Základy složitosti a vyčíslitelnosti

NTIN090

Petr Kučera

2022/23 (3. přednáška)

Kódování objektů

Kódování objektů (značení)

- Konečné objekty (např. číslo, řetězec, Turingův stroj, RAM, graf nebo formuli) můžeme kódovat binárními řetězci
- Podobně můžeme zakódovat i n-tice objektů

Definice

$\langle X \rangle$ binární řetězec kódující objekt X

$\langle X_1, \ldots, X_n \rangle$ binární řetězec kódující n-tici objektů $X_1, \ldots, X_n$

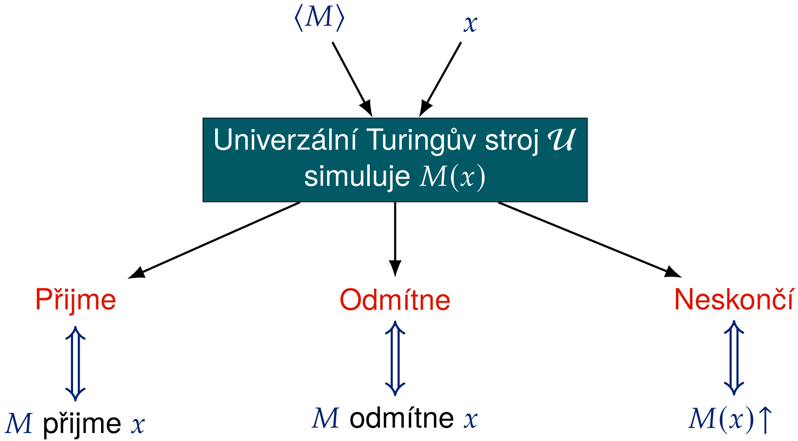
Příklad

$\langle M \rangle$ kód Turingova stroje M

$\langle M, x \rangle$ kód dvojice tvořené Turingovým strojem M a řetězcem x

Univerzální Turingův stroj

Univerzální Turingův stroj



Univerzální Turingův stroj

Vstup	$\langle M, x\rangle$ (M je Turingův stroj, x je vstup)
Univerzální Turingův stroj	simuluje práci stroje M nad vstupem x
Výsledek práce	zastavení/přijetí/zamítnutí vstupu a obsah výstupní pásky je dán výsledkem $M(x)$
Univerzální jazyk	jazyk univerzálního Turingova stroje

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

Univerzální jazyk formalizuje problém PŘIJETÍ VSTUPU

PŘIJETÍ VSTUPU

Instance: Kód Turingova stroje M a vstupní řetězec x

Otázka: Přijme M vstup x?

Struktura $\mathcal{U}$

Popíšeme 3-páskový Univerzální Turingův stroj $\mathcal{U}$

1. páska obsahuje vstup $\langle M, x \rangle$

$\langle M \rangle$	x

Na 2. pásce je uložen obsah pracovní pásky M
Symbol X_j zapsán jako $(j)_B$, bloky mají touž délku b bitů

…	|	0	1	0	|	0	1	0	|	1	0	1	|	1	1	0	|	…

3. páska obsahuje $(i)_B$ reprezentující aktuální stav q_i stroje M

…	1	0	0	1	1	λ	λ	λ	λ	λ	λ	λ	λ	λ	λ	λ	λ	…

Výpočet $\mathcal{U}$

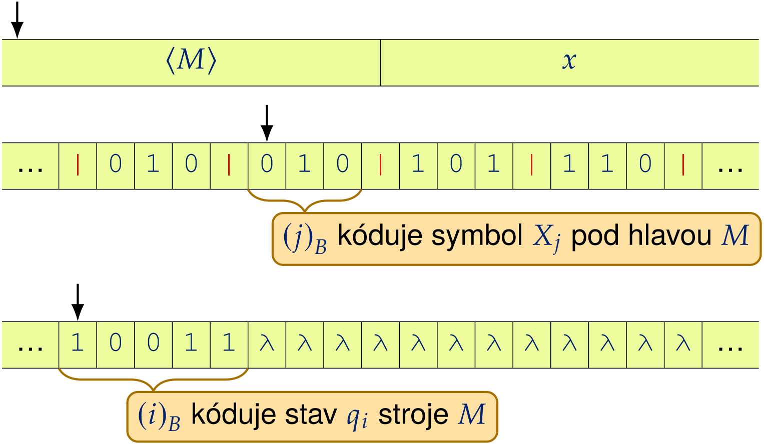
- Vstup $\mathcal{U}$ má dvě části $\langle M \rangle$ a x
  - $\mathcal{U}$ umí číst každou zvlášť
- Simulovaný TS $M = (Q, \Sigma, \delta, q_0, \{q_1\})$
  - Jediný přijímající stav q_1
  - Vstupní abeceda $\{\mathtt{0}, \mathtt{1}\}$
  - Pásková abeceda Σ není omezená
- $\langle M \rangle$ kóduje přechodovou funkci δ
- Výpočet $\mathcal{U}(\langle M \rangle, x)$ má 3 fáze
  1. Inicializace
  2. Simulace
  3. Zakončení

