Základy složitosti a vyčíslitelnosti

NTIN090

Petr Kučera

2022/23 (4. přednáška)

Převoditelnost a úplnost

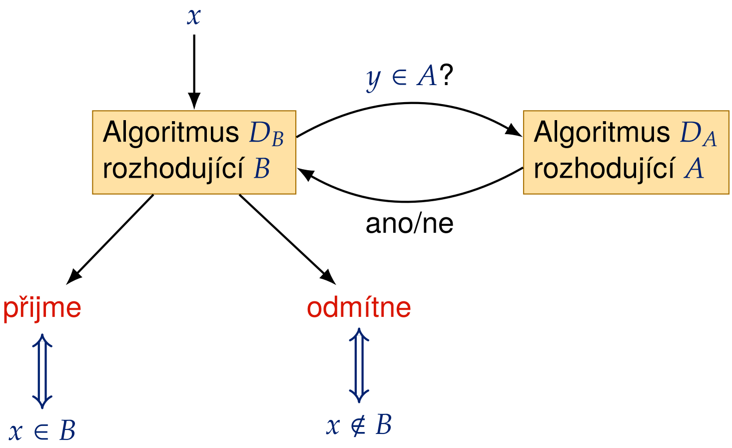
Jak ukazovat nerozhodnutelnost?

Chceme ukázat, že A je nerozhodnutelný jazyk.

1. Vybereme si jiný nerozhodnutelný jazyk B
   - například $B = L_u$
2. Sporem předpokládáme: Máme algoritmus D_A, který rozhoduje A
3. Popíšeme algoritmus D_B, který rozhoduje B
   - D_B může volat D_A jako podprogram
4. Dostáváme spor s nerozhodnutelností B
5. Ukázali jsme, že A není rozhodnutelný

Turingovská převoditelnost (princip)

$B \leq_T A$ = „pomocí A umíme rozhodnout B“



Turingovská převoditelnost

Jazyk $B \subseteq \Sigma^*$ je Turingovsky převoditelný na jazyk $A \subseteq \Sigma^*$ ($B \leq_T A$), pokud existuje algoritmus (Turingův stroj) D_B, který

- rozhoduje B
- může pokládat dotazy typu „$y \in A$?“
  - pro libovolný řetězec $y \in \Sigma^*$
- Tyto dotazy jsou okamžitě správně zodpovězeny (ano/ne)
- Počet dotazů není omezen

- A rozhodnutelný $\implies$ B rozhodnutelný
- B nerozhodnutelný $\implies$ A nerozhodnutelný
- $A \leq_T \overline{A}$ pro každý jazyk A

Problém zastavení

Problém zastavení (Halting problem)

Instance: Kód Turingova stroje M a vstup x.

Otázka: Zastaví se výpočet Turingova stroje M nad vstupem x, tedy $M(x)\downarrow$?

- Odpovídá jazyku $\mathrm{HALT} = \{\langle M, x\rangle \mid M(x)\downarrow\}$

Věta

Jazyk HALT je částečně rozhodnutelný, ale není rozhodnutelný.

Částečná rozhodnutelnost HALT

$$\mathrm{HALT} = \{\langle M, x\rangle \mid M(x)\downarrow\}$$

- Jazyk HALT je částečně rozhodnutelný
- Plyne z existence univerzálního stroje $\mathcal{U}$
- HALT je přijímán následujícím strojem $\mathcal{H}$

Výpočet $\mathcal{H}$ se vstupem $\langle M, x\rangle$

1. Simuluj $M(x)$
2. Přijmi

Nerozhodnutelnost HALT

$$\mathrm{HALT} = \{\langle M, x\rangle \mid M(x)\downarrow\}$$

- Jazyk HALT je nerozhodnutelný
- Sporem: Nechť $\mathcal{H}$ je stroj rozhodující HALT
  - $\mathrm{HALT} = L(\mathcal{H})$ a
  - $\mathcal{H}(\langle M, x\rangle)\downarrow$ pro každý Turingův stroj M a vstup x
- Popíšeme Turingův stroj M_u, který rozhoduje

$$L_u = \{\langle M, x\rangle \mid x \in L(M)\}$$

Stroj M_u rozhodující L_u s pomocí HALT

Výpočet stroje M_u se vstupem $\langle M, x\rangle$

Sestroj Turingův stroj M', který se řídí následujícím algoritmem

> **Výpočet M' se vstupem y**
>
> Pusť $M(y)$
> **if** M zamítl **then**
>   vstup do nekonečné smyčky

// Položíme dotaz jazyku HALT
if $\langle M', x\rangle \in \mathrm{HALT}$ **then**
    přijmi
else
    odmítni

