ДАА

упр.№4, 16 април 2020г.

Hello, world!

$f_1 = 2^{2n} = 4^n \quad f_2 = 1{,}01^{1{,}01^n} \quad f_3 = (n+1)! \quad f_4 = (2n)!$

$f_5 = n^{\lg n} \quad f_6 = 3^{n+\sqrt{n}} \quad f_7 = 4^{n-\sqrt{n}} \quad f_8 = e^n$

$f_5 < f_8 < f_6 < f_7 < f_1 < f_3 < f_4 < f_2$

За дост. големи n:

$1 \lg n \leq \lg(n+1) \leq \lg(2n) = \lg n + 1 \leq 2\lg n$

$f_1 \Leftrightarrow f_3$: $\lg f_1 = n \lg 4 \asymp n$

$\lg f_3 \asymp (n+1) \cdot \lg(n+1) \asymp (n+1) \cdot \lg n \asymp n \lg n \Rightarrow \lg f_3 \succ \lg f_1$

$\lg f_4 \asymp 2n \cdot \lg 2n \asymp n \lg n$

$\lg f_2 = 1{,}01^n \cdot \lg 1{,}01 \asymp 1{,}01^n$

GrahamScan(A[1..n] : array of integers)

```
1.   S ← празен стек (допускаме Θ(1) операции с него)
2.   push(A[1], S)
3.   push(A[2], S)
4.   for i ← 3 to n
5.     while P(S) // предикат, работи за Θ(1) и връща False за празен стек
6.       pop(S) ← ще се изпълни общо ≤ n пъти
7.     push(A[i], S)
```

$T(n) = O(n^2)$

$T(n) = \underbrace{O(n)}_{\text{ред } 6} + \sum_{i=3}^{n} O(1) + \underbrace{\Theta(1)}_{\text{ред } 2,3} =$

$= O(n) + \Theta(n) + \Theta(1) = \Theta(n)$

$M(n) = O(n)$

Fib(n : nonnegative integer)

```
1.  if n < 2
2.     return n
3.  curr ← 1, next ← 1
4.  for i ← 2 to n
5.     next ← next + curr
6.     curr ← next' – curr
7.  return curr
```

$\sum_{i=2}^{n}$

Инв.: При $\forall$ дост. на ред 4 $curr = F_{i-1}$; $next = F_i$

База: При първото дост. $i = 2$, а от ред 3 имаме, че $curr = 1 = F_1$ и $next = 1 = F_2 \Rightarrow$ Инв. е вярна

Подд.: Нека Инв. е вярна за някое дост., което не е посл., т.е. $next = F_i$; $curr = F_{i-1}$

След ред 5 $next' = next + curr \overset{\text{Инв.}}{=} F_i + F_{i-1} = F_{i+1}$

След ред 6 $curr' = next' - curr \overset{\text{Инв.}}{=} F_{i+1} - F_{i-1} = F_i$

При следв. достигане новото $i' = i + 1 \Rightarrow$ спрямо новото i' $next = F_{i'}$; $curr = F_{i'-1}$ $\Rightarrow$ Инв. е изпълнена

Терм: при посл. дост. на ред 4 Инв. ще е вярна и $i = n+1 \Rightarrow$ от Инв. затова $curr = F_n$; $next = F_{n+1}$

Тв: Алг. връща F_n за $\forall n \geq 0$

Док: Нека $n < 2$ → ще се изпълнят редове 1 и 2 ⇒ Тв. е вярно

Нека $n \geq 3$ → ще използваме Инв. на цикъла на ред 4

⇒ от Инв при достигане на ред 7 $curr = F_n$

и това е стойността, която се връща ⇒ Тв. е вярно

⇒ Преди Терм: т.к. на всяка итерация i нараства, а n не се променя, то след краен брой итерации i ще надмине n

5, 4, 3, 2, 1

1 2 3, 1 2, 1 1 1 3 5, 2 4

NumSlopes(A[1..n] : array of integers)

макс. по бр. подмасиви с ненамаляващи ел-ти

ТБ: Алг. връща броя слонове в A[1..n]

```
1.  res ← 1
2.  for i ← 2 to n
3.    if A[i-1] > A[i]
4.      res ← res + 1
5.  return res
```

Док: Ще използваме Инв. на цикъла на ред 2

Инв: При ∀ дост. на ред 2 res = броя слонове в A[1..i-1]

База: При първото дост. i=2 и res=1, което е броя слонове на A[1..i-1] = A[1..1]

Подд.: Нека Инв. е изпълн. на някое дост. на ред 2, което не е посл. ⇒ Инв. е изпълнена

Има 2 случая:

I сл: $A[i-1] > A[i]$. ⇒ $A[i]$ не участва в склона на $A[i-1]$

⇒ брой склонове в $A[1..i]$ = брой в $A[1..i-1] + 1$ - това

е склонът, в който $A[i]$ е сам.

В този ~~момент~~ случай ще се изп. ред 4 и $res' = res + 1 \overset{Инв.}{=}$

брой в $A[1..i-1] + 1$ = брой за $A[1..i]$ ⇒ при след. сост.

на ред 2 $i' = i + 1$ и спрямо новото i' $res =$ брой за $A[1..i'-1]$

⇒ Инв. е вярна

II сл. $A[i-1] \leq A[i]$ ⇒ $A[i]$ ще е част от склона на $A[i-1]$

⇒ брой склонове за $A[1..i-1]$ = брой за $A[1..i]$

Според Инв. $res = \ldots$ за $A[1..i-1] = \ldots A[1..i] = A[1..i'-1]$ спрямо новото i'

⇒ Инв. е вярна.