Inicializace

1. Syntaktická kontrola
   - Pokud první část vstupu není syntakticky správným kódem Turingova stroje, odmítni
2. Určení délky bloku b pro znak na 2. pásce
   - Maximální délka znaku X_i v rámci nějaké instrukce
   - Abeceda Σ obsahuje alespoň $\mathtt{0}$, $\mathtt{1}$ a λ, tedy $b \geq 2$
   - Pracovní abeceda není jinak omezená
3. Přepis vstupu na 2. pásku
   - Překódování vstupu do bloků délky b oddělených |
   - $\mathtt{0}$ je přepsáno na $\mathtt{0}^b$ ($X_0 = \mathtt{0}$)
   - $\mathtt{1}$ je přepsáno na $\mathtt{0}^{b-1}\mathtt{1}$ ($X_1 = \mathtt{1}$)
4. Zapiš $\mathtt{0}$ na 3. pásku
   - Počáteční stav je q_0
5. Návrat všech tří hlav na začátky slov na příslušných páskách

Polohy hlav na začátku simulace kroku M

1. páska na začátku kódu $\langle M \rangle$
2. páska nad blokem symbolu X_j, nad nímž je hlava M
3. páska na začátku čísla stavu q_i



Simulace kroku M

1. Hledej v $\langle M \rangle$ instrukci pro displej (q_i, X_j)
   - Instrukce není nalezena $\Longrightarrow$ simulace končí
   - Jinak označme nalezenou instrukci $\delta(q_i, X_j) = (q_k, X_l, Z)$
2. Na 3. pásce přepiš číslo stavu na $(k)_B$
3. Na 2. pásce přepiš blok pod hlavou na $(l)_B$ (b bitů)
4. Na 2. pásce přesuň hlavu
   - o blok vlevo (je-li $Z = L$)
   - o blok vpravo (je-li $Z = R$)
   - na začátek stávajícího bloku (je-li $Z = N$)
5. Pokud se hlava přesunem dostala mimo použitou část pásky, $\mathcal{U}$ přidá další blok tvaru $\mathtt{0}^{b-2}\mathtt{10}$ ($X_2 = \lambda$)
6. Vrať hlavy do předpokládaných pozic a pokračuj simulací dalšího kroku M

Zakončení

- $\mathcal{U}$ přijme, pokud na 3. pásce je číslo 1 jediného přijímajícího stavu q_1, jinak odmítne
- Pokud chceme simulovat výpočet funkce M, pak je potřeba přepsat pracovní pásku do řetězce z Σ^*

Nerozhodnutelnost Univerzálního jazyka

Vlastnosti univerzálního jazyka

Věta

Jazyk $L_u = \{\langle M, x\rangle \mid x \in L(M)\}$ *je částečně rozhodnutelný, ale není rozhodnutelný.*

- Částečná rozhodnutelnost plyne z existence univerzálního Turingova stroje
- Nerozhodnutelnost ukážeme diagonalizací, plán:
  1. Univerzální jazyk reprezentujeme jako matici A
  2. Jazyk daný doplňkem diagonály A není částečně rozhodnutelný
  3. Z toho dovodíme, že L_u není rozhodnutelný

Univerzální jazyk jako matice

$L_u = \{\langle M, x\rangle \mid x \in L(M)\}$ lze reprezentovat nekonečnou maticí A

$\mathbf{w}_i$ označuje binární řetězec s indexem i

M_i označuje TS s Gödelovým číslem i

Platí: $\mathbf{w}_i = \langle M_i \rangle$

Odpovídá j-tému binárnímu řetězci $\mathbf{w}_j$ s indexem j

indexy binárních řetězců

Gödelova čísla

A	0	1	2	…	i	…	j	…
0	0	1	0	…	0	…	1	…
1	1	1	0	…	1	…	0	…
2	1	1	0	…	0	…	1	…
⋮								
i	0	0	1	…	1	…	0	…
⋮								

$\mathbf{w}_i \in L(M_i)$

$\mathbf{w}_j \notin L(M_i)$

Odpovídá Turingovu stroji M_i s Gödelovým číslem i

Matice univerzálního jazyka

- Každý Turingův stroj M má nekonečně mnoho Gödelových čísel

$\Longrightarrow$ Každému Turingovu stroji M odpovídá nekonečně mnoho řádků v matici A

$\Longrightarrow$ Každému částečně rozhodnutelnému jazyku odpovídá nekonečně mnoho řádků v matici A

