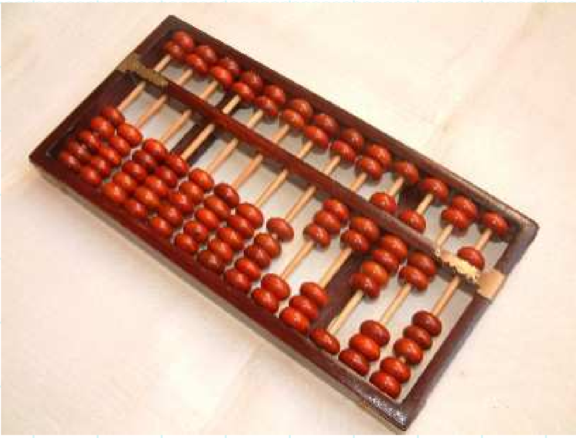
Teoretické základy informatiky

Abacus Machine

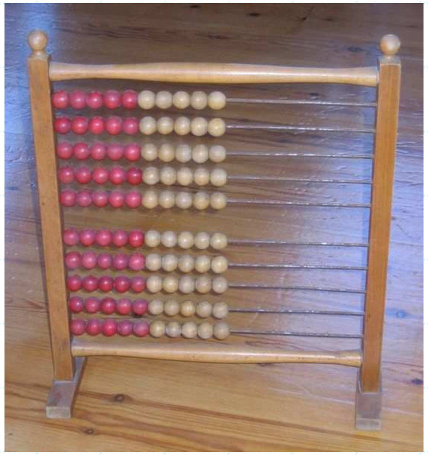
Mgr. Daniela Chudá, PhD., chuda@fiit.stuba.sk

Abacus počiatky: Babylon pred 5000 rokmi, Čína pre 3000 rokmi

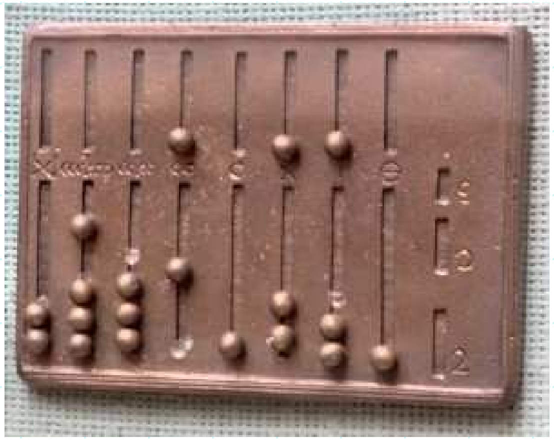
Abacus = počítadlo



A Chinese abacus



School abacus used in Danish elementary school. Early 20th century.



Reconstructed Roman Abacus

Zdroj: http://en.wikipedia.org/wiki/Abacus

Abacus Machine

- Abacus Machine = Počítadlový stroj
- Výpočtový model s operáciami inkrementovania a dekrementovania
- Počítadlový stroj nad prirodzenými číslami s operáciami:
  - +1
  - -1
  - cyklus

Definícia: Abacus Machine

- **Primitívy:**
  Symboly a_k, s_k ($k \in N$) sú jednoduché počítadlové stroje hĺbky 0
- **Rekurzívna definícia:**
  Nech $M_1, M_2, \ldots M_Z$ sú jednoduché počítadlové stroje s hĺbkami $h_1, h_1, \ldots h_Z$, postupnosť $M_1, M_2, \ldots M_Z$ sa nazýva počítadlový stroj s hĺbkou

  $$\overline{h} = \max \{h_1, h_1, \ldots h_Z\}$$
- **Kompozícia:**
  Nech M je počítadlový stroj s hĺbkou h. Jednoduchý počítadlový stroj s hĺbkou h+1 je postupnosť $(M)_k$, $k \in N$.

Neformálny opis počítadlového stroja registre

- pracuje s nekonečným počtom registrov R_i,$i \in N$, v danom čase s použitým konečným počtom registrov,
- v registroch môžu byť uložené ľubovoľne veľké prirodzené (nezáporné celé) čísla,
- predpokladá sa, že na začiatku výpočtu sú registre nastavené na 0,
- do zvolených registrov sa zapíšu vstupné hodnoty, počítadlový stroj vykoná operácie a po skončení výpočtu sú v zvolených registroch výstupné údaje .

