HAL
open science

Jednoduché makro suanpan na kreslení čínského a japonského abaku

Denis Roegel

To cite this version:

Denis Roegel. Jednoduché makro suanpan na kreslení čínského a japonského abaku. Zpravodaj (Československého sdružení uživatelů TeXu, ISSN 1211-6661), 2010, 20 (3), pp.138-151. inria-00548910

HAL Id: inria-00548910

https://inria.hal.science/inria-00548910

Submitted on 27 Dec 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Jednoduché makro suanpan na kreslení čínského a japonského abaku

Denis Roegel

Věnováno 荷花

Abstrakt

Článek představuje způsob, jak si lze v METAPOSTu připravit čínský (算盘, suànpan) a japonský (算盤, そろばん, soroban) abakus.

Jedná se o mechanické počítadlo usnadňující elementární matematické operace a úkony. Kuličkové počítadlo používané v prvních ročnících základní školy u nás je jednou z podob abaku. Podrobněji o abaku samotném viz např. webový rozcestník `http://www.ee.ryerson.ca/~elf/abacus/`.

Článek navíc představí způsob sčítání hodnot a implementaci tohoto algoritmu v METAPOSTu. Makro představené v článku lze rovněž stáhnout ze serveru CTAN.ORG.

Klíčová slova: METAPOST, makro `suanpan`, abakus, suànpan, soroban.

doi: 10.5300/2010-3/138

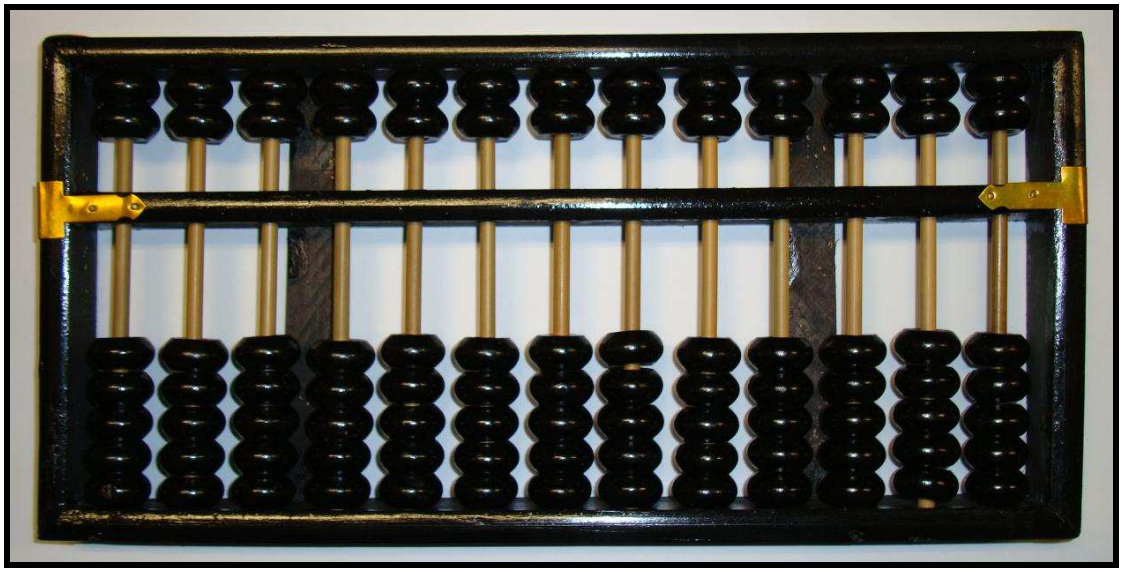
1. Představení abaku

Abakus je jedním z nejstarších počítadel, které se používá dodnes [6, 7, 9–13]. Používají jej především v Asii na základní matematické úkony. Ještě poměrně nedávno byl abakus (anglicky j. č. abacus, mn. č. abaci i abacuses, také v přepisu calculating table, board nebo frame) vyučován na čínských školách a byly z něj skládány zkoušky, pokud měl student zájem se ucházet o některá povolání. V Japonsku byla taková zkouška poprvé skládána v roce 1928 v Tokiu.

Zkušený počtář na abaku může být opravdu rychlý, nezadá si s vámi na běžné kalkulačce, tedy samozřejmě na základní operace s menšími čísly, jako je sčítání a násobení. Abakus může být využit i na náročnější úkony, jako je násobení, výpočet zlomků, druhé mocniny či druhé a třetí odmocniny. Na tento druh úkonů může být potřeba nestandardní abakus, dostatečně velký na to, aby dokázal ukládat potřebné mezivýpočty.

Zjednodušeně řečeno abakus je nástroj, který ukládá čísla pozicí svých korálků (kuliček, oblázků či kamínků; angl. beads) na tyčkách (v žlábcích, na drátech, šňůrkách apod.; angl. on rods). Uložená čísla mohou být jednoduchým

This article is a translation of the article "METAPOST macros for drawing Chinese and Japanese abaci", which first appeared in *TUGboat*, Volume 30 (1), pp. 74–79, 2009. Reprinted with permission. Translation and Czech abstract by Pavel Stříž. The author took the opportunity of this translation in order to make a few minor changes for the sake of clarity, on the suggestion of the translator.