Doplněk diagonály matice A určuje diagonální jazyk

$$\mathrm{DIAG} = \{\langle M\rangle \mid \langle M\rangle \notin L(M)\}$$

- DIAG nemá svůj řádek v matici A
- DIAG není částečně rozhodnutelný

Diagonální jazyk

$\mathrm{DIAG} = \{\langle M\rangle \mid \langle M\rangle \notin L(M)\}$

$\mathrm{DIAG} = \{\mathbf{w}_i \mid \mathbf{w}_i \notin L(M_i)\}$

Odpovídá j-tému binárnímu řetězci $\mathbf{w}_j$ s indexem j

indexy binárních řetězců

Gödelova čísla

$\mathbf{A}$	0	1	2	⋯	i	⋯	j	⋯
0	0	1	0	⋯	0	⋯	1	⋯
1	1	1	0	⋯	1	⋯	0	⋯
2	1	1	0	⋯	0	⋯	1	⋯
⋮								
i	0	0	1	⋯	1	⋯	0	⋯
⋮								
DIAG	1	0	1		0		1	

$\mathbf{w}_i \in L(M_i)$

Odpovídá Turingovu stroji M_i s Gödelovým číslem i

DIAG se liší od každého řádku matice A v diagonálním políčku

$\mathbf{w}_i \notin L(M_i)$

Diagonální jazyk není částečně rozhodnutelný

Věta

Jazyk $\mathrm{DIAG} = \{\langle M\rangle \mid \langle M\rangle \notin L(M)\}$ *není částečně rozhodnutelný*

Důkaz.

Sporem: existuje TS M_D, který přijímá DIAG (tj. $\mathrm{DIAG} = L(M_D)$)

$$
\begin{tikzcd}
 & \langle M_D\rangle \notin L(M_D) \arrow[dd, Leftrightarrow, "\times" description, "\text{Spor}"] \\
\langle M_D\rangle \in \mathrm{DIAG} \arrow[ur, Leftrightarrow, "\text{definice } \mathrm{DIAG}"] \arrow[dr, Leftrightarrow, "\mathrm{DIAG} = L(M_D)"'] & \\
 & \langle M_D\rangle \in L(M_D)
\end{tikzcd}
$$

□

Nerozhodnutelnost univerzálního jazyka

Věta

Jazyk $L_u = \{\langle M, x\rangle \mid x \in L(M)\}$ *není rozhodnutelný.*

Důkaz.

- Sporem: Existuje Turingův stroj M_u, který rozhoduje L_u
  - $L_u = L(M_u)$ a $M_u(\langle M, x\rangle)\downarrow$ pro každý vstup $\langle M, x\rangle$
- Pro každý Turingův stroj M platí

$$\langle M\rangle \in \mathrm{DIAG} \overbrace{\iff}^{\text{definice } \mathrm{DIAG}} \langle M\rangle \notin L(M) \overbrace{\iff}^{\text{definice } L_u} \langle M, \langle M\rangle\rangle \notin L_u$$

- Stroj M_u lze použít k rozhodování DIAG
- Spor s nerozhodnutelností DIAG

$\square$

Vlastnosti (částečně) rozhodnutelných jazyků

Uzavřenost na jazykové operace

Doplněk jazyka L označíme pomocí $\overline{L} = \Sigma^* \setminus L$.

Konkatenací dvou jazyků L_1 a L_2 vznikne jazyk

$$L_1 \cdot L_2 = \{w_1 w_2 \mid w_1 \in L_1, w_2 \in L_2\}.$$

Kleeneho uzávěrem jazyka L je jazyk

$$L^* = \{w \mid (\exists k \in \mathbb{N})(\exists w_1, \ldots, w_k \in L)[w = w_1 w_2 \ldots w_k]\}.$$

Věta

Jsou-li L_1 a L_2 (částečně) rozhodnutelné jazyky, pak $L_1 \cup L_2$, $L_1 \cap L_2$, $L_1 \cdot L_2$, L_1^* jsou (částečně) rozhodnutelné jazyky.

Jsou (částečně) rozhodnutelné jazyky uzavřené na doplněk?

Postova věta

Věta (Postova věta)

Jazyk L je rozhodnutelný, právě když L i $\overline{L}$ jsou částečně rozhodnutelné jazyky.

Důkaz.

