

Úvod do složitosti

Jan Konečný

12. listopadu 2013

Dosud: vyčíslitelnost = které problémy lze řešit na stroji a které ne?

Od teď: složitost = které z řešitelných problémů lze řešit efektivně a které ne?

Agenda

- Složitost – časová a paměťová
- Asymptotická složitost
- Třídy složitosti
- Třídy P, NP

Složitost – už znáte už z předchozích kurzů

Definice

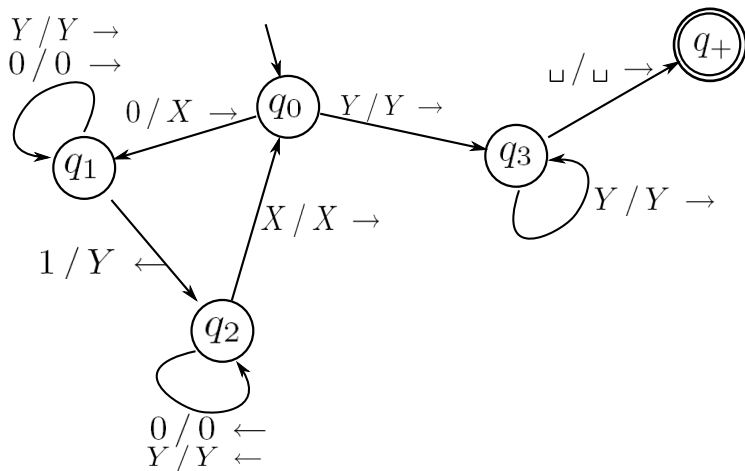
T – TS, zastavuje pro každé slovo.

Časová složitost = funkce $f : N \rightarrow N_0$, kde $f(n)$ je maximální počet kroků při výpočtu nad jakýmkoli vstupem délky n .

Paměťová složitost = funkce $f : N \rightarrow N$, kde $f(n)$ je maximální počet políček použitých při výpočtu nad jakýmkoli vstupem délky n .

? vztah mezi časovou a paměťovou složitostí?

Časová složitost



TS:

určíme: $f(1), f(2), f(3), f(4), \dots$

Časová složitost

$f(1): 0: (q_0, 0, 1) \vdash (q_1, X, 2) \vdash (q_-, X, 3)$ (2 kroky)

$1: (q_0, 1, 1) \vdash (q_-, 1, 2)$ (1 krok)

$f(1) = 2$

$f(2): 00: (q_0, 00, 1) \vdash (q_1, X0, 2) \vdash (q_1, X0, 2) \vdash (q_-, X0, 4)$ (3 kroky)

$01: (q_0, 01, 1) \vdash (q_1, X1, 2) \vdash (q_2, XY, 1) \vdash (q_0, XY, 2) \vdash (q_3, XY, 3) \vdash (q_+, XY, 4)$ (5 kroků)

$10: (q_0, 10, 1) \vdash (q_-, 10, 2)$ (1 krok) $11: (q_0, 11, 1) \vdash (q_-, 11, 2)$ (1 krok)

$f(2) = 5$

Časová složitost

TS:

$$f(n) = \frac{n^2}{2} + n + 1 \text{ (pro } n \text{ sudé)}$$

$$f(n) = \frac{n^2}{2} + n + \frac{1}{2} \text{ (pro } n \text{ liché)}$$

Asymptotická složitost

Definice

$$f, g : N \rightarrow R$$

$g(n)$ je **asymptotická horní hranice** $f(n)$, když existují $c, n_0 \in N$, t.ž. pro každé $n \geq n_0$:

$$f(n) \leq c \cdot g(n)$$

Notace: $f(n) = \mathcal{O}(g(n))$

Asymptotická složitost

Příklad:

$$f_1(n) = 5n^3 + 2n^2 + 22n + 6$$

$$f_1(n) = \mathcal{O}(n^3) \text{ pro } c = 6, n_0 = 10.$$

Ale taky:

$$f_1(n) = \mathcal{O}(n^4)$$

$$f_1(n) = \mathcal{O}(n^5)$$

$$f_1(n) = \mathcal{O}(2^n)$$

...

Motivační tabulka

$f(n)$	20	40	60	80	100	1000
n	20ns	40ns	60ns	80ns	100ns	$1\mu s$
$n \log(n)$	86ns	$0.2\mu s$	$0.35\mu s$	$0.5\mu s$	$0.7\mu s$	$10\mu s$
n^2	$0.4\mu s$	$1.6\mu s$	$3.6\mu s$	$6.4\mu s$	$10\mu s$	1 ms
n^4	0.16ms	2.5 ms	13 ms	41 ms	0.1 s	16.8min
2^n	1 ms	16.8min	36.6let			
$n!$	77let					

Motivační tabulka 2

vstup délky 100 jsme schopni zpracovat... zrychlíme stroj:

$f(n)$	n	$n \log(n)$	n^2	n^4	2^n	$n!$
$1\times$	100	100	100	100	100	100
$100\times$	10000	5362	1000	316	106	100
$1000\times$	100000	43150	3162	562	109	101

Efektivně řešitelné = řešitelné s polynomiální časovou složitostí.