Nerozhodnutelnost HALT

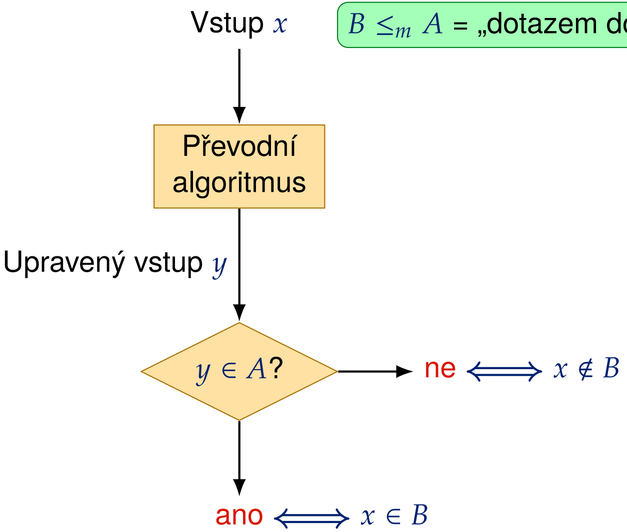
Ukázali jsme, že $L_u \leq_T \mathrm{HALT}$

L_u není rozhodnutelný $\Longrightarrow$ HALT není rozhodnutelný

- Program M_u je velmi specifický
  - Orákula HALT se ptá jen jednou na konec
  - Odpověď dotazu HALT je přímo odpovědí M_u
- Program ukazuje, že L_u je m-převoditelný na HALT

m-převoditelnost (princip)

$B \leq_m A$ = „dotazem do A rozhodneme B“

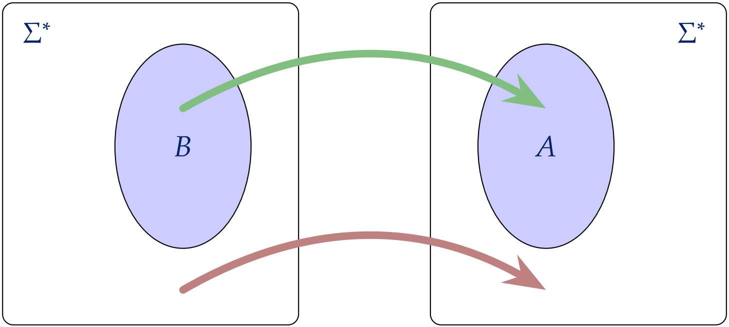


m-převoditelnost (definice)

Definice

Jazyk B je m-**převoditelný** na jazyk A, pokud existuje totální vyčíslitelná funkce f splňující

$$(\forall x \in \Sigma^*)[x \in B \Leftrightarrow f(x) \in A]$$



m-převoditelnost (definice)

Definice

Jazyk B je m-převoditelný na jazyk A ($B \leq_m A$), pokud existuje totální vyčíslitelná funkce f splňující

$$(\forall x \in \Sigma^*)[x \in B \Leftrightarrow f(x) \in A]$$

- $\leq_m$ je reflexivní a tranzitivní relace (kvaziuspořádání).
- Pokud $A \leq_m B$ a B je (částečně) rozhodnutelný jazyk, pak totéž lze říct o A.

$$\underbrace{\overline{L_u}}_{\text{není částečně rozhodnutelný}} \quad \not\leq_m \quad \underbrace{L_u}_{\text{částečně rozhodnutelný}}$$

m-převoditelnost (reflexivita)

Lemma (Reflexivita m-převoditelnosti)

Pro každý jazyk A platí $A \leq_m A$

Důkaz.

- Identita $\mathrm{id}(x) = x$ je totální algoritmicky vyčíslitelná funkce
- Pro každý řetězec $x \in \Sigma^*$ platí

$$x \in A \Leftrightarrow \mathrm{id}(x) \in A$$

$\square$

m-převoditelnost (tranzitivita)

Lemma (Tranzitivita m-převoditelnosti)

Pro každé jazyky A, B a C platí $A \leq_m B \wedge B \leq_m C \implies A \leq_m C$

Důkaz.

- $A \leq_m B$ pomocí funkce g
- $B \leq_m C$ pomocí funkce h
- Definujme funkci $f(x) = h(g(x))$
  - f je totální algoritmicky vyčíslitelná funkce
- Pro každý řetězec $x \in \Sigma^*$ platí

$$x \in A \underbrace{\iff}_{A \leq_m B} g(x) \in B \underbrace{\iff}_{B \leq_m C} h(g(x)) \in C \quad \underbrace{\iff}_{f(x)=h(g(x))} \quad f(x) \in C$$

- $A \leq_m C$ pomocí funkce f

□

m-převoditelnost porovnává obtížnost

Lemma

Nechť B a A jsou dva jazyky, pro něž platí, že $B \leq_m A$.