Dva kroky

„$\Longrightarrow$“ L je rozhodnutelný $\Longrightarrow$ L i $\overline{L}$ jsou částečně rozhodnutelné

„$\Longleftarrow$“ L i $\overline{L}$ jsou částečně rozhodnutelné $\Longrightarrow$ L je rozhodnutelný

$\square$

Postova věta (důkaz „$\Longrightarrow$“)

- Předpokládáme, že $L \subseteq \Sigma^*$ je rozhodnutelný jazyk

$\Longrightarrow$ Existuje Turingův stroj M rozhodující L
  - $L = L(M)$ a $M(x)\downarrow$ pro každý vstup $x \in \Sigma^*$
- Sestavíme Turingův stroj M', který se vstupem x
  1. Pustí $M(x)$
  2. Na závěr zneguje odpověď
     - $M'(x)$ přijme $\Longleftrightarrow$ $M(x)$ odmítne
- M' přijímá $\overline{L}$
- $M'(x)\downarrow$ pro každý vstup $x \in \Sigma^*$

$\Longrightarrow$ $\overline{L}$ je rozhodnutelný jazyk

$\Longrightarrow$ L i $\overline{L}$ jsou částečně rozhodnutelné jazyky

Postova věta (důkaz „ $\Longleftarrow$ “)

- Předpokládáme, že
  - $L = L(M_1)$ pro nějaký Turingův stroj $M_1 = (Q_1, \Sigma, \delta_1, q_0^1, F_1)$
  - $\overline{L} = L(M_2)$ pro nějaký Turingův stroj $M_2 = (Q_2, \Sigma, \delta_2, q_0^2, F_2)$
- Sestavíme Turingův stroj M, který rozhoduje L, tedy
  - $L = L(M)$ a
  - $M(x)\downarrow$ pro každý vstup x
- Idea:
  - Pokud $M_1(x)$ přijme, pak $x \in L$
  - Pokud $M_2(x)$ přijme, pak $x \notin L$

Postova věta (důkaz „ $\Longleftarrow$ “)

Práce M se vstupem x

```
Pusť M₁(x) a M₂(x) paralelně a čekej až jeden z nich přijme
if M₁(x) přijal then
    přijmi
if M₂(x) přijal then
    odmítni
```

Možná implementace M

Přechodová funkce M složená z δ_1 a δ_2

δ $q = (q_1, q_2)$

Stav M reprezentuje stav q_1 stroje M_1 a stav q_2 stroje M_2

Pozice hlavy stroje M_1

Páska stroje M_1

Pozice hlavy stroje M_2

Páska stroje M_2

Uzavřenost na doplněk

Důsledek

- *Třída rozhodnutelných jazyků je uzavřená na operaci doplňku*
- *Třída částečně rozhodnutelných jazyků není uzavřená na operaci doplňku*

- Jazyk L_u je částečně rozhodnutelný, ale není rozhodnutelný
- $\overline{L_u}$ není částečně rozhodnutelný dle Postovy věty
- $\mathrm{DIAG} = \{\langle M\rangle \mid \langle M\rangle \notin L(M)\}$ není částečně rozhodnutelný
- $\overline{\mathrm{DIAG}} = \{\langle M\rangle \mid \langle M\rangle \in L(M)\}$ je částečně rozhodnutelný
    - Plyne z existence univerzálního Turingova stroje

Vztahy tříd jazyků

PD částečně rozhodnutelné jazyky

- *partially decidable*

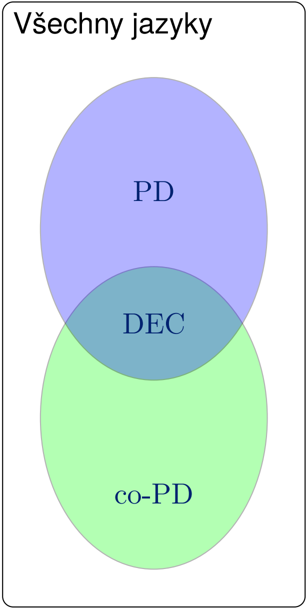
co-PD doplňky částečně rozhodnutelných jazyků

- $L \in \text{co-PD} \Leftrightarrow \overline{L} \in \mathrm{PD}$
- *co-partially decidable*

DEC rozhodnutelné jazyky

- *decidable*

Postova věta: $\mathrm{DEC} = \mathrm{PD} \cap \text{co-PD}$



Algoritmicky vyčíslitelné funkce

Funkce — značení

Pro částečnou funkci $f : \Sigma^* \to \Sigma^*$ definujeme:

Doména f je množina vstupů, pro něž je hodnota f definovaná

$$\operatorname{dom} f = \{x \in \Sigma^* \mid f(x)\downarrow\}$$

Totální funkce f je definovaná pro každý vstup x, tedy $\operatorname{dom} f = \Sigma^*$

Obor hodnot f je množina možných hodnot f

$$\operatorname{rng} f = \{y \in \Sigma^* \mid (\exists x \in \Sigma^*)[f(x)\downarrow = y]\}$$

Značení používáme i pro jiné než řetězcové funkce

- například funkce $f : \mathbb{N} \to \mathbb{N}$

Algoritmicky vyčíslitelné funkce (definice)

Intuitivně

Algoritmicky vyčíslitelná funkce jsou právě ty, jejichž hodnoty lze vyčíslit nějakým algoritmem

Definice

Částečná funkce $f : \Sigma^* \to \Sigma^*$ je algoritmicky vyčíslitelná pokud existuje Turingův stroj M, který ji počítá.

