Зад 1

Решаване на рекурентни уравнения

```
f(n: unsigned int): int
  s ← 10
  while s ≤ n do
      s ← 5.s + 3
  return s
```

a) Върнатата ст-т на $f(n) = ?$

b) Времева сложност $T(n) \asymp ?$

Нека s_k е ст-т на s след k итерации

$$\begin{cases} s_{k+1} = 5.s_k + 3 \\ s_0 = 10 \end{cases} \text{рек. ур-е}$$

Тогава $x^{k+1} = 5.x^k \ //: x^k$ (от хомогенно вс)

$x = 5$ т.е. $\{5\}_m$,

а от нехомогенната вс $\underbrace{3k^0.1^k}_{\{1^k\}}$

т.е. $\{5, 1\}_m$

$s_k = c_1 + 5^k.c_2$ общ вид

Трябват ни c_1 и c_2 да намеря!

За упражнение измислете инварианта и докажете коректност.

Заместваме за $k=0$ и $k=1$:

$$\begin{array}{l|l} k=0 \to & c_1 + c_2 = 10 \quad \cdot(-1) \\ k=1 & c_1 + 5\cdot c_2 = 53 \end{array}$$

$$s_1 = 5\cdot s_0 + 3 = 5\cdot 10 + 3 = 53$$

$$4c_2 = 43 \iff c_2 = \frac{43}{4}$$

Тогава $c_1 = 10 - \frac{43}{4} = -\frac{3}{4}$

Имаме $\boxed{s_k = -\frac{3}{4} + \frac{5^k\cdot 43}{4} = \frac{5^k\cdot 43 - 3}{4}}$ (*)

$$\underbrace{s_k}_{(*)} \le n \iff \frac{43\cdot 5^k - 3}{4} \le n \iff$$

спрямо k искаме да намерим горна граница

$$\iff 5^k \le \frac{4\cdot n + 3}{43} \quad // \log_5$$

$$\iff k \le \log_5\left(\frac{4n+3}{43}\right)$$

Отговор: $f(n) = s_t$, където

$$\boxed{t = \left\lfloor \log_5\left(\frac{4n+3}{43}\right) \right\rfloor}$$

$$T(n) \asymp t \asymp \log n$$

зад (2) $T(n) = 4T(\sqrt[3]{n}) + n$ $\quad = n^{1/3}$

Ще я реша с МТ и със заместване. В час се замотах без МТ да я решаваме, но се оказа по-сложно.

Нека $n = 3^{\log_3 n} \rightarrow n^{1/3} = 3^{\frac{\log_3 n}{3}}$

т.е $T(3^{\log_3 n}) = 4.T(3^{\frac{\log_3 n}{3}}) + 3^{\log_3 n}$

Нека $m = \log_3 n$ и
$\underline{Q(m) = T(3^m)}$

$$Q(m) = 4.\left(\frac{m}{3}\right) + \underbrace{3^m}_{f(m)}$$

$a = 4,\ b = 3,\ k = \log_b a = \log_3 4 \in (1;2)$

Примерно за $\varepsilon = 0{,}1$ то
$m^{k+\varepsilon} = \underline{m^{\log_3 4 + 0{,}1} \alpha\, 3^m}$

Дали е регулярна 3^m?

$$\left(\exists c \in (0,1)\right)\left(\exists m_0 \in \mathbb{N}\right)\left(\forall m \in \mathbb{N}\right)\left[m_0 \le m \Rightarrow a.f\left(\frac{m}{b}\right) \le c.f(m)\right]$$

т.е $4.3^{\frac{m}{3}} \le c.3^m \Leftrightarrow$

$$\frac{4.3^{\frac{m}{3}}}{3^m} \le c \Leftrightarrow$$

$$\frac{4}{3^{\frac{2m}{3}}} \le c$$

за $m \ge 2$, то $\frac{4}{3^{\frac{2m}{3}}} \in (0;1)$

строго монотонно намаляваща редица

Нека $c = \frac{4}{3^{4/3}} = \frac{4}{3\cdot\sqrt[3]{3}} \in (0;1)$

Нека $m_0 = 2$.

