

Dokumentácia k projektu pre predmety IZP a IUS

Iterační výpočty

projekt č. 2

4.12.2013

Autor: Filip Ježovica, xjezovica01@stud.fit.vutbr.cz

Fakulta Informačních Technologii

Vysoké Učení Technické v Brně

Obsah

1	Úvod.....	2
2	Analýza problému a princíp jeho riešení.....	2
2.1	Zadanie problému.....	2
2.2	Druhá odmocnina.....	2
2.3	Arcus sínus.....	3
2.4	Uhly trojuholníka.....	4
3	Návrh riešenia problému	5
3.1	Funkcia druhá odmocnina	5
3.2	Arcus sínus.....	5
3.3	Uhly trojuholníka.....	6
3.4	Presnosť výpočtu.....	7
4	Špecifikácia testov.....	8
5	Popis riešenia.....	9
5.1	Ovládanie programu	9
5.2	Voľba dátových typov	10
5.3	Vlastná implementácia.....	10
6	Záver.....	10
A	Metriky kódu	11

1 Úvod

Táto dokumentácia obsahuje popis návrhu a implementácie programu, ktorý slúži na výpočet matematických funkcií. Program pracuje pomocou iteračných výpočtov, ktoré spracovávajú číselné dáta zadané užívateľom. Program tieto dáta načíta a následne počíta druhú odmocninu, arcus sínus zo zadaného čísla alebo veľkosť vnútorných uhlov trojuholníka. Výsledok danej funkcie programu sa vypíše na obrazovku.

Dokument sa skladá z viacerých častí. V **kapitole 2** sa venujem analýze problémov spojených s definičnými obormy hodnôt jednotlivých funkcií, ktoré sú znázornené grafy týchto funkcií.

Kapitola 3 sa zaoberá riešením problémov iteračných algoritmov a problémov spojených s presnosťou výpočtov. Ďalej sú tu zahrnuté i testy výpočtov pre hraničné, štandardné i chybné vstupy. Testy sú vypísané v **kapitole 4**. **Kapitola 5** je venovaná vlastnej implementácii a spúšťaní programu.

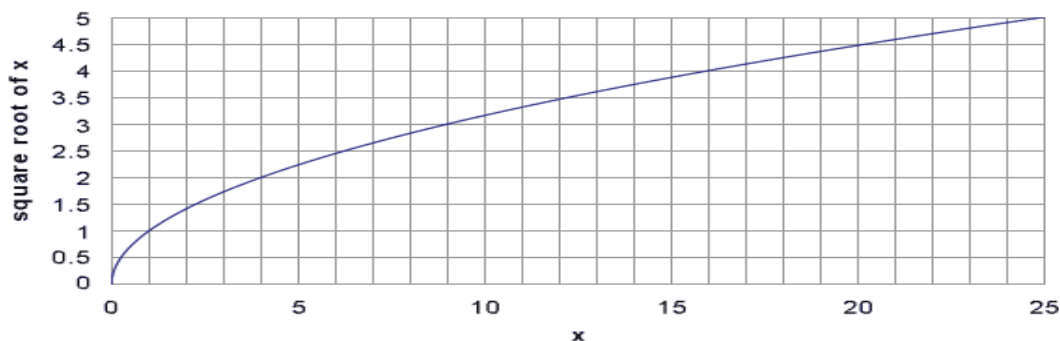
2 Analýza problému a princíp jeho riešenia

2.1 Zadanie problému

Bolo požadované vytvorenie programu v jazyku C, podľa normy C99, ktorý bude počítat vnútorné uhly trojuholníka so zadanou presnosťou. Program musí načítať vstupné údaje zadané užívateľom pomocou argumentov pri spúšťaní programu, ktorými budú súradnice vrcholov trojuholníka v určenom poradí. Uhly sa vypíšu štandardným výstupom na obrazovku (každý na jeden riadok). Ďalej bolo požadované, aby program vypočítal druhú odmocninu z čísla x alebo arcus sínus z čísla x , ktoré je taktiež zadané používateľom. Výsledky budú vypísané štandardným výstupom na obrazovku. Matematické funkcie na výpočet absolútnej hodnoty, druhej odmocniny a arcus sínusu je potrebné implementovať vlastnými funkciami. Zakázaná knižnica `math.h`.