Pro každý vstup $x \in \Sigma^*$ platí

- Je-li $f(x)\uparrow$, pak $M(x)\uparrow$
- Je-li $f(x)\downarrow = y$, pak
  - $M(x)\downarrow$ a
  - na výstupní pásce M je po ukončení výpočtu $M(x)$ řetězec y

Algoritmicky vyčíslitelné funkce

- Vyčíslitelné funkce = částečně rekurzivní funkce
- Totální vyčíslitelné funkce = obecně rekurzivní funkce
- Uvažujeme i funkce jiných typů, například
  - aritmetické funkce
  - funkce více parametrů

Příklad

Například funkce

$$f(x,y) = x^2 + y^2$$

může být realizována řetězcovou funkcí

$$f'(\langle x,y\rangle) = \langle x^2 + y^2\rangle$$

Ne všechny funkce jsou vyčíslitelné

- Vyčíslitelných funkcí je jen spočetně mnoho
  $\Longrightarrow$ ne všechny funkce jsou vyčíslitelné

Příklad

Charakteristická funkce jazyka L_u

$$\chi_u(\langle M, x\rangle) = \begin{cases} 1 & x \in L(M) \\ 0 & x \notin L(M) \end{cases}$$

není algoritmicky vyčíslitelná, protože jazyk

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

je algoritmicky nerozhodnutelný

Vlastnosti (částečně) rozhodnutelných jazyků

Charakteristická funkce rozhodnutelného jazyka

Věta

Jazyk $L \subseteq \Sigma^*$ je rozhodnutelný, právě když jeho charakteristická funkce

$$\chi_L(x) = \begin{cases} 1 & x \in L \\ 0 & x \notin L \end{cases}$$

je algoritmicky vyčíslitelná.

Důkaz.

Důkaz ve dvou krocích

„$\implies$“ L je rozhodnutelný $\implies$ χ_L je algoritmicky vyčíslitelná

„$\impliedby$“ χ_L je algoritmicky vyčíslitelná $\implies$ L je rozhodnutelný

$\square$

Důkaz „$\Longrightarrow$“

- Předpokládáme, že L je rozhodnutelný jazyk
- Existuje Turingův stroj M, který
  - přijímá L ($L = L(M)$)
  - $M(x)\downarrow$ pro každý vstup $x \in \Sigma^*$
- Popíšeme Turingův stroj M', který počítá χ_L

Výpočet M' se vstupem x

```
Simuluj M(x)
if M přijal then
    Zapiš na výstup 1
else
    Zapiš na výstup 0
```

Důkaz „ $\Longleftarrow$ “

- Předpokládáme, že funkce χ_L je algoritmicky vyčíslitelná
- Existuje Turingův stroj M, který počítá χ_L
- $M(x)\downarrow$ pro každý vstup $x \in \Sigma^*$
  - protože $\chi_L(x)\downarrow$ pro každý vstup $x \in \Sigma^*$
- $M(x)$ vypíše na výstup hodnotu $\chi_L(x)$ (1 pokud $x \in L$, jinak 0)
- Popíšeme Turingův stroj $M'(x)$, který
  - přijímá L ($L = L(M')$) a
  - $M'(x)\downarrow$ pro každý vstup $x \in \Sigma^*$

Výpočet M' se vstupem x

```
Simuluj M(x)
if M vypsal 1 then
    přijmi
else
    odmítni
```

Přijetí nebo zastavení

Věta

Jazyk L je částečně rozhodnutelný, právě když existuje Turingův stroj M splňující

$$L = \{x \in \Sigma^* \mid M(x)\downarrow\} \tag{1}$$

Důkaz.

Ve dvou krocích

„$\Longrightarrow$“ L je částečně rozhodnutelný $\Longrightarrow$ existuje M splňující (1)

„$\Longleftarrow$“ Existuje M splňující (1) $\Longrightarrow$ L je částečně rozhodnutelný

$\square$

Důkaz „ $\Longrightarrow$ “

- Předpokládáme, že L je částečně rozhodnutelný
- Existuje Turingův stroj M', který přijímá L ($L = L(M')$)
- Popíšeme Turingův stroj M, který splňuje

$$L = \{x \in \Sigma^* \mid M(x)\downarrow\}$$

Výpočet M se vstupem x

1 Simuluj $M'(x)$
2 **if** $M'(x)$ odmítl **then**
3   vstup do nekonečného cyklu

- Pro každý řetězec $x \in \Sigma^*$

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

Důkaz „ $\Longleftarrow$ “

- Předpokládejme, že M je Turingův stroj splňující

$$L = \{x \in \Sigma^* \mid M(x)\downarrow\}$$

- Popíšeme Turingův stroj M', který přijímá L ($L = L(M')$)

Výpočet M' se vstupem x
1 Simuluj $M(x)$
2 Přijmi

- Platí

$$x \in L \Longleftrightarrow M(x)\downarrow \Longleftrightarrow x \in L(M')$$

- Tedy $L = L(M')$

Domény algoritmicky vyčíslitelných funkcí

Věta

Jazyk L je částečně rozhodnutelný, právě když existuje algoritmicky vyčíslitelná funkce f splňující

$$L = \operatorname{dom} f = \{x \in \Sigma^* \mid f(x)\downarrow\} \tag{2}$$

Důkaz.