Тогава за $\forall m \ge m_0$

ще имаме, че $\frac{4}{3^{\frac{2m}{3}}} \le \frac{4}{3\cdot\sqrt[3]{3}}$ c

Значи е регулярна.

Тогава по III-ти случай МТ, то

$Q(m) \asymp 3^m$, а за $m = \log_3 n$,

то $T(n) \asymp n$.

Зад (2) $T(n) = 2 \cdot T(\sqrt{n}) + 1$

Чрез заместване ще го решим. ($\sqrt{n}$ пречи).

Преставяме $n = 2^{2^m}$ за $m = \lg\lg n$

Тогава $n^{1/2} = (2^{2^m})^{1/2} = 2^{\frac{2^m}{2}} = 2^{2^{m-1}}$

Нека $Q(m) = T(2^{2^m}) = 2T(2^{2^{m-1}}) + 1$

т.е. $Q(m) = 2 \cdot Q(m-1) + 1$. Разписваме $Q(m)$

Тогава $Q(m) = 2^2 \cdot Q(m-2) + 2 + 1 =$

$= 2^3 \cdot Q(m-3) + 4 + 2 + 1 = \ldots =$

$= 2^m \cdot \underbrace{Q(0)}_{\text{const}} + \sum_{i=0}^{m-1} 2^i \asymp 2^m + 2^{m-1} + 2^{m-2} +$

$\ldots + 2 + 1 = \sum_{i=0}^{m} 2^i \overset{\text{ТВ}}{=} 2^{m+1} - 1 \asymp 2^m$

за $m = \lg\lg n$, то $T(n) \asymp \lg n$

Зад (3)

```
f(n: unsigned int): int
 if n=0
   return 0
 a ← n.n.(n+1).(n+1)/4
 for k ← 1 to n-1 do
     a ← a - f(k)
 return a
```

$f(n) = ?$ и $T(n) \asymp ?$

Да разнищим $f(n)$.

$$f(n) = \underbrace{\frac{n^2 \cdot (n+1)^2}{4}}_{\text{на}} - \underbrace{f(1) - f(2) - \ldots - f(n-1)}_{\text{пълна история}} =$$

$$= \frac{n^2 \cdot (n+1)^2}{4} - \sum_{i=1}^{n-1} f(i)$$

Нека го запишем така:

(1) $$f(n) + \underbrace{\sum_{i=1}^{n-1} f(i)} = \sum_{i=1}^{n} f(i) = \frac{n^2 \cdot (n+1)^2}{4}$$

Нека сега заместим $n \mapsto n-1$:

$$f(n-1) + \sum_{i=1}^{n-2} f(i) = \boxed{\sum_{i=1}^{n-1} f(i) = \frac{(n-1)^2 \cdot n^2}{4}} \quad (2)$$

Сега нека заместим (2) в (1)

$$f(n) + \frac{(n-1)^2 \cdot n^2}{4} = \frac{n^2 \cdot (n+1)^2}{4} \iff$$

$$f(n) = \frac{n^2 \left((n+1)^2 - (n-1)^2\right)}{4} \iff$$

$$f(n) = \frac{n^2 \cdot \cancel{4} n}{\cancel{4}} \iff \underline{f(n) = n^3}$$

Сега остава $T(n) \asymp ?$

Можем да го опишем с рекурентно уравнение от този тип:

$$T(n) = \underline{b} + (T(1) + c) + (T(2) + c) + \ldots + (T(n-1) + c) =$$

работа по инициализация

рекурсивни извиквания

всичката работа без рекурсията

$$= b + \sum_{k=1}^{n-1} (T(k) + c) = b + (n-1).c + \sum_{k=1}^{n-1} T(k)$$

Нека приложим свещият трик

$n \mapsto n-1$:

$$T(n-1) = b + (n-2).c + \sum_{k=1}^{n-2} T(k)$$

и сега $T(n) - T(n-1) = c + T(n-1) \Leftrightarrow$

$$T(n) = 2.T(n-1) + c = \underbrace{2.T(n-1)}_{\{2\}_m \text{ корен от хом. част}} + \underbrace{c.1^n}_{\{1\}_m \text{ корен от нехом.}}$$

Общото решение:

$$T(n) = c_1.2^n + c_2.1^n \asymp \underline{2^n}$$

Решаване на рекурентни ур-я, за които Master Th е приложима

Master Theorem

Нека $a \geq 1$ и $b > 1$ са константи и нека $f(n)$ е положителна ф-я.