Neformálny opis počítadlového stroja sémantika

- a_i pripočítavanie (addition) +1 $R_i \leftarrow R_i+1$
- s_i odpočítavanie (subtraction) ⊖1 $R_i \leftarrow R_i \ominus 1$

  len pre nezáporné čísla

$$x \ominus y = \begin{cases} x - y & \text{ak } x > y \\ 0 & \text{inak} \end{cases}$$

- $M_1M_2 \ldots M_n$ zreťazenie počítadlových strojov, vykonávanie sekvenčne za sebou
- $(M)_k$ cyklus počítadlových strojov, otestuje sa register R_k, v prípade, že je nenulový vykoná sa operácia zodpovedajúca stroju M. Stroj M sa cyklicky vykonáva až kým R_k nenadobudne nulovú hodnotu.

  while $(R_k>0)$ M

Prepis AM pomocou G

$G=(N,T,P,M)$

$N=\{M\}$

$T=\{a_k, s_k, (,)_k\}$

P: $M \rightarrow a_k \mid s_k$

$M \rightarrow MM$

$M \rightarrow (M)_k$

Príklad AM $f(i)=2+i$

Vstup R_1: i

Výstup R_1: $f(i)=2+i$

Výpočet na AM

$a_1 a_1$

Príklad AM $f(i,j)=i+j$

Vstup $R_1: i$

$R_2: j$

Výstup $R_1: f(i,j)=i+j$

Výpočet na AM (deštruktívne kopírovanie)

R_1:3	R_2:2
4	1
5	0

$(s_2 a_1)_2$

zovšeobecnenie: $(s_j a_i)_j$ $R_j \rightarrow R_i$

R_2 odpočítavam –1 až po 0 a zároveň pripočítavam +1 ku R_1

Príklad AM $f(i,j)=i+j$

Vstup R_1: i

R_2: j

Výstup R_3: $f(i,j)=i+j$

Výpočet na AM (deštruktívne kopírovanie)

$$(s_1 a_3)_1 (s_2 a_3)_2$$

Príklad AM $f(i,j)=i+j$

Vstup R_i: i

R_j: j

Výstup R_j: $f(i,j)=i+j$

Výpočet na AM (nedeštruktívne kopírovanie, zachováva len R_i)

$(s_i a_k a_j)_i \ (s_k a_i)_k$

Príklad AM $f(i,j)=i+j$

Vstup R_1: i

R_2: j

Výstup R_3: $f(i,j)=i+j$

Výpočet na AM (nedeštruktívne kopírovanie)

$$(s_1 a_0 a_3)_1 \, (s_0 a_1)_0 \, (s_2 a_0 a_3)_2 \, (s_0 a_2)_0$$

Príklady

$(s_i)_i$	vynulovanie registra R_i
$a_j(a_j)_j$	nekonečný cyklus
$(a_j)_j$	ak $R_j > 0$, nekonečný cyklus

Poznámky

Keď konštruujem počítadlové stroje, môžu byť syntakticky správne, ale nemusia byť sémanticky zmysluplné.

Môže byť veľmi ťažké zistiť, čo robí počítadlový stroj.

Nie je triviálne zistiť, či niekedy skončí počítadlový stroj, ktorý syntakticky správne zapíšem – algoritmicky neriešiteľné (problém zastavenia TM).

Veta o ekvivalencii medzi TM a AM

Funkcia f: $N^k \rightarrow N$ je T-vypočítateľná práve vtedy ak f je vypočítateľná na počítadlovom stroji (AM).

Ekvivalencia výpočtových modelov

Veta 6.4.1 *(O ekvivalencií výpočtových modeloch) Nasledujúce výpočtové modely sú ekvivalentné:*

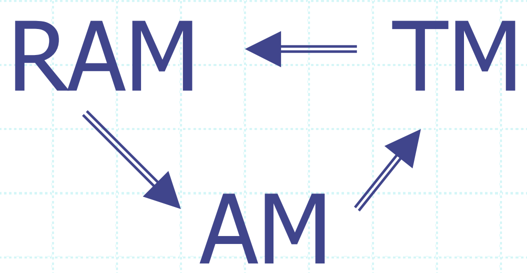
1. *Turingov stroj*
2. *Počítadlový stroj*
3. *Stroj RAM*

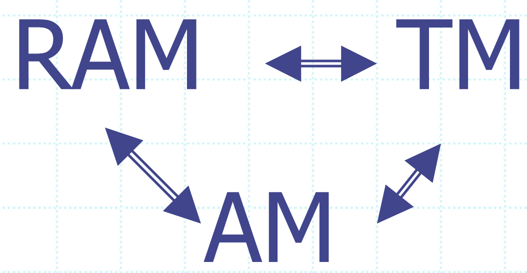
$$
\begin{tikzcd}
\mathrm{RAM} \arrow[dr, Rightarrow] & & \mathrm{TM} \arrow[ll, Rightarrow] \\
& \mathrm{AM} \arrow[ur, Rightarrow] &
\end{tikzcd}
$$

Ekvivalencia výpočtových modelov

Veta 6.4.1 *(O ekvivalencií výpočtových modeloch) Nasledujúce výpočtové modely sú ekvivalentné:*