1. *A rozhodnutelný $\Longrightarrow$ B je rozhodnutelný*
2. *A částečně rozhodnutelný $\Longrightarrow$ B je částečně rozhodnutelný*

- Předpokládejme, že M_A je TS, který přijímá/rozhoduje A
- Popíšeme TS M_B, který přijímá/rozhoduje B

Výpočet stroje M_B se vstupem x

$y \leftarrow f(x)$ `// ` f `ukazuje, že` $B \leq_m A$
Pusť $M_A(y)$
if M_A přijal **then**
    přijmi
else
    odmítni

m-převoditelnost a nerozhodnutelnost

Lemma

Nechť B a A jsou dva jazyky, pro něž platí, že $B \leq_m A$.

1. *A je rozhodnutelný $\Longrightarrow$ B je rozhodnutelný*
2. *A je částečně rozhodnutelný $\Longrightarrow$ B je částečně rozhodnutelný*

Důsledek

Nechť B a A jsou dva jazyky, pro něž platí, že $B \leq_m A$.

1. *B je nerozhodnutelný $\Longrightarrow$ A je nerozhodnutelný*
2. *B není částečně rozhodnutelný $\Longrightarrow$ A není částečně rozhodnutelný*

Převod L_u na HALT

- $L_u = \{\langle M, x\rangle \mid x \in L(M)\}$
- $\mathrm{HALT} = \{\langle M, x\rangle \mid M(x)\downarrow\}$

Lemma

$L_u \leq_m \mathrm{HALT}$

Idea:

- Definujeme $f(\langle M, x\rangle) = \langle M', x\rangle$ tak, aby platilo pro každý vstup $x \in \Sigma^*$

$$M \text{ přijme } x \iff M'(x)\downarrow$$

- Popíšeme úpravu M na M'
- Algoritmus počítající f provede tuto transformaci

$L_u \leq_m \text{HALT}$

$L_u = \{\langle M, x\rangle \mid x \in L(M)\}$

$\text{HALT} = \{\langle M, x\rangle \mid M(x)\downarrow\}$

Ke stroji M definujeme stroj M' takto

Výpočet M' se vstupem y
Pusť $M(y)$ **if** M zamítl **then** vstup do nekonečné smyčky

- Pro každý řetězec $y \in \Sigma^*$ platí:

$$y \in L(M) \iff M'(y)\downarrow$$

- Pro každou dvojice M a x platí

$$\langle M, x\rangle \in L_u \iff x \in L(M) \iff M'(x)\downarrow \iff \langle M', x\rangle \in \text{HALT}$$

$L_u \leq_m \mathrm{HALT}$

- Funkce $f(\langle M, x\rangle) = \langle M', x\rangle$
  - je totální algoritmicky vyčíslitelná a
  - ukazuje, že $L_u \leq_m \mathrm{HALT}$

Neprázdnost jazyka

Problém neprázdného jazyka

Instance:	Kód Turingova stroje M
Otázka:	Je $L(M) \neq \emptyset$?

$$\mathrm{NE} = \{\langle M \rangle \mid L(M) \neq \emptyset\}$$

Věta

Jazyk NE je částečně rozhodnutelný ale není rozhodnutelný

- Částečnou rozhodnutelnost jsme již ukázali dříve
- Nerozhodnutelnost ukážeme tím, že $L_u \leq_m \mathrm{NE}$

$L_u \leq_m \mathrm{NE}$

$L_u = \{\langle M, x\rangle \mid x \in L(M)\}$

$\mathrm{NE} = \{\langle M\rangle \mid L(M) \neq \emptyset\}$

K dvojici M a x definujeme Turingův stroj M'

Výpočet M' se vstupem y

```
Pusť M(x)    // M' ignoruje svůj vstup!
if M přijal then
 | přijmi
else
 └ odmítni
```

- Platí

$$L(M') = \begin{cases} \Sigma^* & x \in L(M) \\ \emptyset & x \notin L(M) \end{cases}$$

$L_u \leq_m \mathrm{NE}$

- Pro každý kód dvojice $\langle M, x \rangle$ platí

$$\langle M, x \rangle \in L_u \iff x \in L(M) \iff L(M') \neq \emptyset \iff \langle M' \rangle \in \mathrm{NE}$$

- Funkce $f(\langle M, x \rangle) = \langle M' \rangle$
  - je totální algoritmicky vyčíslitelná a
  - ukazuje, že $L_u \leq_m \mathrm{NE}$

m-úplné problémy

Definice

Jazyk A je m-**úplný**, pokud

1. A je částečně rozhodnutelný a
2. pro každý částečně rozhodnutelný jazyk B platí, že $B \leq_m A$

- Ty „nejtěžší“ z částečně rozhodnutelných jazyků
- m-úplnost implikuje nerozhodnutelnost
- Pokud $A \leq_m B$, B je částečně rozhodnutelný jazyk a A je m-úplný jazyk, pak B je též m-úplný
  - Plyne z tranzitivity m-převoditelnosti

m-úplnost univerzálního jazyka

Věta

Jazyk $L_u = \{\langle M, x\rangle \mid x \in L(M)\}$ *je* m*-úplný.*

Důkaz.

