Dokumentácia k projektu na predmety IZP a IUS

Iteračné výpočty

projekt č.2

29. decembra 2013

Autor: Marek Marušic, xmarus05@stud.fit.vutbr.cz
Fakulta Informačných Tecnologií
Vysoké Učení Technické v Brne

1 Úvod

Počítanie uhlov trojuholníka pomocou matematických operácií +, -, *, / je z pohľadu algoritmizácie veľmi zaujímavý problém. Samozrejme pomocou kalkulačky vypočítame uhly trojuholníka behom pár sekúnd, no dnes tu nie sme kvôli tomu aby sme ukazovali ako krásne vieme počítať na kalkulačke.

Tento dokument popisuje návrh a implementáciu aplikácie pre iteračné výpočty. Navrhnutý program funguje ako konzolová aplikácia, ktorá z argumentov programu načíta súradnice bodov trojuholníka, resp. vstupné čísla na odmocnenie, alebo získanie arkus sínusu a následne na štandardný výstup vypíše získaný výsledok.

2 Analýza problému a princíp jeho riešenia

V tejto kapitole sa pozrieme podrobnejšie na počítanie uhlov trojuholníka a niekoľko potrebných funkcií pre tento výpočet. Ďalej sa v tejto kapitole zameriame na rôzne možnosti, ktoré by sa dali pri výpočte použiť. Samozrejme musíme dodržať, že počítanie robíme len pomocou základných matematických funkcií.

2.1 Zadanie problému

Cieľom tohto projektu je vytvorenie programu v jazyku C, ktorý vypočíta matematické funkcie arkus sínus, druhú odmocninu a veľkosti uhlov trojuholníka. Program počíta s presnosťou na 11 platných číslic. Program musí načítať parametre a čísla z argumentov programu. Výsledok bude vypísaný na štandardný výstup. Je zakázané používať matematické funkcie a konštanty z hlavičkového súboru „math.h“. Tieto matematické funkcie majú svojim chovaním čo najlepšie napodobňovať obdobné matematické funkcie zo štandardnej knižnice jazyka C - „math.h“.

2.2 Presnosť výpočtu

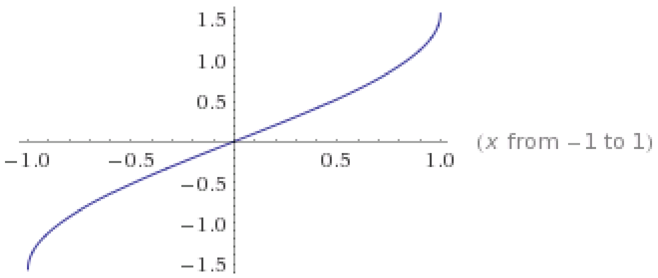
Presnosť výpočtu je zadaná na 11 platných číslic. Najskôr si treba uvedomiť čo je to platná číslica. Prvá platná číslica je prvá nenulová číslica zľava. Počet platných číslic sa počíta od tejto prvej platnej číslice. Kedže rátame na 11 platných číslic, znamená to, že nás zaujíma prvých 11 nenulových číslic zľava.

2.3 Konvergencia funkcií

Konvergentná rada je nekonečná rada čiastočný súčtov, ktorá má konečnú limitu. Pokiaľ sa chceme k výsledku čo najrýchlejšie a najpresnejšie priblížiť, je treba nájsť interval, v ktorom funkcia najlepšie konverguje. Ak by sme počítali v iných intervaloch bol by výpočet oveľa dlhší a náročnejší.

2.4 Arkus kosínus

Arkus sínus je cyklometrická funkcia. Je to inverzná funkcia goniometrickej funkcie sínus. Táto funkcia je nepárna a ostro rastúca na celom svojom definičnom obore.