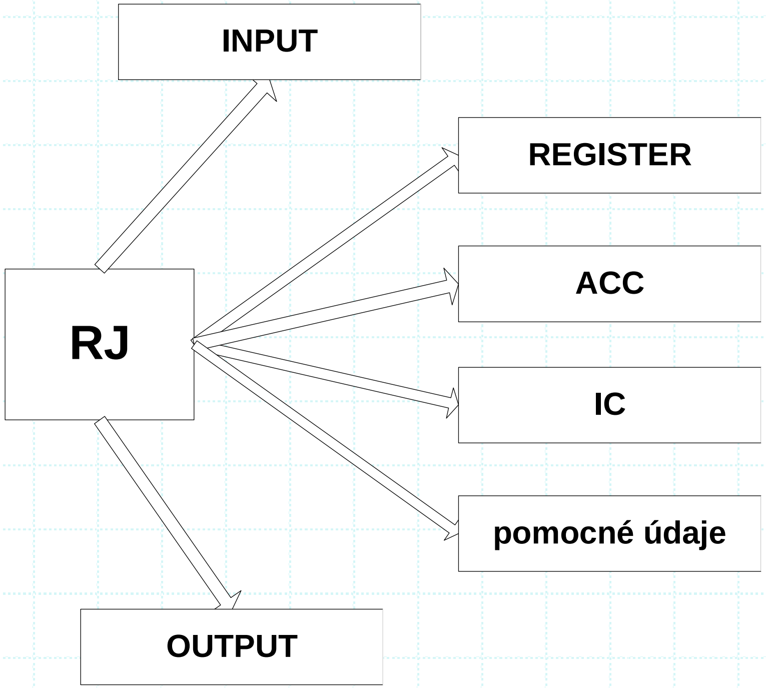
1. *Turingov stroj*
2. *Počítadlový stroj*
3. *Stroj RAM*





Ekvivalencia RAM <=> TS

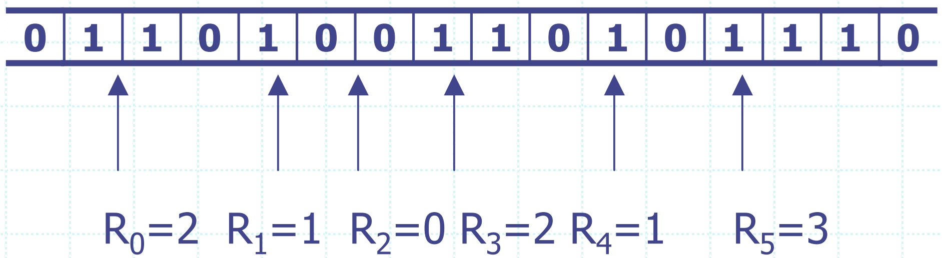
Simulácia RAM na TS – 6 páskový TS, simuluje sa každá inštrukcia zvlášť podľa polohy IC



AM ⇨ TM reprezentácia registrov

AM má nekonečne veľa registrov $R_0, R_1, \ldots$. tie potrebujem reprezentovať na TM → na páske v unárnej sústave.

dátová časť:



AM ⇨ TM definícia 4 jednoduchých TM

SEARCH (R_k)

na začiatku je čítacia hlava na pevnej nule, SEARCH (R_k) posunie hlavu ne register R_k – vyhľadanie hodnoty, ktorá sa nachádza v registri R_k

SHIFT (0R)

posun nuly o jedno políčko doprava, modifikuje pásku

SHIFT (0L)

posun nuly o jedno políčko doľava, modifikuje pásku

TEST (ZERO)

kontrolujem, či daný register (kde sa nachádza hlava) obsahuje 0 hodnotu (posunie sa vľavo)

AM ⇨ TM

definícia 4 jednoduchých TM

SHIFT (0R)

0	1	1	0	1	0	0	1	1	0	1	0	1	1	1	B

0	1	1	0	1	0	0	1	1	1	0	1	0	1	1	1

$\delta(q_x,1)=(q_x,1,R)$

$\delta(q_x,0)=(q_{x+1},1,R)$

$\delta(q_{x+1},1)=(q_x,0,R)$

$\delta(q_{x+1},0)=(q_{x+1},0,R)$

$\delta(q_{x+1},B)=(q_{x+2},0,L)$

$\delta(q_{x+2},n)=(q_{x+2},n,L)$ $\quad n\in\{0,1\}$

$\delta(q_{x+2},B)=(q_{xF},B,R)$

AM ⇨ TM definícia 4 jednoduchých TM

TEST (ZERO)

posuniem sa na najľavejší symbol na páske a buď skončím na q_{F0} alebo q_{F1} . Akceptujem: q_{F0} test bol nulový, q_{F1} test bol nenulový.