Нека

$$T(n) = \underbrace{a}_{\text{брой на рекурсивните извиквания}} \cdot T\left(\frac{n}{\underbrace{b}_{\text{брой части, на които делим входните данни}}}\right) + \underbrace{f(n)}_{\text{всичката работа без рекурсията}}$$

/* Различните случаи, които ще опише тази Th не покриват всички възможности! */

Тогава асимптотиката на $T(n)$ е:

* Нека за улеснение $k = \log_b a$, т.е. $n^k = n^{\log_b a}$ е големината на дървото!!!

(сл. 1) Ако $n^{k-\varepsilon} \succeq f(n)$ за някое $\varepsilon > 0$, то $T(n) \asymp n^k$

(сл. 2) Ако $n^k \asymp f(n)$, то $T(n) \asymp n^k . \log n$

дълбочина на дървото е $\log_b n \asymp \log n$

Сл. 3 Ако $n^{k+\varepsilon} \preceq f(n)$ за някое $\varepsilon > 0$ и

f е регулярна ф-я т.е.

$(\forall c \in (0;1))(\exists n_0 \in \mathbb{N})(\forall n \in \mathbb{N})$

$\left[n \geq n_0 \Rightarrow a \cdot f\left(\frac{n}{b}\right) \leq c \cdot f(n) \right]$, то

$T(n) \asymp f(n)$

Сл.★ Специален случай

Ако $f(n) \asymp n^k \cdot \log^t n$ за някаква

константа $t \geq 0$, то

$T(n) \asymp n^k \cdot \log^{t+1} n$

Накратко:

Сл. 1 най-тежко е изчислението за

стигнем до листата

Сл. 2 всички нива са еднакво тежки

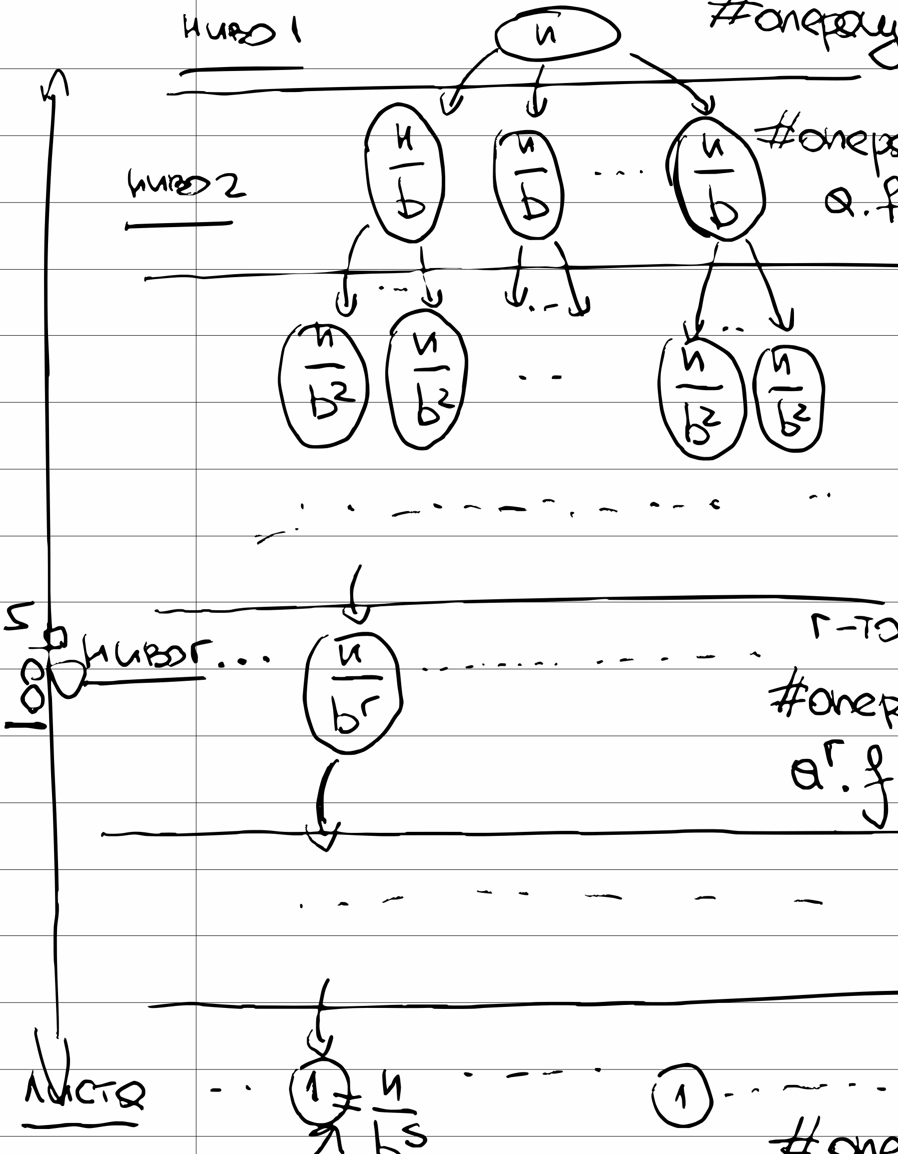
за изчисление (със сложността

нужна да изчислим листата * по

\# нива на дървото)

Сл. 3 най-тежък брой операции

на ниво 1



#операции $f(n)$

#операции $a \cdot f\left(\frac{n}{b}\right)$

r-то влизане

#операции $a^r \cdot f\left(\frac{n}{b^r}\right)$

с-ва логаритми

заради не
се занимаваме
със закръгляне

$\Rightarrow s = \log_b n$ дълбочина на дърво

#операции

$a^s \cdot f(1) =$
$= a^{\log_b n} \cdot f(1) =$
$= n^{\log_b a} \cdot f(1)$

Време е за решаване задачи

Зад 1) $T(n) = 128 \cdot T\left(\frac{n}{2}\right) + n^3 \cdot \log n$

$a = 128,\ b = 2,\ k = \log_2 128 = \log_2 2^7 = 7$

В кой случай сме, ами в случай I

за $\varepsilon = 0{,}2$ примерно, то

$n^{k-\varepsilon} = n^{7-0{,}2} = n^{6{,}8} > f(n) = n^3 \cdot \log n$

защото L'Hôpital

$\lim\limits_{n\to\infty} \frac{n^{6{,}8}}{n^3 \cdot \log n} = \lim\limits_{n\to\infty} \frac{n^{3{,}8}}{\log n} = \lim\limits_{n\to\infty} \frac{3{,}8 \cdot n^{2{,}8}}{1/n} =$

$= 3{,}8 \cdot \lim\limits_{n\to\infty} n^{3{,}8} = \infty$

Така $T(n) \asymp n^k = n^7$

Зад 2) $T(n) = 10000 \cdot T\left(\frac{n}{10}\right) + n^4$

$a = 10000,\ b = 10$

$k = \log_{10} 10^4 = 4$

т.е. $n^k \asymp n^4 \rightarrow T(n) \asymp n^k \cdot \log n =$

II ри случай $= n^4 \cdot \log n$

зад. 3 $T(n) = 3,5 \cdot T\left(\frac{n}{2}\right) + n^5$ всяка ф-я на-голяма от n^k е асимптотично отговор

$a = 3,5$; $b = 2$

$k = \log_2 3,5 \in (1; 2)$ Случай 3

за $\varepsilon = 0,1$

$f(n) = n^5 > n^{k-0,1} = n^{\log_2 3,5 + 0,1}$

$$\lim_{n\to\infty} \frac{n^5}{n^{\log_2 3,5 + 0,1}} = \lim_{n\to\infty} n^{4,9 - \log_2 3,5} = \infty \quad 4,9 - \log_2 3,5 > 0$$

Остава условие за регулярност

$\forall c \in (0; 1)\ \exists n_0\ \forall n \geq n_0$

$a \cdot f\left(\frac{n}{b}\right) \leq c \cdot f(n)$?