Obrázok 1: Graf funkcie arkus sínus

Definičný obor funkcie je interval <-1,1> a obor hodnôt je $\left\langle -\frac{\pi}{2}, \frac{\pi}{2} \right\rangle$. Pre výpočet hodnoty arkus sínus je skoro nevyhnutné použitie nekonečnej rady. Výpočet konverguje pokým nieje dosiahnutá požadovaná presnosť (presnosť na 11 platných číslic). Zvyčajne sa pri takomto zadaní používa Taylorov polynom [1].:

$$\begin{aligned}\arcsin(x) &= x + \frac{1}{2} \cdot \frac{x^3}{3} + \frac{1 \cdot 3}{2 \cdot 4} \cdot \frac{x^5}{5} + \frac{1 \cdot 3 \cdot 5}{2 \cdot 4 \cdot 6} \cdot \frac{x^7}{7} + \cdots \\ &= \sum_{n=0}^{\infty} \left(\frac{(2n)!}{2^{2n}(n!)^2} \cdot \frac{x^{2n+1}}{2n+1} \right), \quad |z| < 1\end{aligned}$$

2.5 Druhá odmocnina

Druhá odmocnina je špeciálnym typom obecnej odmocniny. Odmocňovanie je v matematike inverzná operácia umocňovania. Odmocnina sa dá vypočítať pomocou Newtonovej metódy:

$$\sqrt{x} = y_n, \quad y_0 = 1, \quad y_{i+1} = \frac{1}{2}\left(\frac{x}{y_i} + y_i\right)$$

Počítame ju s presnosťou na 11 platných číslic. V obore reálnych čísel odmocnina zo záporných čísel neexistuje.

2.6 Uhly trojuholníka

Trojuholník je jeden zo základných rovinných geometrických útvarov s troma vrcholmi, stranami a vnútornými uhlami. Je to dvojrozmerný útvar. Súčet vnútorných uhlov v trojuholníku je 180°. Pri zadaných súradniciach bodov alebo dĺžok strán sa dá veľkosť uhlov vypočítať pomocou goniometrických funkcií alebo pomocou kosínusovej vety.

3 Návrh riešenia problému

Pri návrhu riešenia budem vychádzať z predchádzajúcej kapitoly 2. V prvom rade treba spracovať spúšťacie argumenty programu z príkazového riadku, podľa ktorých bude program ďalej pracovať. Ďalej podľa argumentov budeme tlačiť na štandardný výstup nápovedu, odmocninu zo zadaného čísla, arkus sínus zo zadaného čísla alebo uhly trojuholníka zo zadaných súradníc bodov trojuholníka.

3.1 Výpočet druhej odmocniny

Pre výpočet druhej odmocniny zo zadaného čísla je možné použiť rôzne metódy a postupy. Ja som po preštudovaní a analýze zvolil Newtonovu metodu.Vybral som ju pretože v zadaní máme určenú práve túto metódu, aby sme sa lepšie naučili pracovať v jazyku C. Voľba algoritmu bola dosť predvídateľná.

Začal som cyklom, ktorý sa opakoval, pokým prírastok, ktorý sme pripočítavali v každej iterácií k číslu, nebol menší ako daná presnosť na 11 platných číslic. V každej iterácii som použil vzorec Newtonovej metódy. Pre výpočet odmocniny som použil typ *double*. Pokiaľ je číslo na vstupe záporné vraciam hodnotu nan (Not a Number), pretože ako sme si spomenuli v predošlej kapitole v obore reálnych čísel neexistuje odmocnina zo záporných čísel.

3.2 Arkus sínus

Pre výpočet arkus sínus som v kapitole 2.4 navrhol jeden spôsob riešenia výpočtu pomocou Taylorovho polynomu. Tento spôsob nie je príliš vhodný pre hodnoty v intervaloch <0.9,1> a <-1,-0.9>, pretože je konvergencia zlá a neúnosná pre svižný beh programu. Preto som sa rozhodol pátrať ďalej a našiel som vzorec, v ktorom získavam arkus sínus pomocou rekurzívneho použitia arkus sínus [2]:

$$\begin{cases} -\frac{1}{2}\pi + \sin^{-1}\left(\sqrt{1-x^2}\right) & \text{for } x < 0 \\ \frac{1}{2}\pi - \sin^{-1}\left(\sqrt{1-x^2}\right) & \text{for } x > 0 \end{cases}$$

Tento vzorec som použil len pre hodnoty, ktoré horšie konvergujú, pre ostatné hodnoty od 0 až po 0.9 konverguje Taylorov polynom veľmi dobre. Samozrejme, ako vidíme je treba použiť pre výpočet typ double.

