4. oktober 2024

DN 0 reši čimprej !!!
Teoretična

$O(f(n)) = \{ g(n) \mid (\exists x > 0)(\exists n_0 \in \mathbb{N})(\forall n) (n \geq n_0 \Rightarrow g(n) \leq x \cdot f(n)) \}$

na neki točki g ne raste hitreje kot f

$o(f(n)) = \{ g(n) \mid (\forall x > 0)(\exists n_0 \in \mathbb{N}) (\forall n)(n \geq n_0 \Rightarrow g(n) < x \cdot f(n)) \}$

na neki točki g zmiraj raste počasneje kot f

$\Omega(f(n)) = \{ g(n) \mid (\exists x > 0)(\exists n_0 \in \mathbb{N})(\forall n) (n \geq n_0 \Rightarrow g(n) \geq x \cdot f(n)) \}$

g raste vsaj tako hitro kot f

$\Theta(f(n)) = \{ g(n) \mid g \in O(f(n)) \wedge g \in \Omega(f(n)) \}$

g raste enako hitro kot f

Algoritem ima časovno zahtevnost

$T(n) = \frac{1}{2} n^2 - 3n$

Pokaži da je $T(n) \in \Theta(n^2)$

① $\underbrace{T(n)}_{g} = O(\underbrace{n^2}_{f})$

Iščemo $x > 0$ in $n_0 \in \mathbb{N}$, da velja

$\frac{1}{2} n^2 - 3n \leq x n^2 \quad /:n^2$ za vse $n \geq n_0$

$\frac{1}{2} - \frac{3}{n} \leq x$. Vzemimo $c = \frac{1}{2}$

Torej $\frac{1}{2} - \frac{3}{n} \leq \frac{1}{2}$ ker leva stran ne bo nikoli več kot $\frac{1}{2}$

$\underline{\underline{\frac{3}{n} \geq 0}}$ Vzemimo $n_0 = 1$.

② $\underbrace{T(n)}_{g} = \Omega(\underbrace{n^2}_{f})$

Iščemo $x > 0$ in $n_0 \in \mathbb{N}$, da velja

$\frac{1}{2} n^2 - 3n \geq x n^2 \quad /:n^2$ za vse $n \geq n$

$\frac{1}{2} - \frac{3}{n} \geq c$. leva stran ne bo nikoli več kot $\frac{1}{2}$

$\frac{1}{2} - \frac{3}{n} \geq \frac{1}{4} \quad /\cdot 4n$ Vzamemo $c = \frac{1}{4}$

$2n - 12 \geq n \Rightarrow n \geq 12$

Vzamemo $n_0 = 12$

Pokaži da velja

(i) $2^{n+1} \in O(2^n)$

(ii) $2^{2n} \notin O(2^n)$

① $\underbrace{2^{n+1}}_{g} \in O(\underbrace{2^n}_{f})$

Iščemo $c > 0$ in $n_0 \in \mathbb{N}$, da za vse $n \geq n_0$ velja:

$2^{n+1} \leq c \cdot 2^n$

$2 \cdot 2^n \leq c \cdot 2^n$

$2 \leq c$

Vzemimo $c = 2$

Vzemimo $n_0 = 1$ saj velja za vse saj je rešitev neodvisna od n

② Po definiciji

$2^{2n} \notin O(2^n)$

kdaj $g \notin O$

torej

$\neg (\exists c > 0)(\exists n_0 \in \mathbb{N})(\forall n)(n \geq n_0 \Rightarrow g(n) \leq c f(n))$

$(\forall c > 0)(\forall n_0 \in \mathbb{N})(\exists n)(n \geq n_0 \wedge g(n) > c f(n))$

Naj bosta $c > 0$ in $n_0 \in \mathbb{N}$ poljubna

iščemo n da velja

$n \geq n_0$ in $2^{2n} > c \cdot 2^n \quad /:2^n$

Sledi

$2^n > c \quad / \log_2$

$n > \log_2(c)$

Vzemimo $n = \log_2 c + n_0$

__Opomba:__ če je $f \in o(g)$ potem $g \notin O(f)$

② z pomočjo o

$\underbrace{2^{2n}}_{g} \notin O(\underbrace{2^n}_{f})$

Pokažimo $2^n = o(2^{2n})$

$$\boxed{g(n) = O(f(n)) \Leftrightarrow \lim_{n \to \infty} \frac{g(n)}{f(n)} = 0 \qquad g' \quad f'}$$

$\lim_{n \to \infty} \frac{2^n}{2^{2n}} = \lim_{n \to \infty} \frac{1}{2^n} = 0$

Algoritem z izbiro

V.P. polje A celih števil z dolžino n

I:P urejeno polje A

```
for i in range (1, n)
    min_index = i
    for j in range (i+1, n)
        if A[j] < A[min index]
            min index = j
    swap (A[i], A[min_index])
```

① Pokaži pravilnost delovanja algoritma z izbiro uporabljajoč zančno invarianco

② Analiziraj časovno zahtevnost

Zančna invarianca: lastnost ki velja za vse iteracije

Zančna invarianca na začetku i-te iteracije velja:

podpolje $A[1 \ldots i-1]$ pravilno urejeno (od najmanjšega do največjega)

in vsebuje $i-1$ najmanjših elementov v A

Baza: $i = 1$ Na začetku prve iteracije velja da je podpolje $A[1 \ldots 0]$ urejeno in vsebuje 0 najmanjših elemento A ✓

Indukcijski korak: $i \to i+1$ Pokažimo $i+1$ iteracije velja da je $A[1 \ldots i]$ urejeno in vsebuje prvih i najmanjših elementov v A

Dovolj je da na začetku $i+1$ iteracije element $A[i]$ ~~ith~~ i-ti najmanjši element

Poglejmo si i-to iteracijo:

Notranja zanka poišče i-ti najmanjši element v A, ki je v A[min_index]

Klic swap (A[i], A[min_index]) zagotovi, da je i-ti ~~element~~ najmanjši element shranjen v A[i]

Sedaj uporabimo zančno invarianco poglejmo $i = n$

Po zančni invarianci je $A[1:n-1]$ urejeno in vsebuje $n-1$ prvih najmanjših elementov potem takem je $A[n]$ največji in je A pravilno urejen.