

**ИЗПИТ ПО “ДИЗАЙН И АНАЛИЗ НА АЛГОРИТМИ”  
(ПОПРАВИТЕЛНА СЕСИЯ — СУ, ФМИ, 27 ЮНИ 2017 Г.)**

Задача №	1	2	3	4	5	6	Общо
Получени точки							
Максимален брой точки	20	20	20	40	30	20	150

**Задача 1.** В клетките на електронна таблица са въведени формули. Например, ако в клетката B8 е написана формулата “= A4 + C7 – 2 \* D9”, то трябва първо да се пресметнат стойностите на клетките A4, C7 и D9, а след това — на B8. Предложете възможно най-бърз алгоритъм, който да определя в какъв ред да се изчисляват стойностите на клетките. Опишете алгоритъма словесно.

**Задача 2.** Оценете по порядък времевата сложност на алгоритъма:

```
P(A[1...n]: array of integers)
s ← 15
for k ← n downto 1 do
    s ← s + A[k] - P(A[1...k-1])
return s
```

**Задача 3.** За  $\forall n$  намерете граф  $G$  с  $n$  върха и връх  $r \in G$ , та ако пуснем BFS и DFS от  $r$ , стекът на DFS да има  $n - 1$  върха в някой миг, а опашката на BFS винаги да има най-много един връх.

Намерете друг граф  $G$  и връх  $r \in G$ , но сега стекът винаги да е с максимум два върха, а опашката да има  $n - 1$  върха в някой миг.

**Задача 4.** Професор има  $n$  научни публикации. Масивът  $A[1...n]$  съдържа данни за това, колко пъти всяка от тях е била цитирана от други учени; тоест  $A[k]$  е броят на цитиранията на  $k$ -тата публикация. Предложете алгоритъм за пресмятане на т. нар.  $h$ -индекс — най-голямото цяло  $h$ , за което е вярно, че професорът има поне  $h$  публикации, всяка от които е цитирана поне  $h$  пъти.

Алгоритъм с времева сложност  $T(n) = O(n \log n)$  се оценява с 20 точки; за сложност  $T(n) = O(n)$  се дават 40 точки. За по-бавни алгоритми: 0 точки.

Опишете алгоритъма с думи и го демонстрирайте. Анализирайте сложността.

**Задача 5.** По дадени цели положителни числа  $A_1, A_2, \dots, A_n$  и  $B$  намерете броя на решенията на уравнението

$$A_1 x_1 + A_2 x_2 + \dots + A_n x_n = B$$

в цели неотрицателни числа, тоест броя на наредените  $n$ -орки  $(x_1, x_2, \dots, x_n)$  от цели неотрицателни числа, които удовлетворяват уравнението.

а) Предложете итеративен алгоритъм. Опишете го на псевдокод като функция `numSolEq(A[1...n]: array of int, B: int): int`

с време  $O(nB)$  и динамична таблица с  $O(nB)$  клетки. **(10 точки)**

б) Демонстрирайте алгоритъма при  $A = (2; 3; 5)$  и  $B = 9$ . **(10 точки)**

в) Оптимизирайте сложността по памет до динамична таблица с  $O(B)$  клетки.

Опишете оптимизирания алгоритъм на псевдокод. **(10 точки)**

**Задача 6.** Докажете, че задачата за разпознаване, дали даден ориентиран граф съдържа хамилтонов цикъл, остава NP-пълна, когато графът е двуделен.