Obrázek 1: Tradiční čínský abakus (算盘, suànpan) se všemi korálky na hodnotě nula. Fotografie je z autorovy kolekce.

způsobem měněna a počtář pracuje s tímto opakujícím se výpočetním vzorem poměrně snadno.

Abakus má dlouhou historii a existovala celá plejáda jeho variant, které v článku představeny nebudou. Stáří čínského abaku je odhadováno na tisíc let, možná je ještě starší. Jiné civilizace, jako byl Řím a Řecko, používaly podobné nástroje, kdy čísla byla ukládána za pomoci oblázků či speciálních žetonů.

V článku představíme makro naprogramované v METAPOSTu, které jednak nakreslí běžné asijské typy abaku a navíc kroky, jak se s nimi počítá.

2. Typy abaku

Zaměříme se na dva typy abaku – standardní verzi čínského a japonského počítadla, které jsou používány i v současnosti.

2.1. Čínský typ abaku: *suànpan*

Čínská verze je nazývána 算盘 (*suànpan*). Slovo 算 (suàn) v čínštině znamená „počítat“ a slovo 盘 (pan) „v rámečku“ nebo na „na destičce“. Abakus *suànpan* může mít celou řadu délek. Běžně má 13 tyček s pěti korálky ve spodní části rámečku (někdy jako pozemské korálky; angl. earth beads) a dvěma korálky (někdy jako nebeské korálky; angl. heaven beads) v horní části (model 2/5 či 2:5), viz Obrázek 1. Horní a spodní část je oddělena příčkou či jinou formou přepážky.

Každý korálek v horní části má hodnotu pěti korálků v části spodní. Pracujeme-li se čtyřmi spodními korálky a jednou horní, je to dostatečné na výpočty v desítkové soustavě. Vypadá to, že prémiové korálky byly původně použity na vyjádření číslic šestnáctkové soustavy, což byla soustava vhodná pro tradiční vážení, kdy jeden jīn (斤) je roven šestnácti liǎng (两) (což je přibližně 50 gramů). Můžeme zároveň vzít v úvahu, že tyto prémiové korálky zjednodušovaly a urychlovaly některé z matematických operací, více o tom [11, str. 85].

2.2. Japonský typ abaku: *soroban*

Japonská verze 算盤 (そろばん, *soroban*) je obdobou čínského *suànpanu*, která ji dala svůj původ. „盤“ je tradiční znak pro žlábek, tento symbol je používán i v současnosti. Také základní verze *sorobanu* má obvykle 13 tyček, jsou zde jen čtyři korálky ve spodní části a jeden v části horní (model 1/4). V některým případech se můžeme setkat s variantou, že je ve spodní části pět korálků a v horní je jen jeden (model 1/5).

Varianta *soroban* má ještě jednu vlastnost, která ji odlišuje od *suànpanu*, a to, že každá třetí tyčka je označena puntíkem. Jedná se o tzv. jednotkové tyčky (angl. unit rods), můžeme si to v současnosti přiblížit k psaní oddělovačů tisíců. Usnadňuje to na *sorobanu* výpočty a zároveň to zpřehledňuje nastavování korálků. Tímto jednoduchým trikem si lze zvýraznit desetinnou tečku.

3. Makro `suanpan` připravené v METAPOSTu

Jak se s abakem pracuje ukážeme na obrázcích, které jsme vygenerovali v METAPOSTu. METAPOST je silný grafický nástroj, který se výborně hodí právě na takové technické a geometrické rýsování [3,5]. Všechny podklady v tomto článku byly připraveny makrem `suanpan`, volně dostupného na serveru CTAN.ORG. Na makro je však potřeba nahlížet jen jako na jádro, které však může být snadno rozšířeno, např. na úpravu tvaru korálků nebo na představení složitějších algoritmů než toho, který si za chvíli ukážeme.

V době psaní tohoto článku byly známy jen další dva balíčky ze světa TEXu, konkrétně od Alaina Delmotta, na kresbu *sorobanu*, buď přes balíček PSTricks nebo PGF, bohužel ani jeden z těchto balíčků (zatím) neumí žádný z výpočetních postupů [1,2].

4. Počítání na abaku

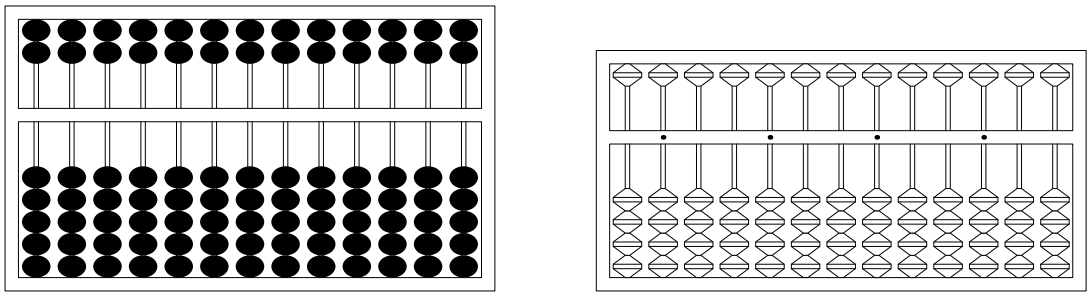
Výpočty s abakem začínají jeho vynulováním, tedy navrácením všech korálků do původní polohy, jinými slovy co nejvzdáleněji od příčky, poté nastavením

první hodnoty posuny korálků, a poté následuje matematický úkol dle známého postupu. Výsledek se dá následně přečíst.