- Částečná rozhodnutelnost plyne z existence univerzálního TS
- Nechť A je libovolný částečně rozhodnutelný jazyk
- Existuje Turingův stroj M přijímající A ($A = L(M)$)
- Funkce $f(x) = \langle M, x\rangle$ je totální algoritmicky vyčíslitelná a platí

$$x \in A \iff x \in L(M) \iff \langle M, x\rangle \in L_u \iff f(x) \in L_u$$

- Funkce f tedy ukazuje $A \leq_m L_u$

$\square$

Další úplné jazyky

Důsledek

Jazyky

$$\mathrm{HALT} = \{\langle M, x\rangle \mid M(x){\downarrow}\}$$
$$\mathrm{NE} = \{\langle M\rangle \mid L(M) \neq \emptyset\}$$

jsou m-úplné

Důkaz.

- Oba jazyky jsou částečně rozhodnutelné
- Ukázali jsme, že $L_u \leq_m \mathrm{HALT}$ a $L_u \leq_m \mathrm{NE}$
- m-úplnost plyne z tranzitivity m-převoditelnosti

□

Riceova věta

Riceova věta

Věta (Riceova věta)

Nechť $\mathcal{C}$ je třída částečně rozhodnutelných jazyků a položme

$$L_{\mathcal{C}} = \{\langle M\rangle \mid L(M) \in \mathcal{C}\}$$

Jazyk $L_{\mathcal{C}}$ rozhodnutelný, právě když je třída $\mathcal{C}$ triviální, tedy buď je prázdná nebo obsahuje všechny částečně rozhodnutelné jazyky.

Problém, v němž se ptáme, zda daný program přijímá jazyk s danou netriviální vlastností P, je nerozhodnutelný.

Riceova věta (důsledky)

Nerozhodnutelnost následujících jazyků plyne z Riceovy věty

- $\mathrm{NE} = \{\langle M\rangle \mid L(M) \neq \emptyset\}$
  - C je třída neprázdných částečně rozhodnutelných jazyků
- $\mathrm{Fin} = \{\langle M\rangle \mid L(M)$ je konečný jazyk$\}$
  - C je třída konečných jazyků
- $\mathrm{Inf} = \{\langle M\rangle \mid L(M)$ je nekonečný jazyk$\}$
  - C je třída nekonečných částečně rozhodnutelných jazyků
- $\mathrm{Dec} = \{\langle M\rangle \mid L(M)$ je rozhodnutelný jazyk$\}$
  - C je třída rozhodnutelných jazyků
- $\mathrm{Reg} = \{\langle M\rangle \mid L(M)$ je regulární jazyk$\}$
  - C je třída regulárních jazyků
- $\mathrm{Primality} = \{\langle M\rangle \mid L(M) = \mathrm{PRIME}\}$
  - $\mathrm{PRIME} = \{\langle p\rangle \mid p$ je prvočíslo$\}$
  - $C = \{\mathrm{PRIME}\}$
- $\mathrm{Hello} = \{\langle M\rangle \mid M$ přijímá řetězec $\texttt{Hello}\}$

Riceova věta (nepoužitelnost)

Věta (Riceova věta)

Nechť $\mathcal{C}$ *je třída částečně rozhodnutelných jazyků a položme*

$$L_{\mathcal{C}} = \{\langle M\rangle \mid L(M) \in \mathcal{C}\}$$

Jazyk $L_{\mathcal{C}}$ *rozhodnutelný, právě když je třída* $\mathcal{C}$ *triviální, tedy buď je prázdná nebo obsahuje všechny částečně rozhodnutelné jazyky.*

NELZE použít na jazyky

$$\begin{aligned} L_u &= \{\langle M, x\rangle \mid x \in L(M)\} \\ \mathrm{HALT} &= \{\langle M, x\rangle \mid M(x)\!\downarrow\} \\ \mathrm{DIAG} &= \{\langle M\rangle \mid \langle M\rangle \notin L(M)\} \end{aligned}$$

Riceova věta (důkaz)

Věta (Riceova věta)

Nechť $\mathcal{C}$ je třída částečně rozhodnutelných jazyků a položme

$$L_{\mathcal{C}} = \{\langle M\rangle \mid L(M) \in \mathcal{C}\}$$

Jazyk $L_{\mathcal{C}}$ rozhodnutelný, právě když je třída $\mathcal{C}$ triviální, tedy buď je prázdná nebo obsahuje všechny částečně rozhodnutelné jazyky.

