

- [2] *Jimoyiannis, A., Komis, V.*: Computer simulations in physics teaching and learning: a case study on students' understanding of trajectory motion. *Computers & Education* [online]. 2001, roč. 36, č. 2, s. 183-204 [cit. 2012-11-26]. ISSN 0360-1315. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S0360131500000592>
- [3] *Černý, M.*: Metodický portál, Články: „Počítačové simulace ve výuce fyziky“ [online]. 09. 11. 2010.[cit. 09. 11. 2010.]. ISSN 1802-4785. Dostupný z: <http://clanky.rvp.cz/clanek/c/G/9707/POCITACOVE-SIMULACE-VE-VY-UCE-FYZIKY.html>
- [4] *Burk, J.*: Raising the energy level of my physics classes Home About Computational Modeling More than a blogroll Physics Teacher Camp Algodoo physics-a virtual physics lab for computational thinking. *Quantum Progress* [online]. 2011 [cit. 2012-11-26]. Dostupné z: <http://quantumprogress.wordpress.com/2011/11/27/algodoo-physics-a-virtual-physics-lab-for-computational-thinking/>
- [5] *Drage, Ch.*: Putting the fizz into physics – the Algodoo revolution. *MJO: learning, teaching, technology*, [online]. 2011 [cit. 2012-11-26]. Dostupné z: <http://www.agent4change.net/bett-week/news/1298>

Indické násobenie a program MultInd

TOMÁŠ BENE

Fakulta prírodných vied UKF, Nitra

V tomto článku bude predstavený jeden z najstarších algoritmov, ktoré ľudstvo v minulosti používalo. Jedná sa o známy algoritmus indického násobenia, ktorý po drobných úpravách používame dodnes. V článku bude predstavený vlastný počítačový program MultInd, ktorý sa zaoberá problematikou indického násobenia. Program využíva viacero rôznych funkcií, pomocou ktorých bude pre žiakov zaujímavý a atraktívny.

Indické násobenie

Najskôr uvedieme základné vlastnosti súčasného spôsobu násobenia, ktorý sa vyučuje na školách. Súčasný spôsob násobenia sa zakladá na vý-

počte čiastočného súčinnu. Pri počítaní čiastočných súčinnov sa využíva násobenie a zároveň sa pripočítavajú desiatkové zložky predchádzajúcich násobilkových spojov. Spojením sčítania a násobenia vzniká náročný proces, ktorý môže byť pre žiakov problematický. Aby vyriešili príklad $6\,521 \times 254$, musia postupovať nasledovne.

6521
. 254

26084
32605
13042

1656334

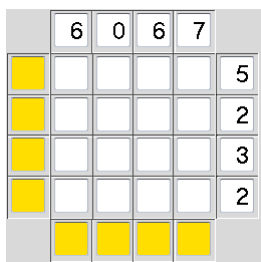
Oba činitele podpisujú pod seba, jednotky pod jednotky, desiatky pod desiatky, atď. Pokračujú čiastočným násobením každej číslice druhého činiteľa s každou číslicou prvého činiteľa známym spôsobom. Nakoniec všetky čiastočné súčiny sčítajú a dostanú konečný výsledok. Pri počítaní čiastočných súčinnov okrem násobenia využívajú aj sčítanie desiatkových zložiek, ktoré vznikli pri predchádzajúcich násobilkových spojoch.

K algoritmu, v ktorom je násobenie od sčítania oddelené, dospejeme postupnými úpravami tak, že zápis prepíšeme do štvorcovej siete, kde smer sčítania vyznačíme uhlopriečkami a vzniknú trojuholníkové políčka. Tým vo výpočte oddelíme násobenie od sčítania, ktoré realizujeme uhlopriečne.