2.2 Druhá odmocnina

Druhá odmocnina je špeciálnym prípadom odmocniny všeobecnej a preto sa často používa názov iba "odmocnina". Zapisuje sa ako $\sqrt{\quad}$.



Obrázok 1: Graf funkcie druhej odmocniny z čísla x

$$x * x = x^2 \Rightarrow \sqrt{x^2} = x \quad (1)$$

Druhá odmocnina hľadá také číslo, ktoré ak umocníme na druhú vznikne nám pôvodné číslo. Ak umocníme záporné číslo na druhú, výsledkom bude kladné číslo. Podľa toho už vieme, že druhá odmocnina zo záporného čísla neexistuje. Výpočet druhej odmocniny implementujem pomocou Newtonovej metódy - vid' **Kapitola 3.1**.

2.3 Arcus sínus

Je to ^{*1}cyklometrická funkcia. Prvým krokom pri tvorení funkcie bude určenie definičného oboru hodnôt. Časť funkcie sínus je prostá a je možné k nej definovať funkciu inverznú. Arcus sínus je inverznou funkciou k funkcii sínus na intervale $[-\frac{\pi}{2}, \frac{\pi}{2}]$ - vid' **obrázok 2**.

Na **obrázku 2** môžeme pozorovať výmenu definičného oboru s oborom hodnôt.

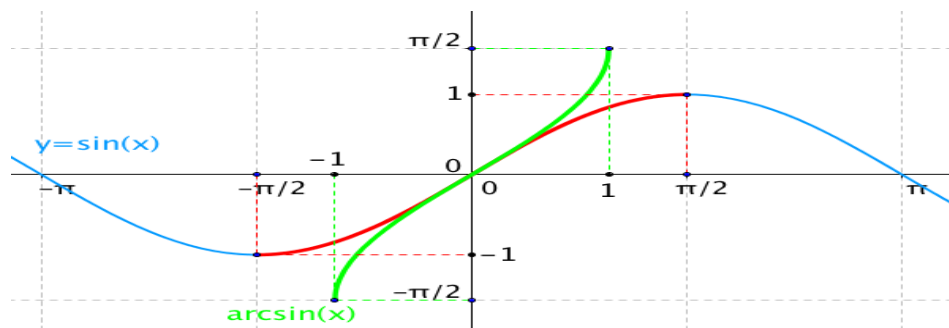
Definičné obory a obory hodnôt oboch funkcií: sínus(2), arcus sínus(3):

$$f = \{[x, y] \mid y = \sin x, x \in \mathbb{R}, y \in [-1, 1]\} \quad (2)$$

$$f = \{[x, y] \mid y = \arcsin x, x \in [-1, 1], y \in [-\frac{\pi}{2}, \frac{\pi}{2}]\} \quad (3)$$

^{*1} cyklometrické funkcie: sú to inverzné funkcie ku goniometrickým funkciám. Pretože goniometrické funkcie sú prosté len na určitých intervaloch, existujú inverzné funkcie iba na týchto intervaloch.

^[1](Bartsch, 1996)

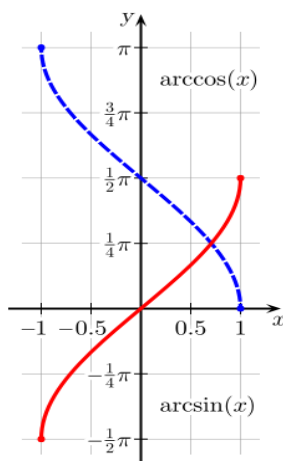


Obrázok 2: Graf funkcie sínus a arcus sínus. Sínus znázornený modrou(časť keď je funkcia prostá červenou) a arcus sínus zelenou.

2.4 Uhly trojuholníka

Výpočet uhlov trojuholníka bude realizovaný pomocou pytagorovej a kosínusovej vety. Pre výpočet arcus kosínusu je daný vzťah(4). Známy je i vzťah prevodu arcus sínusu s arcus kosínusom - vid' **obrázok 3** , vid' vzťah (9). Potom bude možné využiť funkciu na výpočet arcus sínusu - vid' **kapitola 3.2**. Z týchto vzťahov je zrejmé, že ďalej bude potrebné využiť i druhú odmocninu - vid' **kapitola 3.1**. Výsledkami budú veľkosti uhlov trojuholníka v radiánoch. Pri výpočtoch bude potrebné poznať dĺžky jednotlivých strán. Viac v riešení daného problému v **kapitole 3.3**.