Bacha! Nebrat to dogmaticky:

$$f(n) = 2^{100}n \text{ vs. } f(n) = 2^{n^{0.0001}}$$

Definice

$\text{TIME}(t(n)) = \{L \mid L \text{ rozhodovaný TS v čase } \mathcal{O}(t(n))\}.$

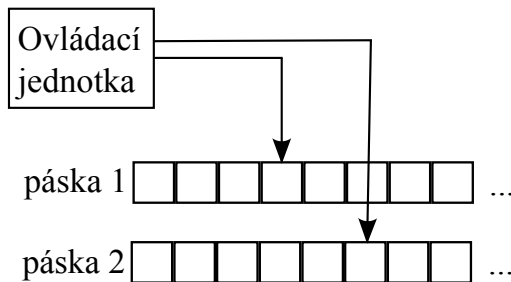
$\text{SPACE}(s(n)) = \{L \mid L \text{ rozhodovaný TS v paměti } \mathcal{O}(s(n))\}.$

$$\{0^k 1^k \mid k > 0\} \in \text{TIME}(n^2)$$

Složitost a rozšíření modelu TS?

- TS s oboustranně nekonečnou páskou,
- vícepáskový TS,
- nedeterministický TS.

vícepáskový TS

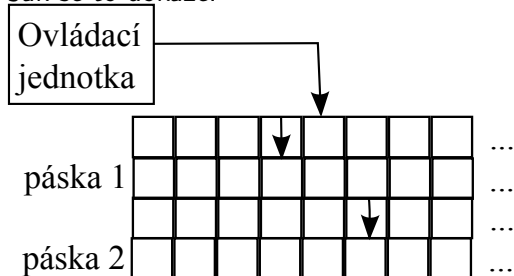


Časová: vícepáskový TS časová složitost $t(n)$... TS časová složitost $\mathcal{O}(t^2(n))$

Paměťová: nemění se.

vícepáskový TS

Jak se to dokáže:



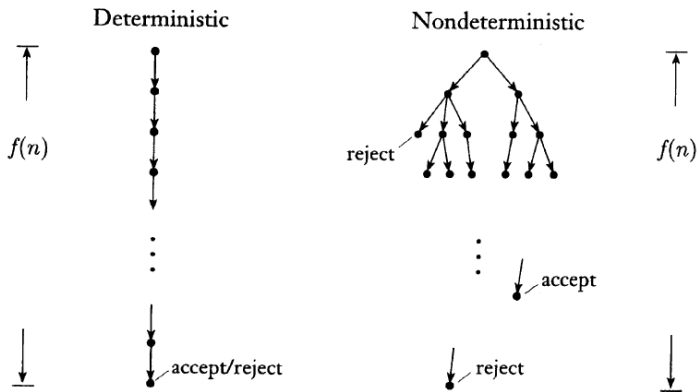
Složitost nedeterministického TS:

Definice

NTS rozhodující jazyk.

Časová složitost = funkce $f : N \rightarrow N_0$, kde $f(n)$ je maximální počet kroků použitých v jakékoli větvi jeho výpočtu při jakémkoli vstupu délky n .

Paměťová složitost = funkce $f : N \rightarrow N$, kde $f(n)$ je maximální počet políček použitých v jakékoli větvi jeho výpočtu při jakémkoli vstupu délky n .



Časová: NTS časová složitost $t(n)$... TS časová složitost $2^{\mathcal{O}(t(n))}$.

Paměťová: NTS paměťová složitost $t(n)$... TS časová složitost $O(t^2(n))$.

Definice

$\text{NTIME}(t(n)) = \{L \mid L \text{ rozhodovaný NTS v čase } \mathcal{O}(t(n))\}.$

$\text{NSPACE}(s(n)) = \{L \mid L \text{ rozhodovaný NTS v paměti } \mathcal{O}(s(n))\}.$

Třídy P a NP

Definice

$$P = \bigcup \{ \text{TIME}(n^k) \mid k \in \mathbb{N} \}$$

$$NP = \bigcup \{ \text{NTIME}(n^k) \mid k \in \mathbb{N} \}$$

Příklad: $\{0^k 1^k \mid k > 0\} \in P$

Definice

Verifikátor pro jazyk A je TS V , t.ž.

$$A = \{w \mid [w, c] \in V \text{ pro nějaké } c \in \Sigma^*\}$$

.

Říkáme, že c je certifikát; V verifikuje A .

Pokud V pracuje v čase t , říkáme, že A je verifikovaný v čase t (s tím, že se uvažuje jen délka slova w při určování složitosti)

Definice

Jazyky verifikované v polynomiálním čase = NP.

$P \subseteq NP$

?

- $P \subseteq NP$
- $P = NP$

Pikantnost: P /NP Policy

The JACM frequently receives submissions purporting to solve a long-standing open problem in complexity theory, such as the P/NP problem. Such submissions tax the voluntary editorial and peer-reviewing resources used by the JACM, by requiring the review process to identify the errors in them. The JACM remains open to the possibility of eventual resolution of P/NP and related questions, and continues to welcome submissions on the subject. However, to mitigate the burden of repeated resubmissions of incremental corrections of errors identified during editorial review, the JACM has adopted the following policy:

No author may submit more than one paper on the P/NP or related long-standing questions in complexity theory in any 24 month period, except by invitation of the Editor-in-Chief. This applies to resubmissions of previously rejected manuscripts.

Co nás teda čeká příště?

- Problémy, které jsou P,
- Problémy, které jsou NP,
- Vztah P a NP,
- NP-úplné problémy

Pojmy k zapamatování:

Časová a paměťová složitost TS, časová a paměťová složitost NTS, asymptotická složitost, \mathcal{O} -notace, třídy složitosti, třída P, třída NP, třída P-SPACE, třída NP-SPACE.

- 1 Ukažte, že P je uzavřená vzhledem k: obrácení, sjednocení, konkatenaci, uzávěru, doplňku.
- 2 Ukažte, že NP je uzavřená vzhledem k: obrácení, sjednocení, konkatenaci, uzávěru.