

## КОНТРОЛНА РАБОТА №2 по ДАА, КОМП. НАУКИ, 03.06.2011

Име:..... Ф№:..... Група:....

Задача	1	2	3	4	5	6	ОБЩО
<i>получени точки</i>							
<i>от максимално</i>	$36 = 9 + 10 + 9 + 8$	11	11	11	14	17	100

**Зад. 1** Дадени са следните четири програмни фрагмента. За всеки от тях, определете асимптотичната сложност по време като функция на  $n$ . Във **в)**, функцията `pow()` е стандартната функция за степенуване в **C**: `pow(x, y)` връща  $x$  на степен  $y$ .

**а)**

```
void f1(int n) {
    int i, x = 0, y = 0;
    if(n <= 1) return;
    for(i = 1; i <= n*n; i *= 2) x++;
    f1(n/2);
    for(i = 1; i <= n/2; i ++) y++;
    f1(n/2); }
```

**б)**

```
int f2(int n) {
    int i = 2, j = 6, a = n+7, z = 0;
    if(n <= 1) return 1;
    for(; j - i > 1; i ++, j --, a ++) {
        z += f(a - (i+j));
        a --; }
    return z; }
```

**в)**

```
int f3(int n) {
    int i, s = 0;
    if(n <= 1) return 2;
    for(i = 1; i <= 5; i ++) {
        s += f(n-1);
        s += f(n-2); }
    s += f(n-2);
    for(i = pow(6, n); i > 0; ) {
        s ++; i --; }
    return s; }
```

**г)**

```
int f4(int n) {
    int i, s = 0;
    if(n <= 1) return 1;
    for(i = 1; i <= n; i ++) {
        s += f(n-1); }
    return s; }
```

Всяка от следващите пет задачи иска да се измисли и опише алгоритъм. Описанието на алгоритмите трябва да бъде съвършено ясно, без никакво двусмислие или неясноти. Пишете детайлен псевдокод или словесно описание, от което да е абсолютно ясно как да се напише програма, имплементираща предложения алгоритъм. Не дискутирайте примери и не разказвайте как бихте решили задачата за конкретен пример. Описания на алгоритми,

които съдържат неясноти или са прекалено недетайлни или използват конкретни примери няма да бъдат оценявани.

Добре е да се даде аргументация за коректността на всеки описан алгоритъм – колкото по-формална и прецизна, толкова по-добре. Задължително е да се даде оценка на сложността по време на всеки алгоритъм. Колкото по-прецизно бъде обоснована дадена оценка за сложност, толкова по-добре.

**Зад. 2** Предложете колкото е възможно по-бърз (в асимптотичния смисъл) алгоритъм, който има вход масив от цели числа, поне две на брой, и връща индексите на два елемента от масива, такива че произведението на тези елементи е максимално.

**Зад. 3** Предложете колкото е възможно по-бърз (в асимптотичния смисъл) алгоритъм, който връща медианата на несортиран масив  $A[1, \dots, n]$ . Приемете, че  $A[1, \dots, n]$  съдържа възрастите на хора от някакво множество.  $A$  именно, има множество от  $n$  хора  $\{h_1, h_2, \dots, h_n\}$  и за всяко  $i$ , такава че  $1 \leq i \leq n$ ,  $A[i]$  е възрастта на  $h_i$ .

**Зад. 4** Предложете колкото е възможно по-бърз (в асимптотичния смисъл) алгоритъм, който изчислява симетричната разлика на два несортирани масива с различни големина  $A[1, \dots, n]$  и  $B[1, \dots, m]$ . Масивите са от естествени числа.

**Зад. 5** Предложете колкото е възможно по-бърз (в асимптотичния смисъл) алгоритъм, който изчислява второто по тегло покриващо дърво на свързан неориентиран тегловен граф  $G(V, E, w)$ . Второто по тегло дърво означава следното. Нека  $T_1, T_2, \dots, T_p$  са всички покриващи дървета на  $G$ , сортирани във възходящ ред по тегло. Тогава  $T_1$  е дърво с минимално тегло – то е дървото, което изчисляват известните ви алгоритми за минимално покриващо дърво. В тази задача се иска да се изчисли  $T_2$  – следващото по тегло дърво.

**Зад. 6** Предложете колкото е възможно по-бърз (в асимптотичния смисъл) алгоритъм за следната задача. Даден е масив от цели числа,  $n$  на брой. Да се намерят индекси  $i$  и  $j$ , такива че сумата  $\sum_{k=i}^j A[k]$  се максимизира. *Забележка:* Въпросната сума трябва да е неотрицателна. Ако всички числа в масива са отрицателни, тази сума трябва да е нула.