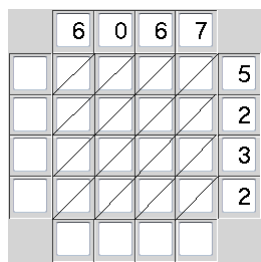
Teraz si prakticky na príklade popíšeme, ako daný algoritmus indického násobenia funguje. Tento algoritmus aplikuje pri násobení mechanický postup. Pokúsime sa vyriešiť príklad $6\,067 \times 5\,232$. Najprv nakreslíme sieťovú schému, ktorej rozmery budú m krát n , kde m je počet číslic prvého činiteľa a n je počet číslic druhého činiteľa. V našom prípade sieťová schéma bude štvorcová s rozmermi 4×4 . Na jej horný okraj zľava doprava napíšeme prvý činiteľ a na pravý okraj zhora dole napíšeme druhý činiteľ. Výsledok týchto úprav vidíme na obr. 1.

Na obr. 1 tiež vidíme naznačené na ľavom a dolnom okraji žltou farbou miesto pre výsledok. Ďalej je potrebné rozdeliť štvorcovú schému hlavnou uhlopriečkou tak, aby každé vnútorné políčko schémy bolo rozdelené. Toto rozdelenie slúži k vypočítaniu výsledku, ktoré bude popísané v ďalšej časti. Obr. 2 zobrazuje sieťovú schému pripravenú k samotnému výpočtu.

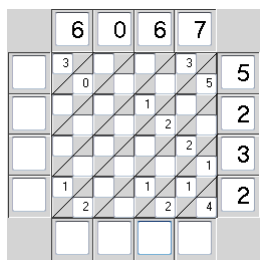
Dostávame sa k samotnému riešeniu príkladu. Postup riešenia je veľmi jednoduchý. Vynásobíme každú číslicu prvého činiteľa s každou číslicou druhého činiteľa a výsledok tohto násobenia zapíšeme do prieniku riadku a stĺpca, v ktorom tieto násobené číslice ležia. Keďže je každé políčko rozdelené uhlopriečkou na dve časti, jednotky zapíšeme pod uhlopriečku a desiatky nad uhlopriečku. Činitele môžeme násobiť v ľubovoľnom poradí. Na obr. 3 je čiastočné riešenie príkladu.



Obr. 1



Obr. 2



Obr. 3



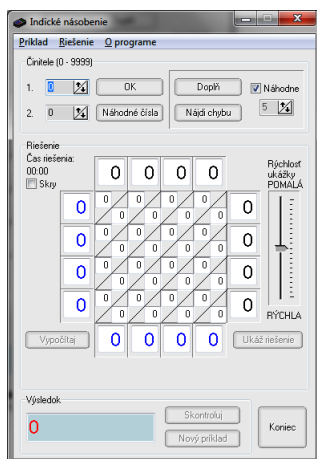
Obr. 4

Pokračujeme v násobení až kým schéma nebude úplne vyplnená. Nakoniec pristúpime k nájdeniu výsledku po uhlopriečkach. Sčítame všetky číslice v sieťovej schéme, ktoré sa nachádzajú medzi dvoma rovnakými uhlopriečkami. Nakoľko pri sčítaní dochádza k prechodu desiatok do ďalšej uhlopriečky, začíname od pravej dolnej uhlopriečky a končíme ľavou hornou. Obr. 4 farebne odlišuje skupinu číslic, ktoré sa spolu sčítajú, pričom začíname políčkami označenými žltou farbou, pokračujeme modrou, zelenou, atď. Výsledky sčítania zapisujeme do spodného a ľavého okraja príslušnej diagonály. Celkový výsledok dostaneme po prečítaní číslic smerom od ľavého okraja k dolnému okraju.

Konečný výsledok nášho príkladu $6\,067 \times 5\,232$ je $31\,742\,544$. Na lepšie precvičenie tejto problematiky vznikol počítačový program MultInd, ktorý je k dispozícii na voľné stiahnutie z linku, ktorý je uvedený na konci článku. Program je možné použiť na prvom aj druhom stupni základných škôl a využíva sa na precvičenie malej násobilky a sčítania v obore prirodzených čísel. Program využíva viacero techník precvičovania, ktoré si teraz popíšeme.

Program MultInd

Po spustení programu sa zobrazí jeho hlavné okno (obr. 5).