4.1. Základní pozice abaku

Obrázek 1 ukazuje základní pozici skutečného *suànpanu* a Obrázek 2 srovnává abakus čínský (obrázek vlevo) s japonským (ten napravo). Dvě části abaku jsou rozděleny dělicí příčkou, chceme-li přepážkou či trámem (angl. horizontal bar, crosspiece, beam nebo reckoning bar).

V základní poloze jsou všechny korálky co nejdále vzdáleny od příčky, představuje to na všech tyčkách hodnotu nula. Každý korálek představuje jeden řád v desítkové (někdy i šestnáctkové) soustavě. Jednotky jsou nejčastěji vpravo. Tyčky se obvykle číslují, ale tato vlastnost může být v našem makru vypnuta nastavením přepínače `rod_numbers` typu pravda-nepravda (angl. boolean), jak je předvedeno v ukázce.



Obrázek 2: Základní pozice počítadel: 算盘(*suànpan*, vlevo) a 算盤(*soroban*, vpravo).

Užijeme-li makro `suanpan`, lze základní pozici *suànpanu* získat následujícím způsobem:

```
input suanpan
setup_abacus(N=13, NBL=5, NBU=2,
             bead="suanpan", units=0);
beginfig(1);
  rod_numbers:=false;
  reset_abacus;
  draw_abacus;
endfig
end
```

Makrem `setup_abacus` můžeme nastavit počet tyček (`N`), stejně tak počet korálků v obou částech abaku (`NBL` a `NBU`), typ korálků (`bead`) a zdali si přejeme značení u každé třetí tyčky (`units`). Parametry jsou zadávány jako dvojice *proměnná=žádaná hodnota*. V současné verzi makra lze nastavit dva typy ko-

rálků, odpovídající textovým řetězcům `"suanpan"` (téměř korálky kruhu, jemně zploštělé po délce) a `"soroban"` (korálky tvaru dvoukužele).

4.2. Nastavování hodnot

Nastavení hodnoty na abaku znamená posunutí korálků k příčce. Korálek ve spodní části má hodnotu jedné příslušného řádu (jednotky, desítky, stovky atd.) a korálek v horní části představuje hodnotu pěti. Nastavení čtyř korálků ve spodní části a jednoho korálku v horní dává dohromady $5+4 = 9$. Pokud jsou použité všechny korálky na čínském typu 2/5, a v horní části se stále pracuje s ekvivalentem pěti, dostáváme maximální hodnotu na tyčce rovnu $5 + 5 + 5 = 15$. Tímto způsobem můžeme nastavit všechna čísla od nuly (výchozí postavení) po hodnotu patnáct (maximální hodnota).

Co se týká makra `suanpan`, zde je počet korálků v každé části uložen ve dvou polích s dimenzí. Libovolná hodnota může být u každého pole nastavena ručně takto (Obrázek 3):

```
beginfig(3);
  reset_abacus;
  valL[1]:=2; valL[3]:=5;
  valU[2]:=1; valU[4]:=2;
  draw_abacus;
endfig;
```

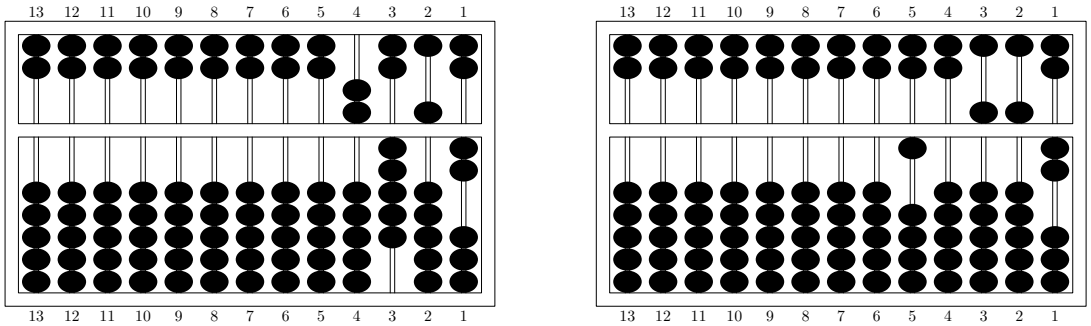
Zpracování tímto způsobem může být užitečné tehdy, když je potřeba nastavit nějakou v desítkové soustavě nestandardní situaci (nastavení čísla 10 552 na Obrázku 3, vlevo). Tohle je konkrétně výše zmíněný případ, kdy bylo potřeba na tyčce nastavit pět korálků ve spodní části. Obvykle nám ve spodní části vystačí vždy jen čtyři korálky. Obdobná výjimka platí pro dva korálky v horní části, počítáme-li v desítkové soustavě.

Makro `suanpan` v současné verzi nepodporuje výpočty v šestnáctkové soustavě, ale mohou být jednoduše zařízeny nadstavením výpočtů založených na soustavě desítkové.

Pomocné makro `set_abacus_val` dokáže operaci nastavení počátečních hodnot zautomatizovat (Obrázek 3, vpravo):

```
beginfig(4);
  reset_abacus;
  set_abacus_val("10552");
  draw_abacus;
endfig;
```

Máme-li pod `N` uložen počet tyček, je automaticky brán zprava jen navolený počet znaků z textového řetězce vstupujícího do makra `set_abacus_val`. Ostatní cifry jsou ignorovány.



