

Priloge : listingi makro programov za izračun :

- xNaKvadrat
  - KvadratniKoren
  - eNaX
  - l0NaX
  - NaravniLogaritem
  - DekadniLogaritem
- 
- zaslonska slika s primerom nastavitve
  - disketa s programi, tekstom in sliko

Pozdravljeni !

Kar nekaj časa je že preteklo odkar sem napisal makro programe za izračunavanje trigonometričnih funkcij. Že takrat sem razmišljal o programih, ki bi izračunavali tudi druge pogosteje uporabljane funkcije. Lotil sem se dela in jih "spesnil".

Programi, ki vam jih tokrat pošiljam, izračunavajo naravni in dekadni logaritem, kvadrat in kvadratni koren, ter eksponentni funkciji  $e^x$  in  $10^x$  izbranega števila. Izračunavanje potenčnih in eksponentnih funkcij v Wordu na srečo ni poseben problem, saj pri delu s tabelami lahko za računanje potenc in korenov uporabimo operator " $^$ ". Lahko bi izkoristil to možnost, kljub temu pa sem tudi te funkcije poskusil izračunati z uporabo osnovnih operatorjev (+, -, \*, /, MOD), ki jih pozna WordBasic. Z nekaj nenavadnimi prijemi mi je, tako vsaj upam, to tudi uspelo. Malo telovadbe pa možganom tudi ne škoduje !

V predlogo "Normal.dot" sem poleg gumbov za trigonometrijske funkcije vdelal dodatne funkcijske gumbe in dodal meni **Kalkulator**, v katerem sem zbral vse matematične makro programe. Način dela je že znan: natipkamo število, za katerega iščemo funkcijo, število z levim gumbom miške pobarvamo ter pritisnemo na gumb željene funkcije, ali pa jo sprožimo v meniju **Kalkulator**.

Če prav razumem, urednik te rubrike avtorjem makro programov predlaga naj njihovo delovanje na kratko razložijo, zato bom poskusil narediti tudi to. Bralcem s slabšimi živci pa kljub temu priporočam, da ta del preskočijo. Kot se spodobi, pojdimo po vrsti, od enostavnejših k zahtevnejšim programom.

## Program "X<sup>2</sup>"

Izbrano število program zapiše v spremenljivko "a" in jo najprej testira. Če je a = 0 ali 1, je izračun tako rekoč že končan, ker je  $0^2 = 0$  in  $1^2 = 1$ . Rezultat testiranja program vpiše v spremenljivko "R\$" in ga s izpiše podprogramom IZPIS.

Če vhodni podatek ni enak 0 ali 1, program zmnoži a\*a, rezultat vpiše v "R\$" nato pa ga v podprogramu ZAOKROZEVANJE zaokroži na N decimalk ter ga izpiše.

## Program "Kvadratni koren"

Izbrano število program zapiše v spremenljivko "a" in "z". Vhodni podatek nato testiramo. Če je negativen, moramo upoštevati da je  $\sqrt{-1} = i$ , zato spremenljivka "a" spremeni predznak, v spremenljivko "T" pa vpišemo indeks 1. Ker je  $\sqrt{1} = 1$  in  $\sqrt{0} = 0$  lahko z nadaljnim testiranjem program včasih kar zaključimo.

Če se vhodni podatek razlikuje od 0 ali 1, izračunamo njegov kvadratni koren po Newtonovi iteracijski metodi iskanja približkov. Uvedemo spremenljivko "a0" in vanjo vpišemo a+1, nato pa izračunamo 1. približek kvadratnega korena:  $a_1 = (a_0 + a/a_0) / 2$ . Z zanko nato iščemo približke, dokler naslednji približek ni enak prejšnjemu. Rezultat se po določanju števila decimalnih mest izpiše. Če smo iskali kvadratni koren negativnega števila je T=1, program pa k rešitvi pripiše še imaginarno število "i".

## Programa "e<sup>x</sup> in 10<sup>x</sup>"

Oba programa računata po isti metodi, program za 10<sup>x</sup> je v bistvu le nekoliko prirejen program za izračun e<sup>x</sup>. "e" je Neperjevo število, ki je osnova naravnih logaritmov (  $e \approx 2.71828\dots$  ), osnova dekadnih pa je seveda število 10.

Za lažje razumevanje delovanja programov si naprej oglejmo nekaj matematičnih pravil. Za funkcijo e<sup>x</sup> velja da je :

$$e^{4.5} = e^4 \times e^{0.5} = e \times e \times e \times e \times e^{0.5}$$

$$e^{-4.5} = 1 / e^{4.5} = 1 / (e \times e \times e \times e \times e^{0.5})$$

Oglejmo si kako ti pravili iskoristimo v makro programu za izračun e<sup>x</sup> :

Vhodni podatek, to je eksponent funkcije, se vpiše v spremenljivko "P", njegova absolutna vrednost pa v "a". Če je eksponent prevelik, je računanja takoj konec in sledi izpis opozorila. Če je eksponent v dopustnih mejah, ga program razstavi na celi in decimalni del, ju vpiše v "x" in "k", ter izračuna dva vmesna rezultata. Izračun e<sup>x</sup> za celi del eksponenta je preprost, e<sup>x</sup> -krat množimo samega s seboj in dobimo "rezultat1". Če je eksponent samo

celo število (  $k=0$  ), je računanje skoraj končano in program se nadaljuje v podprogramu ULOMEK. Če pa je del eksponenta decimalno število, program izračuna še drugi vmesni rezultat. Računanje tega rezultata je v bistvu seštevanje členov funkcije  $e^x$  razvite v vrsto (Matematični priročnik, Bronštejn-Semendjajev, izdaja 1967, Str.378). Če pri računanju členov vrste pride do prekoračitve, se računanje konča. Sledi množenje obeh vmesnih rezultatov, nato pa za negativne eksponente podprogram ULOMEK izračuna nov rezultat, tako da 1 deli s prejšnjim rezultatom. Podprogram ZAOKROZEVANJE končni rezultat zaokroži na N decimalk ter ga pošlje v IZPIS.