Obr. 5

Práca s programom začína nadefinovaním oboch činiteľov do editačných políčk vľavo hore. Rozsah týchto vstupných polí je obmedzený na celé kladné čísla od 0 do 9 999. Môžeme si vybrať z dvoch spôsobov, ako tieto činitele do programu nadefinujeme:

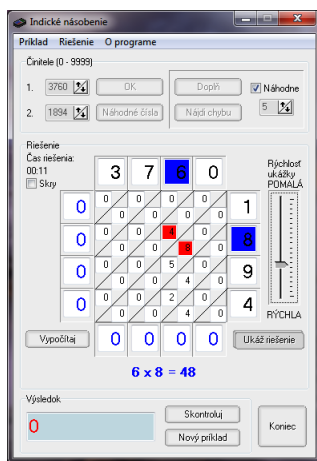
1. Priamo zadáme vstupné činitele a klikneme na tlačidlo „OK“
2. Klikneme na tlačidlo „Náhodné čísla“, program za nás vygeneruje vstupné činitele a my ich už len potvrdíme tlačidlom „OK“

Počas zadávania činiteľov sú ostatné funkcie programu nedostupné. Po potvrdení činiteľov budú tieto prepísané do pripravenej sieťovej schémy rovnakým spôsobom, ako bolo popísané v predchádzajúcej kapitole. Našou úlohou bude vyplniť jednotlivé násobilkové spoje v strede schémy. Výsledok dostaneme sčítaním všetkých čísel v jednotlivých uhlopriečkach. Tento výsledok sa nám automaticky prepíše aj do spodného políčka „Výsledok“.

Výučba pomocou programu

Ak nevieme, ako máme postupovať pri vyplňaní sieťovej schémy, môžeme využiť výučbovú funkciu programu. Program nás postupne prevedie

všetkými čiastočnými súčinnmi a vypočíta výsledok. Počas riešenia príkladu program zobrazuje kroky farebne aj slovne. Navyše je možné rýchlosť riešenia upraviť posunutím jazdca na pravej strane hore alebo dole, čím sa zníži, resp. zvýši rýchlosť riešenia. Program zobrazuje riešenie pre činitele, ktoré sme zadali do editačných políčk. Riešenie príkladu programom spustíme kliknutím na tlačidlo „Ukáž riešenie“. Počas riešenia nie je možné do programu zadávať akékoľvek hodnoty ani klikať na tlačidlá až do vyriešenia príkladu.



Obr. 6

Príklad môžeme nechať vypočítať programom aj bez ukážky hľadania riešenia jednoducho kliknutím na tlačidlo „Vypočítaj“.

Samostatné riešenie príkladov

Po vysvetlení príkladu programom môžeme prejsť na vlastné počítanie. Klikneme na tlačidlo „Nový príklad“, zadáme nové činitele alebo necháme program tieto činitele vygenerovať a potvrdíme ich stlačením tlačidla „OK“. Následne vyplníme sieťovú schému a nájdeme výsledok. Po vyriešení príkladu sa o správnosti riešenia dozvieme kliknutím na tlačidlo „Skontroluj“. Program nekontroluje iba samotný výsledok, ale celý postup riešenia, teda aj jednotlivé čiastočné súčiny a súčty uhlopriečkových spojov. Pri používaní programu môže nastať otázka, ako zabezpečiť

to, aby príklad vypočítal žiak a nie samotný program? V prípade, ak je príklad vypočítaný programom automaticky, program o tom upozorní nápisom pod sieťovou schémou: „Riešenie nájdené programom“.

Program navyše meria čas výpočtu. Tento čas sa začína počítať okamihom kliknutia na tlačidlo „OK“ a zastaví sa kliknutím na tlačidlo „Skontroluj“ za podmienky, že výsledok aj postup riešenia sú správne. Meranie času riešenia môžu učitelia využiť na motivovanie žiakov a na aktivovanie súťažného ducha medzi žiakmi. Tiež učiteľ môže zadať príklad, za ktorý dostanú dobrú známku prví traja najrýchlejší riešitelia.