$\delta(q_y,0)=(q_{y0},0,L)$

$\delta(q_y,1)=(q_{y1},1,L)$

$\delta(q_{y0},u)=(q_{y0},u,L)$ $\quad u \in \{0,1\}$

$\delta(q_{y1},u)=(q_{y1},u,L)$

$\delta(q_{y0},B)=(q_{F0},B,R)$

$\delta(q_{y1},B)=(q_{F1},B,R)$

AM ⇨ TM

dôkaz indukciou a_k, s_k

a_k: SEARCH (R_k) → SHIFT (0R) +1

s_k: SEARCH (R_k) → SHIFT (0L) -1

AM $\Rightarrow$ TM dôkaz indukciou $M_1 M_2 M_3 \dots M_Z$

Mám $M_1, M_2, M_3 \dots, M_Z$ a z nich skonštruujem $M_1 M_2 M_3 \dots M_Z = M$ môžem ich vedľa seba naukladať.

indukčný predpoklad: nech $M_i \Rightarrow TM_i$

potom mám $TM_1, TM_2, TM_3 \dots, TM_Z$

? ako zostrojím TM, ktorý bude reprezentovať M? $M \Rightarrow TM$

zreťazím ich: $TM_1 \rightarrow TM_2 \rightarrow TM_3 \dots \rightarrow TM_Z$

AM ⇨ TM

dôkaz indukciou $(M)_k$

mám M a zostrojím $(M)_k$

indukčný predpoklad M ⇨ TM

? pýtam sa ako zostrojím TM, ktorý zodpovedá $(M)_k$?

END

q_{F0}

SEARCH (R_k) → TEST (ZERO)

q_{F1}

TM

Veta o ekvivalencii medzi RAM a AM

Funkcia f je vypočítateľná na počítadlovom stroji (AM) práve vtedy ak existuje RAM, ktorý počíta f.

Ekvivalencia výpočtových modelov

AM ⇦ RAM ⇦ TM ⇦ AM

Pozn.

Platí aj opačná implikácia, dokazuje sa ťažšie.

AM ⇨ RAM

štrukturálna indukcia

jednojednoznačné zobrazenie namapovania registrov:

problém R_0 v RAM - akumulátor

predpoklad pre AM - R_1, R_2, R_3, R_4,... R_n

Zostrojím nový AM, kde prečíslujem registre

$R_i \rightarrow R_{i+1}$

Namapujem registre

AM_{NEW}		**RAM**
R_i	$\rightarrow$	R_i

Využívam skutočnosť, že RAM aj AM majú spočítateľne nekonečne veľa registrov.

AM ⇨ RAM   dôkaz indukciou a_k, s_k

a_k:

LOAD k
ADD =1
STORE k

s_k:

LOAD k
SUB =1
STORE k

AM ⇨ RAM dôkaz indukciou $M_1, M_2, M_3 \ldots, M_Z$

predpokladám, že mám $M_1, M_2, M_3 \ldots, M_Z$

indukčný predpoklad $M_j \sim I_j$

AM $M_1 M_2 M_3 \ldots M_Z$ skonštruujem tak, že inštrukcie budú za sebou

$$\left.\begin{array}{l}\rule{2cm}{0.4pt}\\ \rule{2cm}{0.4pt}\\ \rule{2cm}{0.4pt}\end{array}\right\} I_1$$

$$\left.\begin{array}{l}\rule{2cm}{0.4pt}\\ \rule{2cm}{0.4pt}\\ \rule{2cm}{0.4pt}\end{array}\right\} I_2$$

…

$$\left.\begin{array}{l}\rule{2cm}{0.4pt}\\ \rule{2cm}{0.4pt}\\ \rule{2cm}{0.4pt}\end{array}\right\} I_Z$$

AM ⇨ RAM

dôkaz indukciou $(M)_k$

chcem skonštruovať $(M)_k$

indukčný predpoklad $M_j \sim I_j$

musím zostrojiť cyklus

$(M)_k$

next: LOAD k

JZERO end

} I

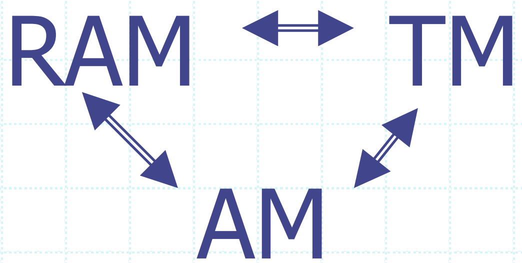
JUMP next

end:

Ekvivalencia výpočtových modelov

Veta 6.4.1 *(O ekvivalencií výpočtových modeloch) Nasledujúce výpočtové modely sú ekvivalentné:*

1. *Turingov stroj*
2. *Počítadlový stroj*
3. *Stroj RAM*



Ďakujem za pozornosť.

chuda@fiit.stuba.sk

STU FIIT