$$\arccos x = \frac{\pi}{2} - x - \frac{1}{2} \cdot \frac{x^3}{3} - \frac{1 \cdot 3}{2 \cdot 4} \cdot \frac{x^5}{5} - \frac{1 \cdot 3 \cdot 5}{2 \cdot 4 \cdot 6} \cdot \frac{x^7}{7} + \dots \quad (4)$$



Obrázok 3: graf znázorňuje arcus sínus(červenou) a arcus cosínus(modoru)

3 Návrh riešenia problému

Po analýze zadaného problému, som začal riešiť problém iteračných algoritmov a jednotlivých častí implementácie.

3.1 Funkcia druhá odmocnina

Problém druhej odmocniny budem riešiť Newtonovou metódou. Je to iteračný výpočet, ktorý má nekonečný počet iterácií. Preto je vhodné v určitý okamžik zastaviť iterácie. Podľa zadania je určená presnosť výpočtu na 11 platných číslíc. O tomto probléme viac v **kapitole 3.4**. Pri implementácii budem postupovať podľa rekurentného vzorca(5).

$$Y_i + 1 = \frac{1}{2} \left(\frac{x}{Y_i} + Y_i \right) \quad (5)$$

Funkcia druhej odmocniny musí akceptovať určité prípady, ktoré môžu nastať. V tom prípade má vypisovať výsledky bez nutnosti výpočtu (optimalizácia):

- $x < 0$ - výsledok je NAN
- $x = 0$ - výsledok je 0

3.2 Arcus sínus

Problém arcus sínusu riešim pomocou Taylorovej rady pre funkciu arcus sínus(6).

$$\arcsin x = x + \frac{1}{2} \cdot \frac{x^3}{3} + \frac{1 \cdot 3}{2 \cdot 4} \cdot \frac{x^5}{5} + \frac{1 \cdot 3 \cdot 5}{2 \cdot 4 \cdot 6} \cdot \frac{x^7}{7} + \dots \quad (6)$$

Tento vzorec som si upravil na tvar pre výpočet vždy nasledujúceho člena. Výpočet arcus sínusu je taktiež iteračným výpočtom a je potrebné zvoliť vhodnú ukončovaciu podmienku - podľa požadovanej presnosti (**Kapitola 3.4**) pri zachovaní presných výsledkov. Kritickými hodnotami sú hodnoty v intervale $< 0.9, 1 >$ a taktiež záporné hodnoty $< -1, -0.9 >$. Pre presný a rýchly výpočet v týchto intervaloch som upravil vzorec ďalšieho člena postupnosti. Zároveň som použil volanie funkcie arcus sínus pre upravený vzorec(7)

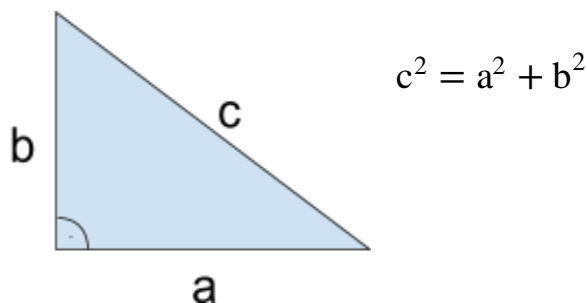
$$\arcsin x = \frac{1}{2} \pi - \arcsin(\sqrt{1 - x^2}) \quad (7)$$

Výpočet arcus sínusu musí akceptovať určité prípady, ktoré môžu nastať a má vypisovať výsledky bez nutnosti výpočtu (optimalizácia):

- $x = 1$ - výsledok je $\pi/2$
- $x = -1$ - výsledok je $-\pi/2$
- $x = 0$ - výsledok je 0

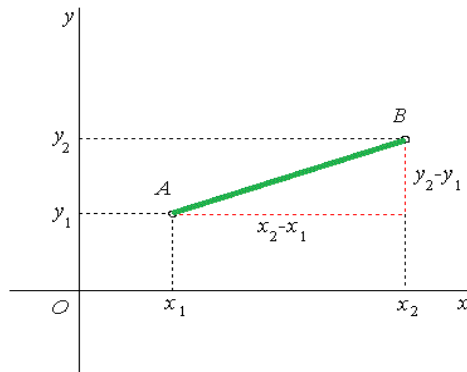
3.3 Uhly trojuholníka

Pri riešení problému výpočtu uhlov trojuholníka, je potrebné riešiť i problém počítania vzdialenosti dvoch bodov v súradnicovom systéme (vektor). Pre vektor využijem pytagorovu vetu - **obrázok 4**. Vektor je určený vzťahom (8) a znázornený na **obrázku 5**.