Nie všetkým však tikanie hodín a pripočítavanie času vyhovuje. Čas riešenia sa dá skryť zaškrtnutím políčka „Skry“. Hodiny bežia na pozadí a po vyriešení príkladu si žiak tieto hodiny môže zobrazíť, ale počas samotného riešenia ich má skryté a nerozptyľujú ho.

Doplnenie chýbajúcich číslíc

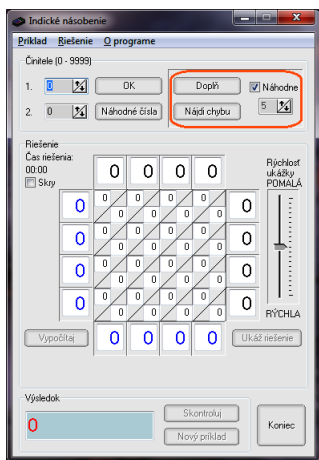
Tento postup práce môžeme uplatniť, keď už žiaci poznajú princíp fungovania algoritmu indického násobenia. Spočíva v tom, že program vygeneruje náhodné činitele, vyplní sieťovú schému, ale niekoľko políčok nechá nevyplnených. Počet týchto políčok je možné nastaviť. Ak necháme zaškrtnuté políčko „Náhodne“, program nevyplní 4-10 políčok schémy. Ak si chceme vybrať fixný počet nevyplnených políčok, odškrtneme políčko „Náhodne“ a do editovacieho políčka pod zaškrťavacím zadáme počet nevyplnených políčok, ktorých počet sa môže pohybovať od 1 do 25. Nakoniec klikneme na tlačidlo „Doplň“ (obr. 7).

Úlohou žiakov bude správne vyplniť chýbajúce políčka. Pritom sa nedjedná iba o políčka čiastočných súčínov, ale chýbať môžu aj číslice samotných činiteľov alebo výsledku. Z tohto dôvodu si program vždy náhodne generuje vlastné činitele. Tým sa dostávame aj k precvičeniu delenia, kedy žiak môže mať k dispozícii iba jedného činiteľa a súčin, prípadne činitele nemusí mať k dispozícii vôbec. Tie si musí dopočítat z ostatných čiastočných súčínov a k chýbajúcim činiteľom sa vrátiť neskôr. Počas vypĺňania chýbajúcich číslíc sa aj tu meria čas výpočtu, pričom program neumožňuje nájsť výsledok automaticky. Pre zadanie nového príkladu je potrebné stlačiť tlačidlo „Nový príklad“, čím sa celá sieťová schéma vymaže a pripraví na nové činitele.

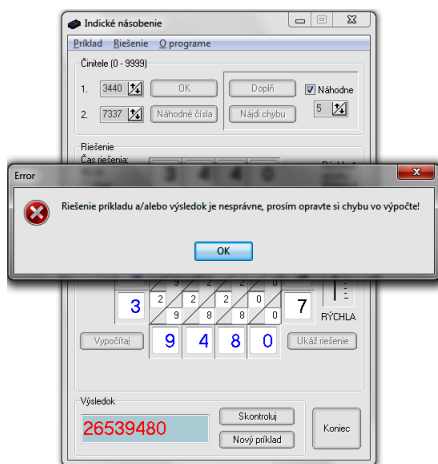
Hľadanie chyby

V tomto prípade program vypočíta príklad, ktorého činitele môžu byť zadané manuálne alebo náhodne vygenerované, ale zároveň zámerne urobí

na viacerých miestach výpočtu chybu (obr. 8). Počet týchto chýb je opäť možné nastaviť rovnakým spôsobom ako v kapitole Doplnenie chýbajúcich číslíc. Úlohou žiakov bude tieto chyby nájsť a opraviť. Na rozdiel od doplnenia chýbajúcich číslíc nemôže byť chyba urobená v samotných činiteľoch. Aj v tomto prípade program meria čas výpočtu, ktorý môže byť použitý na oznámkovanie najrýchlejších žiakov. Žiak sa o správnosti riešenia dozvie kliknutím na tlačidlo „Skontroluj“. Pre zadanie nového príkladu využijeme tlačidlo „Nový príklad“.