- Nechť třída $\mathcal{C}$ je prázdná

$\Longrightarrow \quad L_{\mathcal{C}} = \emptyset$

$\Longrightarrow \quad L_{\mathcal{C}}$ je rozhodnutelný

- Nechť třída $\mathcal{C}$ obsahuje všechny částečně rozhodnutelné jazyky

$\Longrightarrow \quad L_{\mathcal{C}} = \Sigma^*$

$\Longrightarrow \quad L_{\mathcal{C}}$ je rozhodnutelný

Riceova věta (důkaz)

- Dále předpokládáme, že $\mathcal{C}$ je netriviální
- Ukážeme, že
  - $L_u \leq_m L_{\mathcal{C}}$ pokud $\mathcal{C}$ neobsahuje prázdný jazyk
  - $L_u \leq_m \overline{L_{\mathcal{C}}}$ pokud $\mathcal{C}$ obsahuje prázdný jazyk

Riceova věta (důkaz)

- Předpokládejme, že $\mathcal{C}$ neobsahuje prázdný jazyk
  - tedy $\emptyset \notin \mathcal{C}$
- Zvolíme Turingův stroj M_1 takový, že $L(M_1) \in \mathcal{C}$
  - Speciálně $L(M_1) \neq \emptyset$
- Popíšeme totální vyčíslitelnou funkci $f(\langle M, x\rangle) = \langle M'\rangle$ splňující

$$L(M') = \begin{cases} L(M_1) & x \in L(M) \\ \emptyset & x \notin L(M) \end{cases}$$

- Platí tedy

$$x \in L(M) \iff L(M') \in \mathcal{C} \iff f(\langle M, x\rangle) = \langle M'\rangle \in L_{\mathcal{C}}$$

- Funkce f ukazuje, že $L_u \leq_m L_{\mathcal{C}}$

Riceova věta (důkaz)

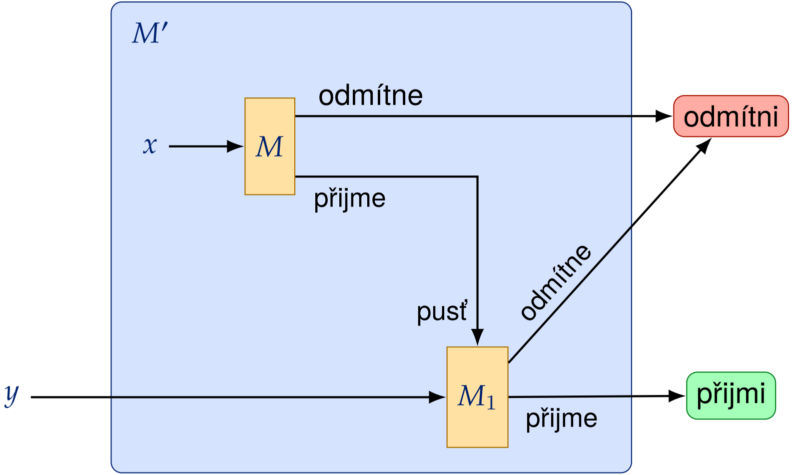
K dvojici M a x definujeme Turingův stroj M'

Výpočet M' se vstupem y

Simuluj $M(x)$
if M přijal **then** // $x \in L(M)$
    Simuluj $M_1(y)$
    if M_1 přijal **then**
        přijmi
odmítni

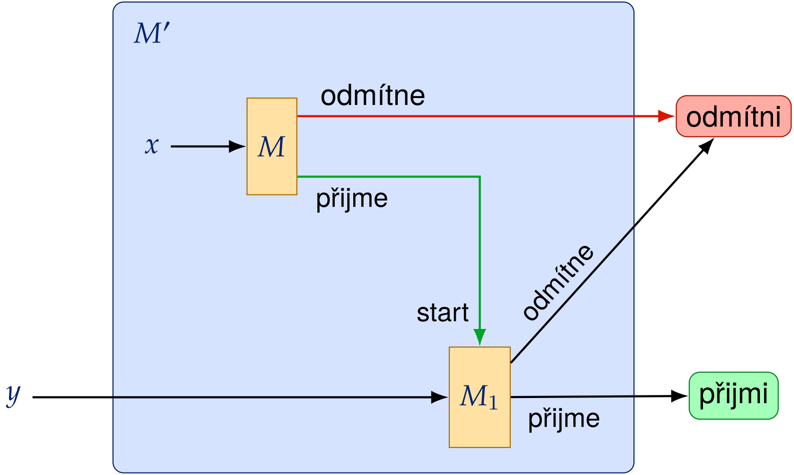
Riceova věta (důkaz)