Obrázok 4: Pytagorova veta, vzťah pre výpočet strany

$$|AB| = \sqrt{[(x_2 - x_1)^2 + (y_2 - y_1)^2]} \quad (8)$$



Obrázok 5: Znáznornenie vzdialenosti dvoch bodov

Pre výpočet arcus kosínusu som sa rozhodol použiť už riešenú funkciu arcus sínus(**Kapitola 3.2**) pre zefektívnenie práce. Ďalej využijem vzťah medzi arcus kosínom a arcus sínusom(9).

$$\arccos x = \frac{1}{2} \pi - \arcsin x \quad (|x| \leq 1) \quad (9)$$

3.4 Presnosť výpočtu

Iteračné výpočty sú nekonečnou postupnosťou. Je to rad čísel, ktoré sa postupne pripočítavajú. Takýmto postupným pripočítavaním sa blížíme k správne výsledku. Ak by sme takýto algoritmus neukončili, bol by to nekonečný cyklus. Preto je potrebné, vo vhodný okamih, iteračný výpočet zastaviť a výsledok zapísať. Zastaviť výpočet znamená, že členy postupnosti budeme stále pripočítavať k výsledku dovtedy, dokiaľ neplatí určitá podmienka. Podmienkou je nami požadovaná presnosť výpočtu.

$$|Y_{i-1} - Y_i| \geq \varepsilon * |Y_i| \quad (10)$$

Tento vzťah(10) znázorňuje absolútnu presnosť. Ak absolútna hodnota rozdielu nového člena rady a predchádzajúceho člena je menšia ako zadaná hodnota presnosti **eps** (ε).

4 Špecifikácia testov

Program som testoval na viacerých strojoch(vlastný PC, školský stroj Merlin). Na vlastnom PC som program testoval pod operačným systémom MS Windows 8 x64, Ubuntu a na školskom pod operačným systémom CentOS 6.4. Výsledky testov sú zdokumentované zo stroja Merlin. Zameral som sa na rôzne typy možných vstupov, ktoré môžu nastať (vrátane chybných).

Test 1: Chybne zadaný parameter → Vypísaná chybová hláška

Parametre: `./proj2 --asin 0gg4`

Výstup: `ERROR! Zle zadane cislo.... Pre viac info spusti napovedu --help`

Test 2: Arcus sínus čísla 0.5

Parametre: `./proj2 --asi`

Výstup: `5.2359877560e-01`

Test 3: Arcus sínus čísla -0.9999999999999999

Parametre: `./proj2 --asin -0.9999999999999999`

Výstup: `-1.5707963119e+00`

Test 4: Arcus sínus mimo definičný obor → vypíše nan (not a number)

Parametre: `./proj2 --asin 6`

Výstup: `nan`

Test 5: Druhá odmocnina z čísla 2

Parametre: `./proj2 --sqrt 2`

Výstup: `1.4142135624e+00`

Test 6: Druhá odmocnina mimo definičný obor → vypíše nan (not a number)

Parametre: `./proj2 --sqrt -4`

Výstup: `nan`

Test 7: Výpočet uhlov trojuholníka

Parametre: `./proj2 --triangle 0 0 1 0 0 2`

Výstup: `1.5707963268e+00`
`1.1071487178e+00`
`4.6364760900e-01`

Test 8: Vrcholy trojuholníka netvorí trojuholník → vypíše nan (not a number).

Parametre: `./proj2 --triangle 1 1 1 1 1 1`

Výstup: `nan`
`nan`
`nan`

5 Popis riešenia

K implementácii zadaných problémov som vychádzal zo záverov popísaných v predchádzajúcich kapitolách.