Obr. 7



Obr. 8

Ďalšie vlastnosti algoritmu a programu

Ako bolo uvedené, rozsah činiteľov je obmedzený na celé čísla v rozsahu od 0 do 9999. Pri násobení desatinných čísel postupujeme rovnakým spôsobom, len pri výsledku oddelíme príslušný počet desatinných miest. Napríklad pri násobení čísel 36,24 a 18,7 zadáme do programu celé čísla 3624 a 187. Ako výsledok dostaneme 677688. Stačí už len oddeliť tri desatinné miesta a dostaneme výsledok pôvodného príkladu, ktorým je 677,688.

Program je možné ovládať ako tlačidlami, tak aj položkami v menu. Ovládanie programu zvládne žiak základnej školy aj na prvom stupni. Je vhodný aj na domácu prípravu pod dozorom rodiča. Žiak si vie program jednoducho stiahnuť z učiteľského počítača a nakopíruje si ho do svojho

domáceho. Nie je potrebná žiadna inštalácia ani konfigurácia programu. Program nezaberá veľa miesta na disku a nie je náročný na hardvérovú konfiguráciu.

Indické násobenie, jeden z najstarších algoritmov násobenia, bol používaný vlašskými kupcami v 15. storočí a nazývali ho aj „per gelosia“, čiže žalúziový, sieťový alebo tiež indický. Oba činitele i súčin zapisovali vodorovne a zvislo tak, že zápis dostal úhľadnú podobu. Pre jeho podobnosť s klasickým spôsobom násobenia je možné ho uplatniť na prvom a druhom stupni základných škôl. Počítačový program MultInd umožňuje zefektívnenie práce a precvičenie si malej násobilky a sčítania prirodzených čísel. Program je voľne šíriteľný a je možné ho stiahnuť z nasledujúceho linku: http://www.benetomas.hostuju.cz/Indicke_nasobenie.zip

Verím, že žiaci si ho obľúbia a budú ho s radosťou využívať aj v domacom prostredí. Dotazy a podnety k programu je možné zasielať mailom na adresu benetomas@centrum.sk.

Literatúra

- [1] *Bach, E. – Shallit, J.*: Algorithmic Number Theory, Vol. 1: Efficient Algorithms. Cambridge, MA: MIT Press, 1996.
- [2] *Dušek, F.*: Matematické zájmové kroužky, SPN, Praha 1971, 1. vyd.
- [3] *Hejný M. a kol.*: Teória vyučovania matematiky 2, SPN, Bratislava 1990.

ZPRÁVY

Celostátní kolo FO 2013

Uspořádání celostátního kola kategorie A 54. ročníku Fyzikální olympiády se ve školním roce 2012/2013 ujalo *Gymnázium Brno, třída Kapitána Jaroše 14* (<http://www.jaroska.cz>) ve spolupráci s Vysokým učením technickým v Brně a Masarykovou univerzitou. Na základě výsledků krajských kol soutěže, které proběhlo 25. 1. 2013 a jehož se zúčastnilo celkem 103 soutěžících, z nichž 60 se stalo

úspěšnými řešiteli, bylo do Brna pozváno 50 nejlepších.

Slavnostního zahájení v úterý 26. 2. ve večerních hodinách v aule gymnázia se kromě členů Ústřední komise FO a pořadatelů zúčastnilo několik významných osobností města Brna, Jihomoravského kraje a dalších hostů, večer zpestřilo vystoupení členů Úžasného divadla fyziky s experimentální show a malé pohostění.

Ve středu dopoledne 27. 2. soutěžící řešili po dobu pěti hodin čtyři teoretické úlohy. Autorem první z nich s názvem *Roztlačování železničního vozu* byl RNDr. Josef Jirů (Gymnázium Pelhřimov); řešitelé za ni získali v průměru