Programa za izračun  $10^x$  deluje povsem enako, le da upošteva naslednji pravili :

$$10^{3.51} = 10^3 \times 10^{0.51} = 10 \times 10 \times 10 \times e^{(0.51 \times \ln 10)}$$

$$10^{-3.51} = 1 / 10^{3.51} = 1 / (10 \times 10 \times 10 \times e^{(0.51 \times \ln 10)})$$

Izračun  $10^x$  za celi del eksponenta je preprost, osnovo  $10^x$  -krat množimo samo s seboj, decimalni del eksponenta pa najprej pomnožimo z  $\ln 10$  ( $\ln 10 = 2.30258\dots$ ), nato seštejemo člene vrste, zmnožimo oba vmesna rezultata itd.

### **Programa “Naravni in dekadni logaritem”**

Tudi ta dva programa sta si zelo podobna. Program za izračunavanje naravnega logaritma je uporabljen tudi za izračun dekadnega.

Spet si naprej oglejmo eno od matematičnih lastnosti logaritmov. Za funkcijo  $\ln X$  velja da je npr. :

$$\ln 45 = 2 \times 2 \times 2 \times \ln \left( \sqrt{\sqrt{\sqrt{45}}} \right)$$

To pravilo je zelo pomembno, ker za računanje logaritmov oba programa uporabljata logaritemsko funkcijo razvito v vrsto (Matematični priročnik, Str.378), po kateri lahko računamo naravne logaritme za števila do 2. Žal se ta vrsta zelo “grdo vede” za zelo majhna števila in za števila, ki so blizu 2. Vrednost nekaterih členov vrste namreč kaj hitro preseže največje ali najmanjše število, ki ga WordBasic še lahko obdela. Vhodni podatek zato s korenjenjem najprej primerno pripravimo, nato pa obdelamo. V programu za izračun naravnega logaritma števila poteka to takole:

Program najprej oceni smiselnost vhodnega podatka in določi po katerem od treh načinov bo izračunal logaritem.

Če je vhodni podatek večji od 1 in manjši od 1.9 ga takoj obdela v podprogramu LN, kjer seštevava člene vrste, rezultat nato zaokroži na zahtevano število decimalnih mest in ga izpiše.

Če je vhodni podatek večji od 1.9 ga program pošlje v podprogram VELIKI. Ta ga s pošiljanjem v podprogram SQRT tolikokrat koreni, da je rezultat korenjenja manjši od 1.9. Z indeksom "k" hkrati šteje kolikokrat je potekalo korenjenje. Podprogram LN nato poišče logaritem rezultata korenjenja, ki ga potem k - krat pomnožimo z 2. Rezultat te procedure je naravni logaritem vhodnega podatka.

Če je vhodni podatek manjši od 1 ga program pošlje v podprogram MALI. Ta najprej ugotovi ali je podatek zapisan z eksponentnim zapisom, npr. 12e-3, in če je, ga koreni tolikokrat, da je rezultat korenjenja zapisan decimalno. Korenjenje se nato nadaljuje dokler ni na mestu za decimalno piko številka večja od 0, nato pa podprogram LN končno izračuna logaritem rezultata korenjenja. Sledi k - kratno množenje z 2, zaokroževanje in izpis.

Program za izračun dekadnega logaritma deluje povsem enako, rezultat moramo le še nekoliko popraviti. Velja namreč da je :  $\log X = \ln X / \ln 10$ . Zato program v bistvu izračuna naravni logaritem vhodnega podatka, nato pa ga pred zaokroževanjem še deli z 2.3025...

Ker sta oba programa nekoliko počasna, lahko med izvajanjem makro programa za izračunavanje naravnega in dekadnega logaritma v statusni vrstici zdolgočaseno opazujemo koliko korakov bo program še izvedel.

Vsi programi imajo vgrajeno proceduro za zaokroževanje izpisa, ki je napisana nekoliko drugače kot v programih za izračunavanje trigonometričnih funkcij. Upam da bolje!

S spreminjanjem N lahko spremenimo število decimalnih mest. Če za N vstavimo 0, se izpiše le celi del rezultata. Kot decimalni znak uporabljajo programi piko in ne vejico. Nastavitev pike kot decimalnega znaka mora biti usklajena s področnimi nastavitvami Windows.

V programe so vdelani trije različni načini izpisa. Izbiramo jih z ukazom REM, ki ga vključimo pri tistih dveh izpisih, ki ju ne bomo uporabili. Pri 1. načinu izpisa programi pred številko vpišejo funkcijo, za njo pa rezultat, pri 2. načinu programi za številko vpišejo rezultat, za njim pa funkcijo v oklepaju, pri 3. načinu vstavijo programi med številko in rezultat le tabulator. Ta način izpisa je dokaj uporaben, kadar sestavljamo tabelo.

Lepo pozdravljeni !

Marjan Klenovšek, dipl.ing.