3.3 Výpočet uhlov trojuholníka

Zo zadaných súradníc bodov treba najskôr vypočítať dĺžky strán trojuholníka, následne som sa rozhodol použiť pre ďalšie výpočty uhlov kosínusovú vetu pre ktorú platia vzorce:

- $a^2 = b^2 + c^2 - 2 \times b \times c \times \cos\alpha$
- $b^2 = a^2 + c^2 - 2 \times a \times c \times \cos\beta$
- $c^2 = a^2 + b^2 - 2 \times a \times b \times \cos\gamma$

Zo zadaných vzorcov získame kosínus uhlov, z ktorého budeme potrebovať ďalej zistiť uhol pomocou inverznej funkcie kosínusu. Keďže sme si už naprogramovali inverznú funkciu arcus sínus, nebudeme inverznú funkciu kosínusu zbytočne programovať, ale použijeme vzorec pre vzťah arcus sínus a arcus kosínus [3]:

$$\arccos x = \frac{\pi}{2} - \arcsin x$$

Týmto postupom sme prišli ku veľkosti vnútorných uhlov trojuholníka. Tieto veľkosti ukladám do premenných typu double, pretože nám záleží na desatinných miestach.

3.4 Presnosť výpočtov

Väčšina našich výpočtov a výsledkov je iracionálnych alebo ich desatinný rozvoj je veľmi dlhý, preto nám bolo zadané, že máme počítať na presnosť 11 platných číslic. K takejto presnosti sa dokážeme dopracovať pomocou relatívnej alebo absolútnej presnosti. Pri oboch sa používa ε čo je premenná, do ktorej uložíme číslo a ktorá sa používa v podmienke cyklu, kde ak narazíme na prípad, že prírastok je menší ako ε, tak sa cyklus ukončí. Vo vzorci to vypadá nasledovne:

absolútna presnosť = |Yi-1 – Yi| >= ε

relatívna presnosť = |Yi-1 – Yi| >= ε * |Yi|

4 Špecifikácia testov

Pri testovaní programu sa treba zamerať okrem bežných vstupných hodnôt aj na hodnoty krajné, špeciálne hodnoty a rizikové čo vyplýva zo zadania. Ako napríklad už spomínané hodnoty pre arcus kosínus alebo záporné čísla pri výpočte odmocniny. Všetky testy sú skúšané na operačnom systéme Linux.

Test 1: Druhá odmocnina z čísla 3.
Spúšťacie parametre: ./proj2 --sqrt 3
Výstup: 1.7320508076e+00

Test 2: Záporná hodnota pre druhú odmocninu -> hodnota mimo definičný obor
Spúšťacie parametre: ./proj2 --sqrt -3
Výstup: nan

Test 3: Nesprávne zadané číslo pre druhú odmocninu
Spúšťacie parametre: ./proj2 --sqrt 3a
Výstup: „zadane parametre nie su platne cisla“

Test 4: Nesprávny spúšťací argument
Spúšťacie parametre: ./proj2 –sqrtt 3a
Výstup: „Chybné parametre príkazového riadku!“

Test 5: Správne spustenie arkus sínus
Spúšťacie parametre: ./proj2 --asin 0
Výstup: 0.0000000000e+00

Test 6: Hodna, ktorá nieje z def. oboru arkus sínus
Spúšťacie parametre: ./proj2 --asin 2
Výstup: nan

Test 7: Krajná hodnota arkus sínus
Spúšťacie parametre: ./proj2 --asin 1
Výstup: 1.5707963268e+00

Test 8: Zlý spúšťací argument arkus sínus
Spúšťacie parametre: ./proj2 --assin 0
Výstup: „Chybné parametre príkazového riadku!“

Test 9: Správne spustenie trojuholníka
Spúšťacie parametre: ./proj2 --triangle 0 0 1 0 0 2
Výstup: 1.5707963268e+00
1.1071487178e+00
4.6364760900e-01

Test 10: Zadané vrcholy netvoria trojuholník
Spúšťacie parametre: ./proj2 --triangle 0 0 0 0 0 0
Výstup: „Zadane parametre netvoria trojuholnik."