K dvojici M a x definujeme Turingův stroj M'



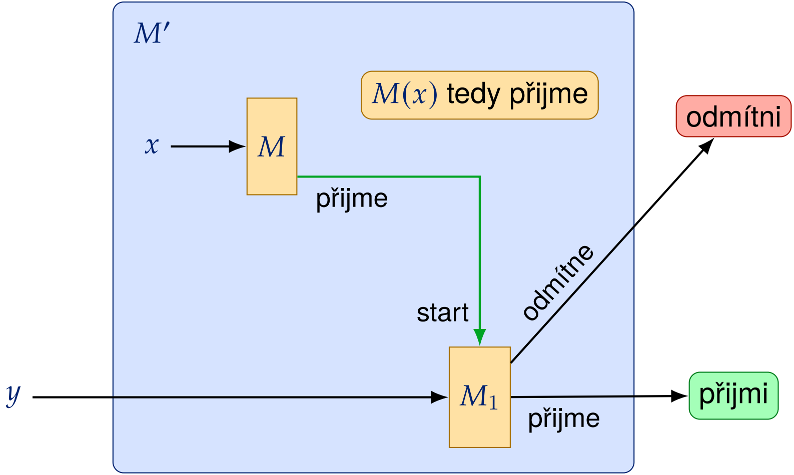
Riceova věta (důkaz, $x \in L(M)$)

Předpokládejme $x \in L(M)$



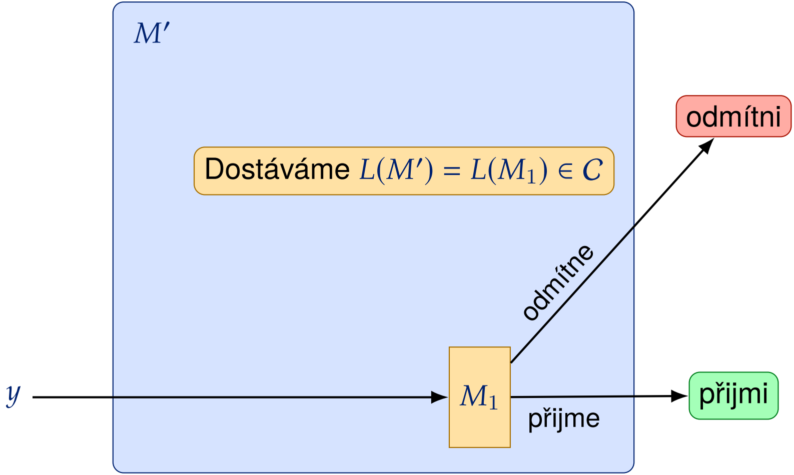
Riceova věta (důkaz, $x \in L(M)$)

Předpokládejme $x \in L(M)$



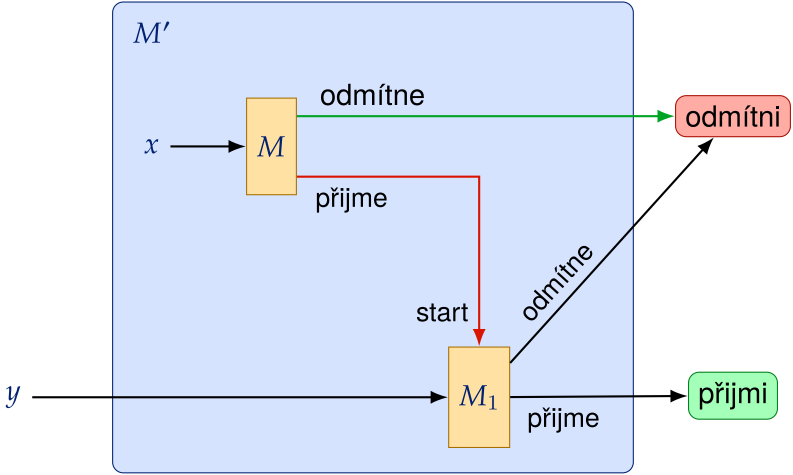
Riceova věta (důkaz, $x \in L(M)$)