- L je částečně rozhodnutelný, právě když existuje TS M splňující

$$L = \{x \in \Sigma^* \mid M(x)\downarrow\} \tag{3}$$

(2) $\implies$ (3) M počítá funkci f

(3) $\implies$ (2) f je funkce počítaná strojem M

$\square$

Existenční kvantifikace

Věta

Jazyk L je částečně rozhodnutelný, právě když existuje rozhodnutelný jazyk B splňující

$$L = \{x \in \Sigma^* \mid (\exists y \in \Sigma^*)[\langle x, y \rangle \in B]\} \tag{4}$$

Důkaz.

Důkaz ve dvou krocích

„$\Longrightarrow$“ L je částečně rozhodnutelný $\Longrightarrow$ existuje rozhodnutelný jazyk B splňující (4)

„$\Longleftarrow$“ existuje rozhodnutelný jazyk B splňující (4) $\Longrightarrow$ L je částečně rozhodnutelný

$\square$

Důkaz „ $\Longrightarrow$ “

- Předpokládáme, že L je částečně rozhodnutelný
- Existuje Turingův stroj M přijímající L ($L = L(M)$)
- Platí

$$L = \{x \mid (\exists n \in \mathbb{N})[\underbrace{M(x) \text{ přijme do } n \text{ kroků}}_{\substack{\text{Rozhodnutelná podmínka,} \\ \text{stačí simulovat } M(x) \text{ po } n \text{ kroků}}}]\}$$

- Stačí tedy definovat

$$B = \{\langle x, \langle n \rangle \rangle \mid M(x) \text{ přijme do } n \text{ kroků}\}$$

- Jazyk B je rozhodnutelný a splňuje

$$L = \{x \in \Sigma^* \mid \underbrace{(\exists y \in \Sigma^*)}_{y=\langle n \rangle}[\quad \underbrace{\langle x, y \rangle \in B}_{M(x) \text{ přijme do } n \text{ kroků}} \quad]\}$$

Důkaz „ $\Longleftarrow$ “

- Předpokládáme, že existuje rozhodnutelný jazyk B splňující

$$L = \{x \in \Sigma^* \mid (\exists y \in \Sigma^*)[\langle x, y\rangle \in B]\}$$

- Popíšeme Turingův stroj M přijímající L ($L = L(M)$)

Výpočet M se vstupem x

```
forall y ∈ Σ* v lexikografickém uspořádání do
    if ⟨x, y⟩ ∈ B then
        přijmi
```

- $x \in L \implies (\exists y \in \Sigma^*)[\langle x, y\rangle \in B] \implies M(x)$ přijme
- $x \notin L \implies (\forall y \in \Sigma^*)[\langle x, y\rangle \notin B] \implies M(x)\uparrow$
- Dohromady $L = L(M)$

Existenční kvantifikace (příklad)

$$\begin{aligned} L_u &= \{\langle M, x\rangle \mid x \in L(M)\} \\ &= \{\langle M, x\rangle \mid (\exists n \in \mathbb{N})[\underbrace{M(x) \text{ přijme do } n \text{ kroků}}_{\substack{\text{Rozhodnutelná podmínka,} \\ \text{stačí simulovat } M(x) \text{ po } n \text{ kroků}}}]\} \end{aligned}$$

- Následující jazyk je rozhodnutelný

$$B = \{\langle M, x, n\rangle \mid M(x) \text{ přijme do } n \text{ kroků}\}$$

- Částečně rozhodnutelný jazyk L_u můžeme zapsat jako

$$L_u = \{\langle M, x\rangle \mid (\exists n \in \mathbb{N})[\langle M, x, n\rangle \in B]\}$$

Uzavřenost na existenční kvantifikaci

Důsledek

Je-li B částečně rozhodnutelný jazyk, pak jazyk

$$A = \{x \in \Sigma^* \mid (\exists y \in \Sigma^*)[\langle x, y \rangle \in B]\}$$

je též částečně rozhodnutelný.

Důkaz.