5.1 Ovládanie programu

Ide o konzolovú aplikáciu (má iba textové ovládanie) (CLI-pure) bez grafického rozhrania(GUI). Program má viac nezávislých funkcií. Všetky platné možnosti sú popísané v nápovede programu. Nápovedu je možné spustiť zadaním argumentu “--help”. Následne je nápoveda vypísaná na obrazovku a program skončí. Nebol prevedený žiadny výpočet. Ak je program spustený bez zadania argumentu vypíše chybovú hlášku, ktorá mu navrhne spustiť nápovedu a skončí. Takéto chybové hlásenie sa zobrazí i pri chybných zadaných argumentoch, neplatných číslach(napr.: obsahujúcich nezmyselné znaky). Pri korektnom zadaní argumentov program vypíše výsledok na štandardný výstup. Možné argumenty popísané v tabuľke:

nápoveda programu	--help
výpočet druhej odmocniny z čísla X	--sqrt X
výpočet arcus sínusu z čísla X	--asin X
výpočet uhlov trojuholníka jeho vrchol A =AX, AY ; B = BX, BY ; C = CX, CY	--triangle AX AY BX BY CX CY

5.2 Voľba dátových typov

Pre ukladanie výsledkov som použil dátový typ **double**. Pre ukladanie načítaných dát (pre výpočet trojuholníka) som použil pole, ktoré uloží šesť hodnôt typu **double**. Pre výpočet druhej odmocniny alebo arcus sínusu, postačuje jediná premenná na uloženie načítaných dát, ktorá je taktiež typu **double**.

5.3 Vlastná implementácia

Program sa skladá z jedného súboru zdrojového kódu, ktorý obsahuje všetky ostatné funkcie. Argumenty príkazového riadku sú spracované priamo vo funkcii **main**. Tým program rozpozná o akú funkciu ide. Čísla načítané ako vstupné reťazce sú analyzované a zisťujú sa chyby. Ak nie sú detekované chyby, reťazce sú prevedené na reálne čísla.

Funkcie **my_asin** a **my_sqrt** majú jednu vstupnú hodnotu - premennú **argX**. Do tejto premennej bolo uložené reálne číslo po prevode. Vo vnútri týchto funkcií okrem výpočtu prebieha aj overenie definičného oboru parametru funkcie (číslo x).

V prípade funkcie na výpočet trojuholníka **triangle** sú súradnice vrcholov uložené do poľa **arg[]**. Funkcia **triangle** má vstupný parameter predaný odkazom. Vo vnútri funkcie prebieha okrem výpočtu aj overenie trojuholníkovej nerovnosti. Všetky funkcie využívajú i funkciu **ABSOLUT**, ktorá slúži pre výpočet absolútnej hodnoty.

V prípade zadania neplatných argumentov príkazového riadku alebo chybné zadáných čísel je spustený výpis chybovej hlášky a program je ukončený.

6 Záver

Program počíta tri úlohy. Typ výpočtu si užívateľ zvolí sám prostredníctvom zadania argumentov pri spustení programu. Na výber má z týchto možností: druhá odmocnina zo zadaného čísla, arcus sínus a výpočet uhlov trojuholníka. Pri nedodržaní de finičných oborov funkcií je užívateľ na túto skutočnosť ihneď informovaný. Funkcia **--sqrt** počíta výsledok so zadaných hodnôt v intervale $< 0, \infty >$.

Funkcia **--asin** so zadaných hodnôt z intervalu $< -1, 1 >$. Mimo tieto intervaly priamo vracia výpis "nan" - not a number.

Program bol kompilovaný a následne testovaný v prostredí operačných systémov MS Windows 8 x64 a GNU/Linux so všetkými navrhnutými hodnotami. Vypočítané hodnoty som porovnal s hodnotami výpočtu funkcií pomocou matematickej knižnice jazyka C (**math.h**). Program dodržiava určené požiadavky na formát vstupných a výstupných dát.

Navrhnuté riešenie je použiteľné i na iných platformách, ktoré dokážu spustiť programy naprogramované v programovacím jazyku C.

Referencie

[1] Bartsch, H.-J. *Matematické vzorce*. 1. vyd. (I. Z. Tichý, Prekl.) Praha: SNTL - Nakladatelství technické literatury, 1983.

[2] HAVRLANT, Lukáš. Cyklometrické arcus funkce. *Matematika polopatě* [online]. 2006 [cit.2013-11-30]. Dostupné z: <http://www.matematika.cz/arcus>

A Metriky kódu

Počet súborov: 1 súbor

Počet riadkov zdrojového kódu: 7 (vrátane funkcie main)

Počet riadkov zdrojového textu: 240 riadkov

Veľkosť statických dát: 520 B

Veľkosť spustiteľného súboru: 10 351 B