Předpokládejme $x \in L(M)$



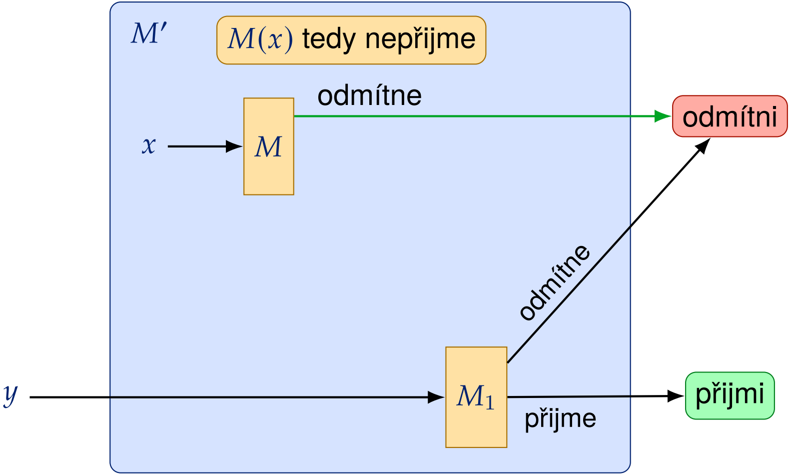
Riceova věta (důkaz, $x \notin L(M)$)

Předpokládejme $x \notin L(M)$



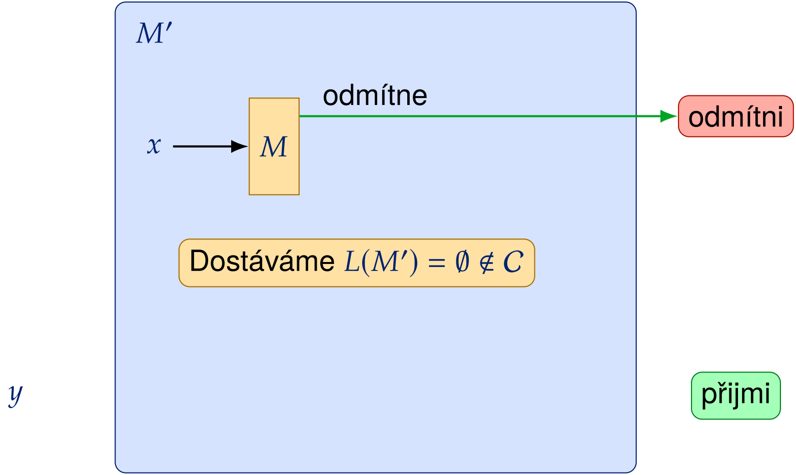
Riceova věta (důkaz, $x \notin L(M)$)

Předpokládejme $x \notin L(M)$



Riceova věta (důkaz, $x \notin L(M)$)

Předpokládejme $x \notin L(M)$



Riceova věta (důkaz, dokončení)

- $x \in L(M) \implies L(M') = L(M_1) \in \mathcal{C} \implies \langle M' \rangle \in L_{\mathcal{C}}$
- $x \notin L(M) \implies L(M') = \emptyset \notin \mathcal{C} \implies \langle M' \rangle \notin L_{\mathcal{C}}$
- Funkce $f(\langle M, x \rangle) = \langle M' \rangle$ je totální algoritmicky vyčíslitelná a platí

$$\langle M, x \rangle \in L_u \iff x \in L(M) \iff f(\langle M, x \rangle) = \langle M' \rangle \in L_{\mathcal{C}}$$

- Dostáváme $L_u \leq_m L_{\mathcal{C}}$
- Pokud $\emptyset \in \mathcal{C}$
  - stačí vyměnit role $\mathcal{C}$ a doplňku $\mathcal{C}$
  - tím ukážeme $L_u \leq_m \overline{L_{\mathcal{C}}}$

Riceova věta (funkce)

Pomocí f_M označíme funkci, kterou počítá Turingův stroj M.

Věta (Riceova věta (funkce))

Nechť C je třída algoritmicky vyčíslitelných funkcí a položme

$$A_C = \{\langle M \rangle \mid f_M \in C\}$$

Jazyk A_C rozhodnutelný, právě když je třída C triviální, tedy buď je prázdná nebo obsahuje všechny algoritmicky vyčíslitelné funkce.