- Existuje rozhodnutelný jazyk C splňující

$$B = \{\langle x, y \rangle \in \Sigma^* \mid (\exists z \in \Sigma^*)[\langle x, y, z \rangle \in C]\}$$

- Platí $A = \{x \in \Sigma^* \mid (\exists \langle y, z \rangle \in \Sigma^*)[\langle x, y, z \rangle \in C]\}$
- A je částečně rozhodnutelný dle předchozí věty

$\square$

Uzavřenost na existenční kvantifikaci (příklad)

$$
\begin{aligned}
\mathrm{NE} &= \{\langle M\rangle \mid L(M) \neq \emptyset\} \\
&= \{\langle M\rangle \mid (\exists x \in \Sigma^*)[x \in L(M)]\} \\
&= \{\langle M\rangle \mid (\exists x \in \Sigma^*)[\langle M, x\rangle \in L_u]\}
\end{aligned}
$$

- Jazyk L_u je částečně rozhodnutelný
- NE je tedy též částečně rozhodnutelný

Vyčíslitelnost jazyků

Enumerátor

Enumerátorem pro jazyk L je Turingův stroj E, který

- ignoruje svůj vstup,
- vypisuje řetězce $w \in L$ na vyhrazenou výstupní pásku
  - například oddělené $\#$
- každý řetězec $w \in L$ je někdy vypsán TS E
- Je-li L nekonečný, E svou činnost nikdy neskončí

Enumerátor pro jazyk NE

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

- Enumerátor pro jazyk NE řeší následující úlohu:
  - Vypiš kódy Turingových strojů, které přijímají alespoň jedno slovo

Enumerátor pro jazyk NE

```
forall ⟨M, x, n⟩ ∈ Σ* v shortlex uspořádání do
    Simuluj výpočet M(x) po nejvýš n kroků
    if M(x) přijal then
        Zapiš ⟨M⟩ na výstup
```

- Každý kód $\langle M\rangle \in \mathrm{NE}$ je vypsán nekonečný počet krát
- Stroje jsou vypisovány v neurčeném pořadí

Enumerátor pro jazyk NE

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

Upravíme enumerátor tak, aby každý kód stroje M s neprázdným jazykem byl vypsán právě jednou

Enumerátor jazyka NE

```
S ← prázdný seznam řetězců
forall ⟨M, x, n⟩ ∈ Σ* v shortlex uspořádání do
    Simuluj výpočet M(x) po nejvýš n kroků
    if M(x) přijal and ⟨M⟩ ∉ S then
        Zapiš ⟨M⟩ na výstup
        Přidej ⟨M⟩ do seznamu S
```

Vyčíslitelnost částečně rozhodnutelných jazyků

Věta

Jazyk L je částečně rozhodnutelný, právě když pro něj existuje enumerátor E.

Důkaz.

Důkaz ve dvou krocích

„ $\Longrightarrow$ “ L je částečně rozhodnutelný $\Longrightarrow$ existuje enumerátor E pro L

„ $\Longleftarrow$ “ Existuje enumerátor E pro L $\Longrightarrow$ L je částečně rozhodnutelný

$\square$

Důkaz „$\Longrightarrow$“

- L je částečně rozhodnutelný
- Existuje rozhodnutelný jazyk B splňující

$$L = \{x \in \Sigma^* \mid (\exists y \in \Sigma^*)[\langle x, y\rangle \in B]\}$$

Enumerátor E jazyka L

forall $\langle x, y\rangle \in \Sigma^*$ v shortlex uspořádání **do**
  if $\langle x, y\rangle \in B$ **then**
    Zapiš x na výstup

- Lze upravit tak, aby E vypsal každé slovo $x \in L$ právě jednou.
- Prvky L jsou vypisovány v neurčeném pořadí

Důkaz „ $\Longleftarrow$ “

- Máme enumerátor E pro jazyk L
- Popíšeme Turingův stroj M přijímající L ($L = L(M)$)

Výpočet M se vstupem x

```
Simuluj E a sleduj výstup
if E vypsal x then
    přijmi
```

$x \in L \implies E$ někdy vypíše x a $M(x)$ přijme

$x \notin L \implies E$ nikdy nevypíše x a $M(x)$ nepřijme (zacyklí se)

Dohromady $L = L(M)$

Enumerátor pro jazyk prvočísel

$$\mathrm{PRIME} = \{\langle p \rangle \mid p \text{ je prvočíslo}\}$$

Úloha: vypisuj prvočísla v rostoucím pořadí

Enumerátor prvočísel

```
forall p ∈ ℕ v rostoucím pořadí do
    if p je prvočíslo then
        Zapiš ⟨p⟩ na výstup
```

Lze zkonstruovat díky tomu, že jazyk PRIME je rozhodnutelný.

Vyčíslitelnost rozhodnutelných jazyků

Věta

Jazyk L je rozhodnutelný, právě když pro něj existuje enumerátor E, který navíc vypisuje prvky L v shortlex pořadí.

Důkaz.

