

Име: _____, ФН: _____, Спец.: _____, Курс: _____

Задача	1	2	3	4	5	6	Общо
получени точки							
максимум точки	20	20	20	20	10	30	120

Забележка: За отлична оценка са достатъчни 100 точки!

Задача 1. Дадени са времевите сложности на няколко алгоритъма:
 $9 \cdot 5^n$, $n^7 \log(8n + 14)$, $7 \cdot 5^{n^2}$, $(5n)^{7n}$, $\sqrt{n^7}$.
 Подредете алгоритмите по бързодействие (най-бързия на първо място). Отговорът да се обоснове!

Задача 2. Да се решат рекурентните уравнения:

а) $T(n) = 10 T(n - 1) - 16 T(n - 2) + 8^n$; б) $T(n) = T(n - 1) + \sqrt{n^{81}}$;

в) $T(n) = 6^n + \sum_{k=0}^{n-1} T(k)$; г) $T(n) = 243 T\left(\frac{n}{3}\right) + 9n^4$.

Задача 3. Имаме следния алгоритъм:

```

MYFUNC(n: integer)
1  s ← 4
2  while s ≤ n do
3      s ← 8 * s + 6
4  return s
    
```

- а) Каква стойност връща алгоритъмът?
 б) Оценете по порядък времевата сложност на алгоритъма.

Задача 4. Даден е масив от n цели положителни числа, които могат да бъдат много големи. Съставете възможно най-бърз алгоритъм, който да намира най-малкото цяло положително число, липсващо в масива. Оценете сложността на предложението от Вас алгоритъм в най-лошия случай.

Задача 5. Даден е масив от числа $A[1 \dots n]$. Докажете, че ако масивът не е сортиран, съществува индекс $i < n$, за който $A[i] > A[i + 1]$.

Задача 6. Алгоритъмът по-долу получава масив A , който съдържа n числа:

```

INVERSIONSORT(A[1...n])
1  i ← 1
2  while i < n do
3      if A[i] > A[i + 1]
4          swap(A[i], A[i + 1])
5          if i = 1
6              i ← i + 1
7          else i ← i - 1
8      else i ← i + 1
    
```

- (а - 15 точки) Докажете, че алгоритъмът сортира масива $A[1 \dots n]$.
 (б - 15 точки) Ако времевата му сложност е $T(n)$, докажете че $T(n) = \Theta(n^2)$.

Упътване: Използвайте наготово зад. 5 за да докажете условие (а) и математическа индукция, за да докажете подусловие (б).

Коментари:

- 1) Задачи 1 и 2 са шаблонни. Не държа много на тях, но е добре да има и такива.
- 2) В задача 3 умишлено не съм дал инвариантата: студентите да си я изведат сами, прави се с линейно-рекурентно уравнение, което се съставя лесно и се решава лесно.

На упражненията сме разглеждали доста такива примери, затова настоявам за зад.3.

- 3) Задача 4 в първата си формулировка е трепач: 'всичко или нищо', което не е много добре. Затова съм приложил и втория вариант, който дава възможност за диференцирано оценяване:
 - 0 т. или 5 т. за решение с квадратична сложност (без сортиране или с бавна сортировка);
 - 10 т. за решение със сложност $\Theta(n \cdot \log n)$ (с някоя от бързите сортировки);
 - 20 т. за решение с модификация на сортирането чрез броене (CountingSort);
 - 0 т. за решение с оригиналното сортиране чрез броене (казано е, че числата са големи).