- Důkaz analogický verzi pro jazyky

Riceova věta (důsledky)

Následující jazyky jsou nerozhodnutelné dle Riceovy věty

- $\mathrm{Tot} = \{\langle M \rangle \mid f_M$ je totální, tedy $\mathrm{dom}\, f_M = \Sigma^*\}$
- $\mathrm{Sum} = \{\langle M \rangle \mid M$ počítá součet dvou čísel$\}$
- $\mathrm{Inc} = \{\langle M \rangle \mid f_M$ je rostoucí$\}$
- $\mathrm{Zero} = \{\langle M \rangle \mid f_M(0)\downarrow = 0\}$
- $\mathrm{ToUpper} = \{\langle M \rangle \mid M$ změní ve vstupu malá písmena na velká$\}$
- $\mathrm{HW} = \{\langle M \rangle \mid M$ vypíše `Hello, world` jako prvních 12 znaků výstupu$\}$

Postův korespondenční problém

Postův korespondenční problém

Instance: Množina „dominových kostek“ P:

$$P = \left\{ \left[\frac{t_1}{b_1} \right], \left[\frac{t_2}{b_2} \right], \ldots, \left[\frac{t_k}{b_k} \right] \right\}$$

kde $t_1, \ldots, t_k, b_1, \ldots, b_k \in \Sigma^*$ jsou řetězce.

Otázka: Existuje párovací posloupnost $i_1, i_2, \ldots, i_l$, kde $l \geq 1$ a $t_{i_1} t_{i_2} \ldots t_{i_l} = b_{i_1} b_{i_2} \ldots b_{i_l}$?

Věta (Bez důkazu)

Postův korespondenční problém je nerozhodnutelný.

Základní třídy složitosti

Rozhodovací problémy

V rozhodovacím problému se ptáme, zda daná instance x splňuje danou podmínku.

- Odpověď je typu `ano`/`ne`.
- Rozhodovací problém formalizujeme jako jazyk $L \in \Sigma^*$ kladných instancí a otázku, zda $x \in L$.
- Příklady rozhodovacích problémů:
  - Je daný graf souvislý?
  - Má daná logická formule model?
  - Má daný lineární program přípustné řešení.
  - Je dané číslo prvočíslem?

Úlohy

V úloze pro danou instanci x hledáme y, které splňuje určitou podmínku

- Odpovědí je zde y nebo informace o tom, že žádné vhodné y neexistuje
- Úlohu formalizujeme jako relaci $R \subseteq \Sigma^* \times \Sigma^*$
  - K dané instanci x hledáme y tak, že $(x, y) \in R$
- Příklady úloh:
  - Nalezení silně souvislých komponent orientovaného grafu
  - Nalezení splňujícího ohodnocení logické formule
  - Nalezení přípustného řešení lineárního programu

Časová a prostorová složitost Turingova stroje

Definice

Nechť M je (deterministický) Turingův stroj a nechť $f : \mathbb{N} \rightarrow \mathbb{N}$ je funkce, která je definovaná pro každý vstup.

- M pracuje v čase $f(n)$, pokud výpočet M nad libovolným vstupem x délky $|x| = n$ skončí po provedení nejvýše $f(n)$ kroků.
- M pracuje v prostoru $f(n)$, pokud výpočet M nad libovolným vstupem x délky $|x| = n$ skončí a využije nejvýš $f(n)$ buněk pracovní pásky.

Základní deterministické třídy složitosti

Definice

Nechť $f : \mathbb{N} \to \mathbb{N}$ je funkce, potom definujeme třídy:

$\mathrm{TIME}(f(n))$ třída jazyků přijímaných Turingovými stroji, které pracují v čase $O(f(n))$

$\mathrm{SPACE}(f(n))$ třída jazyků přijímaných Turingovými stroji, které pracují v prostoru $O(f(n))$

$\mathrm{TIME}(f(n)) \subseteq \mathrm{SPACE}(f(n))$ pro každou funkci $f : \mathbb{N} \to \mathbb{N}$.

- V jednom kroku pohne Turingův stroj hlavou jen o jednu buňku vpravo nebo vlevo
- Stroj použije v každém kroku nejvýš jednu buňku navíc

Význačné deterministické třídy složitosti

- Třída problémů řešitelných v polynomiálním čase

$$P = \bigcup_{k \in \mathbb{N}} \mathrm{TIME}(n^k)$$

- Třída problémů řešitelných v polynomiálním prostoru

$$\mathrm{PSPACE} = \bigcup_{k \in \mathbb{N}} \mathrm{SPACE}(n^k).$$

- Třída problémů řešitelných v exponenciálním čase

$$\mathrm{EXPTIME} = \bigcup_{k \in \mathbb{N}} \mathrm{TIME}(2^{n^k}).$$

Proč polynomy?

Silnější verze Churchovy-Turingovy teze

Reálné výpočetní modely lze simulovat na Turingovu stroji s polynomiálním zpomalením/nárůstem prostoru.

- Polynomy jsou uzavřeny na skládání.
- Polynomy (obvykle) nerostou příliš rychle.
- Definice třídy P nezávisí na zvoleném výpočetním modelu.

Cobhamova-Edmondsova teze, 1965

P odpovídá třídě prakticky řešitelných problémů na počítači.