Důkaz ve dvou krocích

- „$\Longrightarrow$“ L je rozhodnutelný $\Longrightarrow$ existuje enumerátor E pro L, který vypisuje prvky L v shortlex pořadí
- „$\Longleftarrow$“ Existuje enumerátor E pro L, který vypisuje prvky L v shortlex pořadí $\Longrightarrow$ L je rozhodnutelný

$\square$

Důkaz „ $\Longrightarrow$ “

- L je rozhodnutelný
- Popíšeme enumerátor E, který vypisuje slova L v shortlex pořadí

Enumerátor E jazyka L

```
// Podmínku lze ověřit díky
   rozhodnutelnosti L
forall x ∈ Σ* v shortlex uspořádání do
    if x ∈ L then
        Zapiš x na výstup
```

> V případě, že L je konečný jazyk, E se po vypsání posledního slova z L zacyklí.

Důkaz „ $\Longleftarrow$ “

- Máme enumerátor E pro jazyk L
- E vypisuje prvky L v rostoucím shortlex pořadí
- Rozlišíme dva případy
  1. L je konečný jazyk $\Longrightarrow$ L je rozhodnutelný
     - Všechny konečné jazyky jsou rozhodnutelné
  2. L je nekonečný jazyk $\Longrightarrow$ popíšeme stroj M, který rozhoduje L

```
Výpočet M se vstupem x
Simuluj E a sleduj výstup
if E vypsal x then
    přijmi
if E vypsal řetězec y > x then
    odmítni
```

L je nekonečný $\Longrightarrow$ vždy existuje $y > x$ $\Longrightarrow$ algoritmus skončí

Důsledek

Nekonečný jazyk L je částečně rozhodnutelný, právě když je oborem hodnot nějaké totální algoritmicky vyčíslitelné funkce f (tj. $L = \mathrm{rng}\, f$).

„$\Longrightarrow$“ L částečně rozhodnutelný

- máme enumerátor E pro L
- Pro jednoduchost uvažujeme parametry f typu $\mathbb{N}$
- pro $i \in \mathbb{N}$ definujeme

$$f(i) = (i+1)\text{-ní řetězec vypsaný } E$$

- E vypisuje právě řetězce z L
- Možné hodnoty f jsou právě řetězce z L

Vyčíslitelnost jazyků a funkce

Důsledek

Nekonečný jazyk L je částečně rozhodnutelný, právě když je oborem hodnot nějaké totální algoritmicky vyčíslitelné funkce f (tj. $L = \operatorname{rng} f$).

„$\Longleftarrow$“ Máme funkci f

- Popíšeme enumerátor E pro L

```
Výpočet E
forall y ∈ Σ* v shortlex pořadí do
    Zapiš f(y) na výstup
```

- $x \in L$

  $\Leftrightarrow$ existuje y pro nějž $f(y) = x$

  $\Leftrightarrow$ E vypíše x

Vyčíslitelnost rozhodnutelných jazyků a funkce

Definice

Funkce $f : \Sigma^* \longrightarrow \Sigma^*$ je rostoucí, pokud platí, že $u \prec v$ implikuje $f(u) \prec f(v)$ pro každé dva řetězce $u, v \in \Sigma^*$, kde $f(u)\downarrow$ a $f(v)\downarrow$.

Důsledek

Nekonečný jazyk L je rozhodnutelný, právě když je oborem hodnot nějaké rostoucí totální algoritmicky vyčíslitelné funkce f (tj. $L = \mathrm{rng}\, f$).

Vyčíslitelnost rozhodnutelných jazyků a funkce

Důsledek

Nekonečný jazyk L je rozhodnutelný, právě když je oborem hodnot nějaké rostoucí totální algoritmicky vyčíslitelné funkce f (tj. $L = \operatorname{rng} f$).

„$\Longrightarrow$“ L je rozhodnutelný

- Máme enumerátor E, který vypisuje prvky L v rostoucím shortlex pořadí
- Pro jednoduchost uvažujeme parametry f typu $\mathbb{N}$
- Pro $i \in \mathbb{N}$ definujeme

$$f(i) = (i+1)\text{-ní řetězec vypsaný } E$$

- E vypisuje právě řetězce z L
- Možné hodnoty f jsou právě řetězce z L
- f je rostoucí, protože E vypisuje prvky L v rostoucím shortlex pořadí

Vyčíslitelnost rozhodnutelných jazyků a funkce

Důsledek

Nekonečný jazyk L je rozhodnutelný, právě když je oborem hodnot nějaké rostoucí totální algoritmicky vyčíslitelné funkce f (tj. $L = \operatorname{rng} f$).

„$\Longleftarrow$" Máme funkci f

- Popíšeme enumerátor E pro L

Výpočet E

---

forall $y \in \Sigma^*$ v shortlex pořadí **do**
    Zapiš $f(y)$ na výstup

- E vypisuje právě prvky L v shortlex pořadí, protože f je rostoucí
- E tedy ukazuje, že L je rozhodnutelný