Obrázek 3: Nestandardní pozice hodnoty 10 552 v desítkové soustavě (vlevo) a její standardní vyobrazení (vpravo).

4.3. Operace sčítání

Jakmile máme počáteční hodnotu nastavenu, můžeme ji systematicky měnit. V tomto článku se zaměříme jen na operaci sčítání. Dokonce i pro tak triviální operaci můžeme zvolit několik způsobů, my si ukážeme typický způsob sčítání ne zprava doleva, ale zleva doprava.

Abychom ukázali, jak to funguje, připravili jsme v makru `suanpan` makro, chceme-li funkci, `add_val`, která sčítání rozloží na nezbytný počet kroků. Jen si dejme pozor, že tohle makro nemůže být běžnou součástí prostředí `beginfig` a `endfig`, je to z toho důvodu, že makro tato prostředí generuje za svého běhu.

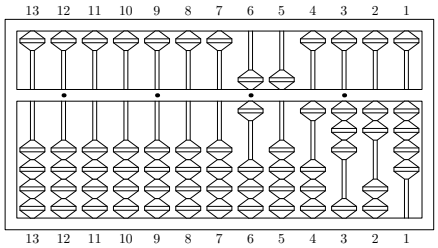
Ukážeme si příklad na *sorobanu* s počáteční hodnotou 651 324 tak, jak to zmiňuje následující zdrojový kód (Obrázek 4):

```
setup_abacus(N=13, NBL=4, NBU=1,
             bead="soroban", units=1);
set_abacus_val("651324");
add_val(v="82363456", iv=100, fig=true);
```

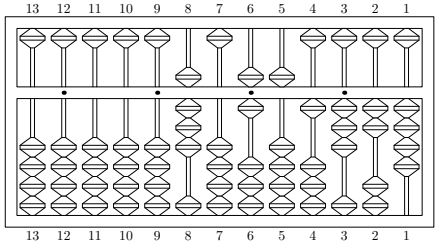
Užíváme-li jako řídicí soubor `abacus.mp`, pak zmíněná sekvence příkazů v něm uložených vygeneruje soubory `abacus.100`, `abacus.101`, ..., `abacus.108`, které již mohou být rutinně přiloženy do TEXového dokumentu.

Po vynulování následují kroky dle Obrázku 4. Úvodní stav (a) ukazuje nastavení prvního sčítance 651 324, zde náš proces sčítání začíná. V prvním kroku (b) přidáváme 8 na osmou tyčku tím způsobem, že přesuneme k příčce tři spodní korálky a jeden horní. S ostatními korálky není hýbáno. Následuje krok (c), který přidává 2 na sedmou tyčku. Zatím máme život snadný, neboť změny se dějí vždy na jedné tyčce, poněvadž byly nastaveny na hodnotu 0. Je to dáno tím, že první sčítanec nemá řád miliónů ani desetimiliónů. V kroku (d) přidáme 3 na šestou tyčku. Z hodnoty 6 se stává hodnota 9.

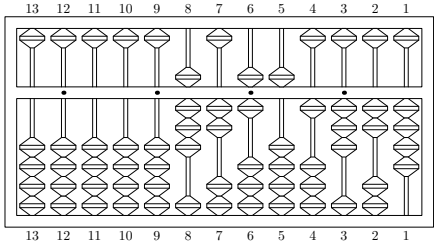
Krok (e) přidává 6 na pátou tyčku, která obsahuje 5. To nám součtem dává 11. Na páté tyčce nastavíme hodnotu 1 a jedničku si budeme chvíli pamatovat a pokusíme se tuto jedničku přidat na tyčku šestou (tyčka vlevo). Tato vzniklá situace



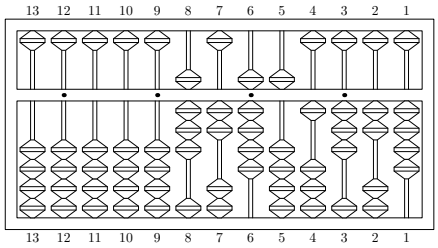
(a) Nastavíme 651 324.



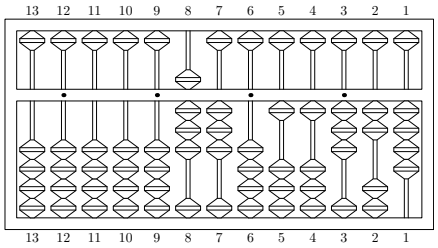
(b) Po přičtení 80 000 000.



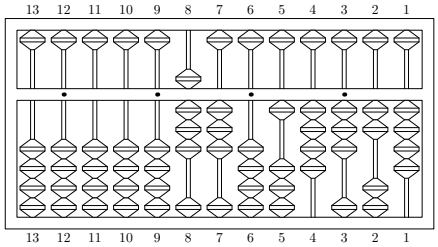
(c) Po přičtení 2 000 000.



