

ДОМАШНА РАБОТА № 2 ПО ДИЗАЙН И АНАЛИЗ НА АЛГОРИТМИ—2010

Име:..... Ф№:..... Група:.....

	1.а	1.б	1.в	2	3.а	3.б	3.в	3.г	3.д	ОБЩО
Оценка:	<i>получена</i>									
	<i>от макс.</i>	<i>10</i>	<i>10</i>	<i>4</i>	<i>10</i>	<i>2</i>	<i>4</i>	<i>12</i>	<i>6</i>	<i>12</i>
										<i>70</i>

Домашното да бъде предадено най-късно на 22 март 2010 г. до края на лекциите. Защипете всички листа с телбод. Ако искате, можете да ползвате разпечатка на този pdf файл като начални страници. В противен случай напишете на първата страница името си, Ф№, номера на групата, и същата табличка за оценяване.

Задача 1 Разгледайте следния алгоритъм.

BUBBLESORT($A[1 \dots n]$: array of nonnegative integers)

```

1  for i ← 1 to n do
2      for j ← n downto i + 1 do
3          if A[j] < A[j - 1]
4              swap(A[j], A[j - 1])
    
```

Целта на задачата е да се покаже, че BUUBLESORT сортира входния си масив. A именно, че

$$A'[1] \leq A'[2] \leq \dots \leq A'[n] \tag{1}$$

където $A'[1 \dots n]$ е масивът след приключване на работата на алгоритъма. Инвариантите, които трябва да посочите и докажете в следващите подусловия, трябва да бъдат съобразени именно с тази крайна цел.

- a)* Намерете и докажете инварианта на вътрешния **for** цикъл (редове 2–4).
- б)* Използвайки заключението на инвариантата от подусловие *a)*, намерете и докажете инварианта за външния цикъл **for** (редове 1–4).
- в)* Използвайки заключението на инвариантата от подусловие *б)*, докажете, че неравенство (1) е изпълнено след приключване на работата на алгоритъма.

Задача 2 Напишете алгоритъм със сложност по време $O(n \lg n)$ в най-лошия случай, който има вход от:

- масив $A[1 \dots n]$ от положителни реални числа и
- положително реално число x

и изход „ДА“, ако в A има два елемента y и z такива, че $y + z = x$, и „НЕ“, в противен случай. Описанието на алгоритъма може да бъде на всякакъв разбираем и недвусмислен псевдокод. Дайте кратка аргументация защо смятате, че алгоритъмът Ви е коректен.

Задача 3 Преди условието на задачата, една дефиниция. Нека $A[1 \dots n]$ е масив от n различни естествени числа. За всяко i и всяко j , такива че $1 \leq i < j \leq n$ и $A[i] > A[j]$, наредената двойка (i, j) се нарича *инверсия*. Забележете, че A да няма инверсии е същото като A да е сортиран.

Задачата прави отправка към алгоритъма INSERTIONSORT:

INSERTIONSORT($A[1 \dots n]$: array of nonnegative integers)

```
1 for i ← 2 to n do
2   key ← A[i]
3   j ← i - 1
4   while j > 0 and A[j] > key do
5     A[j + 1] ← A[j]
6     j ← j - 1
7   A[j + 1] ← key
```

a) Посочете в явен вид всички инверсии на масива $[2, 3, 8, 6, 1]$.

b) Какъв е максималният възможен брой инверсии в $A[1 \dots n]$? Какво характеризира масив, имащ максимален брой инверсии?

в) Разгледайте алгоритъма INSERTIONSORT и си представете работата му върху A . Докажете по индукция по i , че за всяка позиция i където $2 \leq i \leq n$, броят на инверсиите от вида $(k, A[i])$ в оригиналния масив A^\dagger е равен на броя на изпълненията на тялото на цикъла **while** (редове 5 и 6) при съответната стойност на i .

г) Използвайки резултата от подусловие в), открийте и докажете връзка между броя на инверсиите в оригиналния входен масив A и сложността по време на INSERTIONSORT. А именно, ако входът има m инверсии, изразете асимптотичната сложност по време като функция на n и m .

д) Предложете алгоритъм, който изчислява броя на инверсиите в масив $S[1 \dots n]$ и има сложност по време в най-лошия случай $O(n \lg n)$. Упътване: това подусловие няма нищо общо с предишните две. Направете модификация на MERGESORT.

[†]Под „оригиналният масив A “ разбираме A преди алгоритъмът да започне да размества елементите му.