Algoritmų Analizė 1 N.D

Ričardas Čubukinas 1910620
Informatika III kursas
VU MIF

2022 m. vasario 24 d.

1 Uždavinys

(a)

$$f(n) = \overbrace{\sum_{k=2}^{n-1} k^2}^{1} - (\overbrace{\sum_{k=3}^{n} k}^{2})^2$$

1. $\sum_{k=2}^{n-1} k^2 = \sum_{k=1}^{n-1} k^2 - 1 = \frac{(k-1)((k-1)+1)(2(k-1)+1)}{6} - 1 = \frac{k(k-1)(2k-1)}{6} - 1$
2. $\sum_{k=3}^{n} k = \sum_{k=1}^{n} k - 3 = k * \frac{k+1}{2} - 3 = \frac{k^2+k-6}{2}$

$$\begin{aligned} f(n) &= \sum_{k=2}^{n-1} k^2 - (\sum_{k=3}^{n} k)^2 \\ &= \frac{k(k-1)(2k-1)}{6} - 1 - (\frac{k^2+k-6}{2})^2 \\ &= \frac{-3k^4 - 2k^3 + 27k^2 + 38k}{12} \end{aligned} \tag{1}$$

(b)

Kadangi funkcija neauga eksponentiškai, rasime asimptotiką, t.y a ir b tokias kad, $f(n) \sim an^b$, kai $n \to \infty$

$$\lim_{n\to\infty} \frac{\overbrace{-3n^4} - 2n^3 + 27n^2 + 38n}{12an^b} = 1 \tag{2}$$

$$\frac{-3 * n^4}{12an^b} \Rrightarrow \begin{cases} a = -\frac{1}{4} \\ b = 4 \end{cases} \tag{3}$$

$$f(n) \sim -\frac{n^4}{4} \tag{4}$$

2 Uždavinys

(a)

Duotas sveikų skaičių masyvas $A[1:n]; c = 2$.

```
for j := 1 to n do
  C[j] := 0
  i := n
    while i >= j do
      C[j] := C[j] + A[i]
      i := i - 1
  if C[j] < 0 then C[j] := 0
```

$2(n+1)$
n
n
$\sum_{i=1}^{n} i + n$
$\sum_{i=1}^{n} i$
$\sum_{i=1}^{n} i$
$2n$

$$\begin{aligned} L(n) &= 2(n+1) + n + n + 3\sum_{i=1}^{n} i + n + 2n \\ &= 2n + 2 + n + n + \frac{3n(n+1)}{2} + n + 2n \\ &= \frac{3n^2 + 17n + 4}{2} \end{aligned} \tag{5}$$

(b)

$$L(n) \sim a * n^b \tag{6}$$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 1 \tag{7}$$

$$\lim_{n\to\infty} \frac{\overbrace{3n^2} + 17n + 4}{2an^b} = 1 \tag{8}$$

$$\frac{3n^2}{2an^b} \Rrightarrow \begin{cases} a = \frac{3}{2} \\ b = 2 \end{cases} \tag{9}$$

$$L(n) \sim \frac{3n^2}{2} \tag{10}$$

(c)

Pasirinksime „blogiausius“ duomenis kai $n = c = 2$. Pakanka jog paskutinis masyvo skaičius būtų neigiamas ir masyvo skaičių suma būtų neigiama, tarkime $A = [-2, -5]$. Kadangi $L(2) = \frac{3\cdot4+17\cdot2+4}{2} = 25$, tai bus atlikti 25 žingsniai:

```
1  j := 1
2  1 <= 2? TRUE
3  C[1] = 0
4  i := 2
5  2 >= 1? TRUE
6  C[1] := -5
7  i := 1
```

```
1 >= 1? TRUE
C[1] := -7
i := 0
0 >= 1? FALSE
-7 < 0? TRUE
C[1] := 0
j := 2
2 <= 2? TRUE
C[2] := 0
i := 2
2 >= 2? TRUE
C[2] = -5
i := 1
1 >= 2? FALSE
-5 < 0? TRUE
C[2] = 0
j := 3
3 <= 2? FALSE
```

(d)

Skaičiuojant programos vykdymo laiką kiekvieną komandos eilutę „dauginsime“ iš konstantos c_i atitinkančios tos eilutės vykdymo laiką:

`for j := 1 to n do`	$c_1 2(n+1)$
`  C[j] := 0`	$c_2 n$
`  i := n`	$c_3 n$
`    while i >= j do`	$c_4(\sum_{i=1}^{n} i + n)$
`      C[j] := C[j] + A[i]`	$c_5 \sum_{i=1}^{n} i$
`      i := i - 1`	$c_6 \sum_{i=1}^{n} i$
`  if C[j] < 0 then C[j] := 0`	$c_7 2n$

Turime $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$. Apskaičiuojame $T(n)$:

$$\begin{aligned}
T(n) &= c_1 2(n+1) + n(c_2 + c_3 + c_4 + 2c_7) + (c_4 + c_5 + c_6)(\frac{n^2}{2} + \frac{n}{2}) \\
&= \frac{c_4 + c_5 + c_6}{2} n^2 + (c_2 + c_3 + \frac{3c_4}{2} + \frac{c_5}{2} + \frac{c_6}{2} + 2c_7)n + 2c_1 \\
&= An^2 + Bn + C
\end{aligned} \tag{11}$$

Taigi, $T(n) = \Theta(n^2)$. Kadangi $\Theta(n^d)$, kai $n \to \infty$ programos vykdymo laiko eilė $d = 2$. Kitaip tariant tai yra „kvadratinio sudėtingumo“ algoritmas.