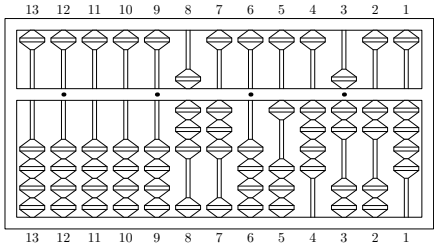
(d) Po přičtení 300 000.



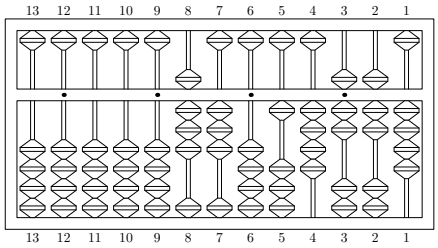
(e) Po přičtení 60 000.



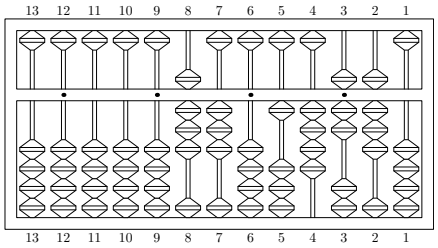
(f) Po přičtení 3 000.



(g) Po přičtení 400.



(h) Po přičtení 50.



(i) Po přičtení 6.

Obrázek 4: Znázornění rozkladu sčítání do devíti kroků na *sorobanu*. (a) znázorňuje nastavení prvního sčítance 651 324, a přidáváme druhého sčítance 82 363 456 v osmi krocích (b)–(i), každý krok reprezentuje jednu cifru druhého sčítance.

nám zatím ukazuje, že jsme tedy změnili počet korálků ve spodní části na tyčce pět, nenastala žádná změna ve smyslu přičtení v její horní části, která se nuluje. Zaměříme naši pozornost na tyčku šest, kde chceme přidat jedničku. Šestá tyčka obsahuje hodnotu 9. Přidáním jedničky se dostáváme na 10 a je potřeba dalšího přesunu jedničky. Vynulujeme šestou tyčku a přidáme jedničku na tyčku sedm. Ta obsahuje hodnotu 2 a tu zvedneme na hodnotu 3. Dalších přesunů už není třeba a krok (e) je hotov, přičetli jsme 60 000.

Celý tento proces se opakuje cifra po cifře, dokud není celý rozložený sčítanec připsán. Vidíme, že přičtení cifry sčítance se občas rozloží na dílčí kroky. Podrobněji nebudeme detaily v článku rozvádět, v principu se jedná jen o analogii práce s jednou tyčkou, jen v několika krocích rozloženou. K procvičení si lze vyzkoušet součet $999\,999\,999 + 1$.

Pokud by výklad u zmíněných maker nebyl dostatečný, odkazujeme čtenáře přímo na zdrojový kód `suanpan.mp` z CTAN.ORG.

Makro `add_val` může být použito i bez generování obrázků tím, že se nastaví `false` u proměnné `fig`. V takovém případě proběhne jen základní algoritmus sčítání a součet je uložen v polích.

Je tu samozřejmě možnost, že by součty mohly být simulovány tím, že výpočet bude proveden mimo makro `suanpan` a hodnoty budou vždy nastaveny pro každý příchozí požadavek. Takto by bylo možné využít makro `suanpan` jako nástroj vykreslování pro jiná makra, programy či nástroje.

Přikládáme ukázku součtu, kdy nevznikají obrázky z jednotlivých mezisoučtů.

```
beginfig(200);
  reset_abacus;
  reset_abacus_gray;
  set_abacus_val("82951324");
  draw_abacus;
endfig;
beginfig(201);
  set_abacus_val("82951324");
  add_val(v="60000", iv=100, fig=false);
  draw_abacus;
endfig;
```

Operace sčítání může způsobit problém s přeplněním a nastavila by se proměnná `overflow` typu pravda-nepravda na hodnotu `true`. Před uskutečněním sčítání si makro `add_val` vynuluje tuto proměnnou na `false`.

4.4. Triky na zrychlení výpočtů

Pokud chceme zefektivnit práci s abakem, je užitečné si zapamatovat některé vzory, které se objevují často a dávají nám prostor k jejich automatizaci. Na

jednoduché ukázce si to osvětlíme. Pokud některá z tyček je ve spodní části nastavena na hodnotu 3 a jeden korálek máme přidat, pak nezbývá nic jiného než čtvrtý korálek posunout směrem k příčce. Zde nemáme co vylepšovat, počet stupňů volnosti je nula.

Pokud je však situace drobně upravena, tedy máme-li přidat tři korálky místo jednoho ke stávajícím třem, pak můžeme pravděpodobně u začátečníka očekávat myšlenkový proces založený na výpočtu $3 + 3 = 6$, poté odečtením 5 a nastavením jednoho korálku ve spodní i horní části abaku. Věřte či nikoliv, ale tato situace je neefektivní, poněvadž výpočetní zátěž leží na počtáři.

Uvažujme takto. Víme-li, že tři korálky nemohou být v dané chvíli přemístěny, pak můžeme rovnou zvážit vztah $3 = 5 - 2$ a dostat se ke dvěma úkonům: za prvé, *přidání* jednoho korálku (hodnoty 5) v horní části, a za druhé, *odebrání* dvou korálků v části spodní. Tohle je příjemná matematická zkratka, která nevyžaduje výpočet $3 + 3 = 6$, a to jen díky tomu, že si zavčas všimneme, že nemůžeme ve spodní části přidat ke třem stávajícím korálkům tři další.

