни?

**Задача 2.** Експертна система разполага с  $m$  доказателства на  $n$  теореми. Всяко доказателство представлява извеждане на една теорема от друга. Системата умее да комбинира известните ѝ доказателства: ако от теоремата  $X$  следва теоремата  $Y$ , а от теоремата  $Y$  следва теоремата  $Z$ , то от теоремата  $X$  следва теоремата  $Z$ .

За всеки от следните проблеми предложете алгоритъм с линейна времева сложност  $\Theta(m + n)$ :

- По две дадени теореми системата трябва да определи дали едната теорема следва от другата; ако да, тогава трябва да се състави доказателство чрез комбиниране на минимален брой от известните  $m$  доказателства.
- Множеството от  $n$  теореми да се разбие на подмножества, всяко от които се състои от еквивалентни теореми.

**Зад. 3.** Колко са  $n$ -цифрените числа, съставени само от цифрите 1, 2 и 3 по следните правила?

- След цифрата 1, ако не е последна, стои задължително цифрата 2.
  - След цифрата 2, ако не е последна, може да стои коя да е от цифрите 1 и 3.
  - След цифрата 3, ако не е последна, може да стои коя да е от цифрите 2 и 3.
- Трите правила образуват една задача, т.е. трябва да бъдат изпълнени и трите.

Предложете итеративен алгоритъм. Опишете го на псевдокод като функция `cnt(n: positive integer) : positive integer` с време  $O(n)$  и количество допълнителна памет  $O(n)$ .

Демонстрирайте алгоритъма при  $n = 6$ .

**Задача 4** Предложете бърз алгоритъм, който проверява дали масив от  $n$  числа съдържа противоположни числа.

**Задача 5.** Намерете асимптотичната сложност на алгоритъма.

```
int f(int n, int m) {
    int i, s = 0;
    if (n == 0 || n == 1)
        return m;
    for(i = 0; i < 5; i++)
    {
        s += f(n-1, m + i);
        s += f(n-2, m + 2*i);
    }
    s += f(n-2, 2*m)*3;
    return s; }
```

**Задача 6.** Какво връща следният алгоритъм? Отговорът да се обоснове чрез инварианта.

```
int h(A[1...2n]: array of int) {
    int s = 0;
    for(int k = 1; k ≤ n; k++)
        s += (A[k] + A[2n+1-k]);
    return s;
}
```