$3,5 \cdot \left(\frac{n}{2}\right)^5 \leq c \cdot n^5 \ /\!/ : n^5 \neq 0$ положит. ф-я за дост. големи n

$\frac{3,5}{2^5} \leq c \Leftrightarrow \frac{3,5}{32} \leq c$

Ако $c = \frac{3,5}{32} = \frac{7/2}{32} = \frac{7}{64} \in (0; 1)$

Тогава за $n_0 = 1$, то $\forall n \geq n_0$ е изп.

$f(n) = n^5$ е регулярна

Така $T(n) \simeq f(n)$.

Зад. 4 BinarySearch(A[1...n]: sorted array of ints, key: int): int

```
if n ≤ 0 then
    return -1
value ← A(⌈n/2⌉)
if value = key then
    return key
else if value < key
    return BinarySearch(A[1...⌈n/2⌉-1], key)
return BinarySearch(A[⌈n/2⌉+1...n], key)
```

Да го опишем:

$T(n) = 1 \cdot T\left(\frac{n}{2}\right) + 1$

разклонение на рекурсията (т.е. само в/у 1 част)

всеки път делим масива на две

работа за всичко без рекурсията

$a = 1,\ b = 2,\ k = \log_2 1 = 0$

$f(n) = 1$ и $n^k = n^0 = 1$

т.е. $n^k \simeq f(n) \rightarrow$ 2-ри случай МТ значи $T(n) \asymp n^k \cdot \log n = \log n$.

зад 5 $sum(A[1..n]: array\ of\ nums): num$

```
if n=0 then
   return 0
else if n = 1 then
   return A[1]
return sum(A[1..⌊n/2⌋]) +
          sum(A[⌊n/2⌋+1...n])
```

$T(n) = 2T\left(\frac{n}{2}\right)+1, \; k = \log_2 2 = 1$

$f(n) = n^0 = 1 \propto n^1 = n^k$ за $\varepsilon = 0{,}1$ примерно става

$1^{ви}$ случай на MT

значи $T(n) \asymp n^k = n$

зад 6 $T(n) = 1 \cdot T\left(\frac{n}{4}\right) + \log n$

$a = 1, \; b = 4, \; k = \log_4 1 = 0$

$f(n) = \log n \asymp n^0 \cdot \log^t n$ за $t = 1$

в специален случай сме, затова

$T(n) \asymp \log^2 n$

HW / a) $T(n) = T\left(\frac{9 \cdot n}{10}\right) + n$

b) $T(n) = 16 \cdot T\left(\frac{n}{2}\right) + n^2$

c) $T(n) = 8 \cdot T\left(\frac{n}{2}\right) + n^3$

d) $T(n) = T(n-1) + \sqrt{n}$

e) $T(n) = 2 \cdot T(\sqrt[4]{n}) + \log(\sqrt[4]{n})$

a) $a = 1$, $b = \frac{10}{9}$, $k = \log_b a = 0$

$f(n) = n$, $n^k = n^0$

за $\varepsilon = 0{,}1$, то $f(n) \succ n^{0+\varepsilon} = n^{0{,}1}$

Дали $f(n)$ е регулярна.

$\exists c \in (0;1)\ \exists n_0\ \forall n \geq n_0\quad a \cdot f\left(\frac{n}{b}\right) \leq c \cdot f(n)\ (*)$

$f\left(\frac{n \cdot 9}{10}\right) \leq c \cdot f(n) \Leftrightarrow$

$\frac{n \cdot 9}{10} \leq c \cdot n \quad // : n$

$\frac{9}{10} \leq c$, за $c = \frac{9}{10} \in (0;1)$ и $n_0 = 1$, то

$\forall n \geq n_0$ $(*)$ е в сила тако $T(n) \asymp f(n)$

по $\overline{\text{III}}$ сл. МТ.