Podobně nerealizovatelné úkony si lze představit i v horní části. Můžeme si připravit podobné schéma. Jakmile není možné přidat korálek o hodnotě pět do horní části, uvažme užití formule $5 = 10 - 5$, technicky přidáme jeden korálek (z pohledu aktuální tyčky je hodnoty 10) ve spodní části další tyčky (opět vlevo od aktuální) a ubereme korálek v horní části původní tyčky. A opět se tento postup opakuje tak dlouho, dokud nesečteme celý sčítanec.

Uvažme situaci, kdy chceme přidat pět nebo více korálků na libovolnou tyčku ve spodní části. Tuto situaci můžeme zapsat buď rozepsáním na $5 + a$ nebo $10 - b$, a je-li jedna z nich použitelná, bude aplikována. Konkrétně můžeme mít tři korálky na tyčce v její spodní části a chceme přidat dalších šest. Buď můžeme tuto situaci rozepsat jako $6 = 5 + 1$ nebo $6 = 10 - 4$. Druhý rozklad nelze technicky zrealizovat, protože nemůžeme odebrat čtyři korálky, když máme jen tři. V tomto případě je však možná první situace. Přidáme jeden korálek ve spodní části a pokusíme se přidat jeden korálek v horní části. Pokud by to v horní části nebylo možné, provedeme další rozklad atd.

Náročnější matematické úkony, jako je násobení, dělení, druhá odmocnina atd., mohou být aplikovány rychle a účinně za použití pomocných tabulek, které si předtím musel počtář vštěpit do paměti. Ukázky takových tabulek pro *soroban* předkládá např. Knott [7].

5. Účelová makra pro rozšíření abaku

Chceme-li někomu vysvětlit práci s abakem, je poměrně často výhodné využít nějaké formy značení nebo zvýraznění jednotlivých korálků. Makro `suanpan` nabízí dvě možnosti. Buď je volený korálek vykreslen jinou barvou, konkrétně šedou, nebo je na něj přidán popisek.

Je poměrně snadné využít makra `set_abacus_gray` a vykreslit vybrané korálky v šedé barvě. Makro má tři parametry, zadávané opět ve formě párů *proměnná=žádaná hodnota*. Proměnná `deck` nám upřesní, jestli pracujeme s horní (`upper`) nebo spodní (`lower`) částí abaku. Další dvě proměnné jsou textové řetězce udávající korálky brané zprava. Hodnota `below` odpovídá korálkům vespod navolené části (horní nebo spodní části abaku), opakem je hodnota `above`. Například zapsání hodnoty 2 při `below` znamená, že budou šedou barvou vykresleny dva korálky zespodu. Tedy ty dva, které jsou brány za první kandidáty na posun k příčce (při volbě `lower`), resp. ty dva, které se jako první vrací do základního stavu (při volbě `upper`).

Při `below` to je obdobné, jako kdybychom brali vrcholky stalagmitu; při `upper` to je, jako bychom odebírali spodní špičky stalaktitu.

V současné verzi makra není tento způsob obarvování zautomatizován, i když je to samozřejmě technicky možné. Můžeme si představit celou řadu možných schémat, které jsme do makra nezanesli. Realizace v současné verzi je podpořena jen základními příkazy.

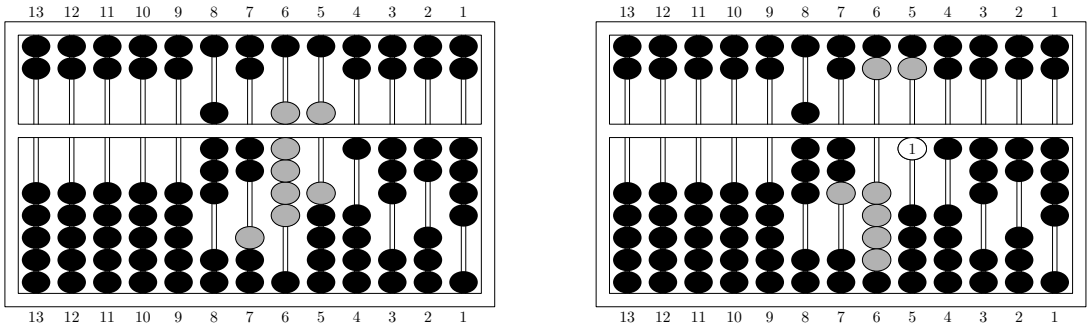
Nechť nám za ukázku poslouží součet z dřívějšího příkladu vyobrazeného na Obrázku 4, konkrétně přechod z kroku (d) do (e), tentokrát však tak, aby byly zvýrazněny všechny posuny korálků. Jednalo se konkrétně o situaci, kdy jsme k mezihodnotě 82 951 324 přičítali 60 000. Výsledek našich snah si můžete prohlédnout na Obrázku 5 a zde přikládáme příslušné zdrojové kódy:

```
beginfig(202);
  reset_abacus;
  reset_abacus_gray;
  set_abacus_val("82951324");
  set_abacus_gray(deck="lower",
     below="1010000", above="0400000");
  set_abacus_gray(deck="upper",
     below="0110000", above="0000000");
  draw_abacus;
endfig;
```

```
beginfig(203);
  reset_abacus_gray;
  add_val(v="60000", iv=100, fig=false);
  set_abacus_gray(deck="lower",
     below="0400000", above="1010000");
  set_abacus_gray(deck="upper",
     below="0000000", above="0110000");
  draw_abacus;
  mark_abacus(5,5)(btex 1 etex);
endfig;
```

V daných ukázkách makro `reset_abacus_gray` plní jen ulohu vynulování dříve šedou barvou vyznačených korálků. Bude to pro čtenáře příjemnější, aby viděl právě jen ten jeden přechod.