Test 11: Nesprávne zadané vrcholy trojuholník
Spúšťacie parametre: ./proj2 --triangle 0 0 1 0 0 2a
Výstup: „zadane parametre nie su platne cisla"

Test 12: Nesprávne zadaný argumen
Spúšťacie parametre: ./proj2 -triangle 0 0 1 0 0 2
Výstup: „Chybné parametre príkazového riadku!"

5 Popis vlastného riešenia

Pri implementácii a riešení som postupoval podla problematiky a postupov popísaných v predchádzajúcich kapitolách.

5.1 Ovládanie programu

Program je ovládaný pomocou príkazového riadku. Správanie programu sa odvíja od zadaných spúšťacích argumentov, ktorých popis sa nachádza v nasledujúcej tabuľke:

--help	Vypíše nápovedu k programu
--sqrt x	Program vypíše na štandardný výstup druhú odmocninu čísla x
--asin x	Program vypíše na štandardný výstup arkus sínus čísla x
--triangle AX AY BX BY CX CY	Program vypíše na štandardný výstup uhly trojuholníka, ktorého súradnice vrcholov sú: AX AY BX BY CX CY.

5.2 Voľba dátových typov

Niektoré dátové typy boli spomenuté v predchádzajúcich kapitolách. Na výpočtové funkcie som použil dátový typ double, pretože sme pracovali s desatinnými číslami, pri ktorých bola vyžadovaná vysoká presnosť.

Pri implementácií chybových stavov som použil na prenos informácií o chybách dátový typ int. Následne som použil typ ukazovateľ na typ char pre chybové hlášky, kvôli jednoduchšiemu spracovaniu.

5.3 Vlastná implementácia

Argumenty príkazového riadku sú spracované pomocou funkcie *spustac,* ktorá následne zavolá funkcie na výpočet a výpis zadaných matematických funkcií, alebo funkciu pre vytlačenie nápovedy s názvom *PrintHelp*.

Na prevod z reťazcov na čísla používam funkciu *strtod,* ktorá vracia chybu ak čísla na vstupe nie sú v správnom formáte, alebo nie sú platné čísla. Ak sme dostali správne čísla, volajú sa výpočtové funkcie *my_sqrt,* m*y_asin* alebo *VelkostUhla* podla zadaných parametrov v príkazovom riadku. Po výpočte sa vytlačí výsledok na štandardný výstup a funkcia *spustac* následne vracia chybovú hodnotu typu *int* s číselnou hodnotou chyby, podľa toho aké chyby nastali behom chodu programu.

Po spracovaní a vypočítaní sa následne vypíšu výsledky, alebo po vrátení chýb sa na výstup vypíše chybová hláška zodpovedajúca typu návratovej hodnoty chyby z funkcie *spustac*.

6 Záver

Program je navrhnutý na výpočet arkus sínus, druhej odmocniny a uhlov trojuholníka. Testované výpočty som porovnával s výsledkami rovnakých matematických funkcií z hlavičkového súboru math.h a výsledné hodnoty boli totožné. Pri výpočte arkus sínus bolo treba použiť malé vylepšenie pre interval <0.9,1>, pretože program nerátal presne a výpočet bol náročnejší.

Program vyhovuje požiadavkám zo zadania projektu a bol bezchybne otestovaný na operačných systémoch Linux a Windows.

Referencie

[1] BARTSCH, H.-J.: Matematické vzorce. Praha: Mladá fronta, 3.vydanie, 1996, 831 s., ISBN 80-204-0607-7

[2] WOLFRAM ALPHA LLC—A WOLFRAM RESEARCH COMPANY. Wolfram|Alpha: Computational Knowledge Engine[online]. © 2013 [cit. 2013-12-3]. Dostupné z: http://mathworld.wolfram.com/InverseSine.html

[3] KORNIŁOWICZ, Artur; SHIDAMA, Yasunari. Inverse trigonometric functions arcsin and arccos. *Formalized Mathematics*, 2005, 13.1: 73-79.

Príloha A

Metriky kódu

Počet súborov: 1 súbor

Počet funkcií: 11

Počet riadkov zdrojového textu: 218 riadkov

Veľkosť statických dát: 4 879 bajtov

Veľkosť spustiteľného súboru: 13 291B (systém Linux, 64 bitová architektúra)