b) $T(n) = 16 \cdot T\left(\frac{n}{2}\right) + n^2$

$a = 16 = 2^4$; $b = 2$; $k = \log_b a = 4$

$n^k = n^4$

$f(n) = n^2$

Тогава за $\varepsilon = 0{,}1$, то

$n^2 = f(n) \prec n^{k-\varepsilon} = n^{3{,}9}$

Тоест по I^ви случай МТ $T(n) \asymp n^k = n^4$

c) $T(n) = 8 \cdot T\left(\frac{n}{2}\right) + n^3$

$a = 8 = 2^3$; $b = 2$; $k = \log_b a = 3$

$n^k = n^3$; $f(n) = n^3$

$f(n) \asymp n^k \xRightarrow[\text{МТ}]{\text{II}^{\text{ри}}\text{ сл.}} T(n) \asymp n^k \cdot \log n =$

$= n^3 \cdot \log n$

d) $T(n) = T(n-1) + \sqrt{n}$

С развиване:

$T(n-1) = T(n-2) + \sqrt{(n-1)}$

$T(n) = T(n-2) + \sqrt{(n-1)} + \sqrt{n} = \ldots =$

$= T(0) + \sum_{i=1}^{n} i^{1/2} \asymp n^{1/2+1} = n^{3/2}$

$k = 1/2 > -1$

e) $T(n) = 2.T(\sqrt[4]{n}) + \log(\sqrt[4]{n})$

Със заместване:

Нека $n = 4^{4^m}$ за $m = \log_4 \log_4 n$.

Тогава

$T(4^{4^m}) = 2.T((4^{4^m})^{1/4}) + \log((4^{4^m})^{1/4}) =$

$= 2.T(4^{4^{m-1}}) + \log(4^{4^{m-1}})$

Нека $S(m) = 2\,S(m-1) + \log(m-1)$.

Сега чрез развиване имаме, че

$S(m-1) = 2.S(m-2) + \log(m-2)$

$S(m) = 4.S(m-2) + 2.\log(m-2) + \log(m-1) =$

$= \dots = 2^m.S(0) + \sum_{i=1}^{m} 2^{m-i}.\log(i-1) \overset{(*)}{=}$

$\sum_{i=1}^{m} 2^{m-i}.\log(i-1) = 2^m.\sum_{i=1}^{m} \frac{\log(i-1)}{2^i} \leq$

$2^m.\sum_{i=1}^{m} \log(i-1) = 2^m.\log(\underbrace{0.1.2\dots(m-1)}_{0}) =$

$\uparrow$ с-во $\log a + \log b = \log(a.b)$

$= 2^m.\log 0 = 0$

Тогава $\overset{(*)}{\leq} 2^m.S(0) \asymp 2^m$

$m = \log_4 \log_4 n \asymp \log_2 \log_2 n$ и така

$T(n) \asymp 2^{\log_2 \log_2 n} \asymp \log n$

Some highlights

* Всяко рекурентно ур-е е рекурсивен алго. и реализира някаква ф-я т.е. определя някаква числова редица. Решаването на рек. ур-е означава да намерим друг еквив. израз (формула) за съшата ф-я, който да е затворена ф-ла. Ползите са, че решението е по-бърз начин за изчисляване на ф-ята и, ако е достатъчно просто, бързо можем да прецениим каква е асимптотиката на нарастването на ф. и ! НЯМА УНИВЕРСАЛЕН МЕТОД ЗА РЕШАВАНЕТО ИМ!
* Сложността на рекурсивен алго. се описва от рекурентно ур-е по правило.
* Извод: рекурентното ур-е е рекурсивен алго., но той е съвършено различен от алгоритъма, чиято сложност описва.

★ Master Th е приложима само за рекурентни ур-я с една поява вдясно.

★ При изследване на сложността на "естествени" алго., то ако:

- са изградени по схема Divide-n-conquer, то ф-ята на нарастване е деление
- прилагат принципа на пълното изчерпване на възможности, то ф-ята на нарастване е изваждане (може да има и други алго., които не са изградени на този принцип и ф-ята им на нарастване да е изваждане).

★ Рекурсивните ур-я, които ще разглеждаме няма да зависят от началните условия, затова ги пропускаме.

★ Master Th не обхваща всички възможни случаи! Може

Ф-ята описваща големината на дървото на рекурсията за е асимптотично несравнима с $f(n)$ или има и други "дупки" м/у случаите, в които можем да попаднем.