Dalším nově představeným makrem na vyznačení korálků je `mark_abacus`. Tohle makro umí překreslit korálek tak, aby mohl obsahovat (krátký) popisek. Konkrétně `mark_abacus(5,5)(btex 1 etex)` zapíše cifru jedna na pátý korálek odspodu na pátou tyčku počítanou zprava. Jednou z výhod tohoto přístupu je, že i když se později korálek posune, značka u něj zůstane, aniž bychom příkaz jakkoliv upravovali nebo si jej znovu volali.



Obrázek 5: Zvýraznění všech přesunů v jednom z mezikroků při sčítání 829 513 24 s 60 000. Všechny korálky, které byly přesunuty, jsou vykresleny šedou barvou, a navíc jsme si vyznačili aktuální hodnotu na tyčce, se kterou bylo pracováno. V našem případě to byla hodnota 60 000, tedy pátá tyčka.

6. Využití abaku v jiných číselných soustavách

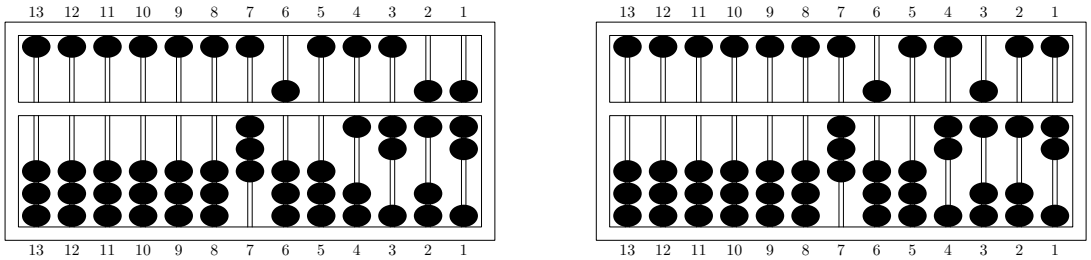
Jak jsme dříve zmínili, čínský abakus může být použit na výpočty v desítkové i v šestnáctkové soustavě, záleží jen na tom, jestli použijeme dva bonusové korálky (jeden v horní části a druhý ve spodní). Po chvíli přemýšlení však můžeme abakus upravit tak, aby sloužil i pro jiné číselné soustavy.

Obrázek 6 nám ukazuje sčítání v osmičkové soustavě. Každá tyčka má čtyři korálky, tři ve spodní části s hodnotou jedna a jeden korálek v horní části s hodnotou čtyři. Tímto způsobem sestavíme hodnoty nula až sedm. Levý obrázek představuje nastavení čísla $3\,401\,256_8$, přičtením 1234_8 dostáváme součet $3\,402\,512_8$, což je vyobrazeno vpravo.

Příprava těchto obrázků je přímočará. Upravíme počet korálků na každé tyčce, dále nastavíme proměnnou `vbu`, která představuje počet korálků v horní části abaku. Takový abakus v osmičkové soustavě má hodnotu horního korálku čtyři, což odpovídá počtu korálků ve spodní části plus jeden. Stačí nám nyní zapsat tento zdrojový kód, abychom dostali náhled před sčítáním a po něm.

Podobně můžeme připravit další číselné podklady. Jen upozorníme zkušené počtáře, že je nutné upravit mnemotechnické pomůcky, které jsme poodhalili v tomto článku dříve tak, aby těmto novým podmínkám vyhovovaly.

```
vbu:=4; % Hodnota kuličky v horní části abaku.
setup_abacus(N=13, NBL=3, NBU=1,
             bead="suanpan", units=0);
beginfig(300);
  reset_abacus;
  reset_abacus_gray;
  set_abacus_val("3401256");
  draw_abacus;
endfig;
beginfig(301);
  add_val(v="1234", iv=100, fig=false);
  draw_abacus;
endfig;
```



Obrázek 6: Sčítání na abaku v osmičkové soutavě. Hodnota vlevo je $3\,401\,256_8$, po přičtení hodnoty 1234_8 dostáváme součet $3\,402\,512_8$, který je k nahlédnutí vpravo.

7. Závěry a možná rozšíření makra suanpan

Cílem článku bylo ukázat jednoduché makro kreslící čínský a japonský abak. Snažili jsme se především poskytnout jisté základní stavební kameny. Vylepšení jsou možná jak na straně grafiky, tak na úrovni algoritmů. Abakus může být vykreslen mnohem realističtěji, a co je obzvláště příjemné, další typy korálků lze jednoduše přidat.

Největší prostor pro rozšíření a vylepšení je ale na straně algoritmů. Současná verze makra obsahuje jen jeden postup na sčítání. Mohli bychom si např. připravit algoritmus pro sčítání z pravé části do levé (tj. od jednotek po vyšší řády). Podobně bychom mohli přidat řadu dalších algoritmů.

Náročnější matematické úkony, jako je násobení, dělení, výpočet druhé či třetí odmocniny, ale celá řada dalších [4, 7, 8, 11], by nemělo být náročné do jádra makra `suanpan` implementovat.

Pro každý z takto zvažovaných algoritmů by bylo vhodné jej naprogramovat tak, aby bylo možné na výstupu, graficky a výpisem proměnných, sledovat jednotlivé kroky a mezikroky matematických úkonů.

Makro `suanpan` vám k tomu všemu dává stavební základnu.

Seznam literatury

[1] Delmotte, Alain. Soroban abacus: package pgf-soroban. [Balíček pgf-soroban na kresbu abaku japonského typu soroban přes TEXový balíček PGF.] [online cit. 12. 11. 2007] Balíček je dostupný na URL: `http://ctan.org/tex-archive/graphics/pgf/contrib/pgf-soroban/`

[2] Delmotte, Alain. Soroban abacus: package pst-soroban. [Balíček pst-soroban na kresbu abaku japonského typu soroban přes balíček PSTricks.] [online cit. 3. 9. 2009] Balíček je dostupný na URL: `http://ctan.org/tex-archive/graphics/pstricks/contrib/pst-soroban/`

[3] Goossens, Michel; Mittelbach, Frank; Rahtz, Sebastian; Roegel, Denis; Voß, Herbert. *The LATEX Graphics Companion.* [Velký průvodce grafikou v LATEXu.] 2. vyd. USA, Boston: Addison-Wesley Professional, 2007. ISBN 978-0-321-50892-8.

[4] Heffelfinger, Totton; Flom, Gary. 算盤 Abacus: Mystery of the Bead. [算盤 Abakus: záhada korálků.] [online cit. 16. 1. 2007] Dostupné na URL: `http://webhome.idirect.com/~totton/abacus/`

[5] Hobby, John. METAPOST: A User's Manual. [METAPOST: Uživatelská příručka.] [online cit. 2. 10. 2009] Aktualizovaná verze původního manuálu. Dostupná z URL: `http://www.tug.org/docs/metapost/mpman.pdf`.

[6] Ifrah, Georges. *The Universal History of Computing: From the Abacus to the Quantum Computer.* [Historie výpočetního světa: od abaku až po kvantové počítače.] 1. vyd. USA, New York: John Wiley and Sons, 2001. ISBN 978--0-471-44147-2.

[7] Knott, Cargill Gilston. The Abacus, in Its Historic and Scientific Aspects. [Abakus z pohledu historického a vědeckého.] *Transactions of the Asiatic Society of Japan*, Volume 14: 18–71, 1886. ISSN 0913-4271.

[8] Kojima, Takashi. *Advanced Abacus: Japanese Theory & Practice.* [Abakus pro pokročilé: teorie a praxe v Japonsku.] Japonsko, Tokio: Charles E. Tuttle and Company, 1963. ASIN B0007DNGUQ.

[9] Shu-T'ien, Li. Origin and Development of the Chinese Abacus. [Vznik a rozvoj čínského abaku.] *Journal of the ACM*, Volume 6 (1): 102–110, 1959. ISSN 0004-5411. doi:10.1145/320954.320962

[10] Martzloff, Jean-Claude. *A History of Chinese Mathematics.* [Historie čínské matematiky.] USA, New York: Springer, 2006. ISBN 978-3-540-33782-9.

[11] Moon, Parry. *The Abacus: Its History, Its Design, Its Possibilities in the Modern World.* [Abakus: historie, umělecké provedení a možnosti využití v současnosti.] USA, New York: Gordon and Breach Science Publishers, 1971. ISBN 978-0-677-01960-4.

[12] Needham, Joseph; Wang, Ling. *Science and Civilisation in China, Vol. 3: Mathematics and the Sciences of the Heavens and Earth.* [Věda a civilizace v Číně, svazek třetí: Matematika a vědy nebes a o Zemi.] Velká Británie, Cambridge: Cambridge University Press, 1959. ISBN 978-0-521-05801-8.

[13] Smith, David Eugene; Mikami, Yoshio. *A History of Japanese Mathematics.* [Historie japonské matematiky.] Chicago: The Open court publishing company, 1914. Přetištěno jako Smith, David Eugene; Mikami, Yoshio. *A History of Japanese Mathematics.* 1. vyd. Cosimo Classics, 2007. ISBN 978-1--60206-664-9.

Summary: METAPOST macros for drawing Chinese and Japanese abaci

This article shows how Chinese (算盘, suànpan) and also a Japanese version of abaci (算盤, そろばん, soroban) can be drawn with METAPOST, and illustrates it with the details of a simple algorithm.

The source codes are included as small parts in the article commented in detail. You may find the original English version of the article in *TUGboat*, see `http://www.tug.org/members/TUGboat/tb30-1/tb94roegel-abacus.pdf`.

Keywords: METAPOST, Chinese and Japanese abacus, abaci, abacuses, suànpan, soroban, `suanpan` macro.

Denis Roegel, roegel@loria.fr
http://www.loria.fr/~roegel
LORIA – Campus Scientifique, BP 239
F-54506 Vandœuvre-lès-Nancy Cedex, France