Dokumentace k projektu pro předměty IZP a IUS

Iterační výpočty

projekt č. 2

27.11.2013

Autor: Martin Kačmarčík
Fakulta informačních technologií
Vysoké učení technické v Brně

Obsah

Kapitola 1

1. Úvod

Iterační výpočty se od těch rekurzivních liší jednou zásadní vlastností. Funkce pro výpočet nevolá sama sebe, jako to je u rekurzivních výpočtů, ale počítá za pomocí tzv. iterací - jednotlivých kroků. Pokaždé, když v cyklu proběhne posloupnost příkazů a cyklus se vrátí zpět na začátek (nebo na konec při nesplnění podmínky) proběhla jedna iterace. Iterace mají oproti rekurzivnímu způsobu výpočtu jednu zásadní výhodu - nejsou tak náročné na paměť. Vždyť uvědomme si, při každém rekurzivním volání, se do zásobníku musí uložit návratová adresa funkce. Kdežto při iteraci, si nic takového neukládám. Tento projekt se zabývá právě iteračními výpočty.

Tento dokument popisuje návrh a implementaci aplikace pro výpočet jednotlivých úhlu trojúhelníku, který je definován pomocí tří bodů, konkrétně jsou body určeny souřadnicemi [X,Y]. Také dokáže vypočítat arkus sinus popřípadě odmocninu reálného čísla. Program je ve formě konzolové aplikace, které jsou data zadána pomocí parametrů a která výsledek vypíše na standardní výstup. Zadává se reálné číslo v intervalu <-1,1> v případě arkus sinus, reálné kladné číslo v případě výpočtu odmocniny, nebo jednotlivé souřadnice v případě výpočtu úhlů trojúhelníku.

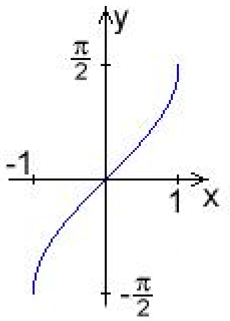
Kapitola 2

2. Analýza problému

Tato kapitola se zabývá matematickou funkcí arkus sinus. Dále popisuje základní vlastnosti odmocniny. Rozebírá Pythagorovu větu. Závěr kapitoly se zabývá výpočtem úhlů rovinného trojúhelníku, konkrétně cosinovou větou.

2.1 Arkus Sinus

Pro představu, co to vlastně arkus sinus je, bych hned nazačátek řekl, že to je inverzní funkce k funkci sinus. V praxi to znamená, že jeho graf je překlopen (zrcadlově) podle funkce y = x.



Obrázek 2.1: Graf funkce arkus sinus[1]

Definiční obor funkce je v intervalu <-1,1>. Obor hodnot je od <π/2, -π/2>. Zajímavá vlastnost je ta, že definiční obor funkce sinus je obor hodnot funkce arkus sinus. Tedy také obor hodnot funkce sinus je definičním oborem funkce arkus sinus. Velice důležitý je také jeho vztah s funkcí arkus cosinus. Ten je uveden v obrázku 2.2.

$$\mathrm{arsin}x + \mathrm{arcos}x = \frac{\pi}{2},$$

Obrázek 2.2: Vztah mezi arkus sinus a arkus cosinus. [1]

Arkus sinus je jedna z mnoha funkcí, která se dá počítat pomocí Taylorovy řady. Více o Taylorově řadě naleznete v kapitole 3.2.

2.2 Odmocnina

Odmocnina je opak k umocňování. Pokud nějaké číslo a umocním nadruhou, vznikne mi $a^{2.}$ Pokud odmocním a^2 získám zpět číslo a. Ovšem má to jeden malý háček. Odmocnina ze záporného čísla není definována. Číslo 'a' tedy náleží kladným reálným číslům včetně nuly. Odmocnina je zapisována tak jak je uvedeno na obrázku 2.3.

$$\sqrt[n]{a}$$

Obrázek 2.3: N-tá odmocnina z čísla a. [1]

Výpočet odmocniny je poněkud složitější oproti umocňování. Každý z nás dokáže z hlavy udělat 2^2, ale udělat √2 už dokáže málo kdo. K výpočtu lze použít tzv. Netwonova metoda (také se nazývá metoda tečen), která je popsána v kapitole 3.3.1.

2.3 Úhel trojúhelníku

Výpočet úhlu trojúhelníku není na papíře nic těžkého. Stačí vzít úhloměr a změřit daný úhel. Pokud mám ovšem zadány pouze souřadnice, potřebuju k tomu matematické vzorce. Konkrétně pro výpočet úhlů trojúhelníku se používá základní práce s vektory a dvě věty a to věta kosinova a Pythagorova.

2.3.1 Kosinova věta

Uvažujme v rovině o trojúhelníku ΔABC. Potom kosinova věta říká, že $a^2 = b^2 + c^2 - 2 \times b \times c \times \cos\alpha$. Zjednodušenně řečeno říká, že mocnina strany je rovna součtu mocnin stran zbylých a rozdílu dvojnásobku součinu zbylých stran a cosinu úhlu, který je naproti uvažované strany. Po jednoduché úpravě rovnice dokážeme získat vzorec pro výpočet jakéhokoliv vnitřního úhlu trojúhelníku.

2.3.2 Pythagorova věta

Pythagorova věta popisuje vztah stran v pravoúhlém trojúhelníku. Prakticky nám říká, jak spočítat délku třetí strany pravoúhlého trojúhelníku, pokud známe zbylé 2. Tento vztah je popsán na obrázku 2.4 kdy 'c' je délka přepony pravoúhlého trojúhelníku, a & b jsou odvěsny tohoto trojúhelníku.

$$c^2 = a^2 + b^2 \,,$$

Obrázek 2.4: Vztah délek stran v pravoúhlém trojúhelníku [1]

Tento vztah nám dále pomůžu k výpočtu délky strany trojúhelníku v soustavě souřadnic. Prakticky budeme počítat délku vektoru a to právě za pomocí Pythagorovy věty.

2.4 Absolutní a relativní přesnost

Přesnost zadaná v tomto projektu je eps = 1×10^{-12} coz v praxi znamená, že přesnost je na 11 platných číslic. Při každé iteraci tedy kontroluji, zda-li přírůstek není menší než zadaná přesnost a tudíž bych měl skončit. Tuto podmínku ovšem mohu testovat pomocí 2 způsobů. Jednak pomocí absolutní a nebo pomocí relativní přesnosti.

2.4.1 Absolutní přesnost

Absolutni přesnost je oproti té relativní konstantní. Testuji přírustek (po nějakém výpočtu) 'p' oproti absolutnímu eps. V našem případě oproti je eps = 1×10^{-12} . Ve while cyklu by to bylo zapsáno takto:

```
while(|p| >= eps) {
provadej vypocet;
}
```

Cyklus se bude opakovat dokud přírustek 'p' neklesne pod hodnotu, za kterou máme označenou naši přesnost.

2.4.2 Relativní přesnost

Absolutní přesnost měla v podmínce přesnost konstantní. Naproti tomu relativní přesnost, je vždy v podmínce relativní vůči výsledku a je proměnná. Podmínka ukončení nespecifikuje absolutní hodnotu posledního členu, ale určuje relativně jeho velikost vzhledem k celkové sumě. [2] V cyklu while by se dala zapsat takto:

```
while( |p| >= eps*|suma| ) {
provadej vypocet;
}
```

Kapitola 3

3. Návrh řešení

Tato kapitola se zabývá výpočtem funkce arkus sinus pomocí Taylorovy řady. Také popisuje Newtonovu metodu pro výpočet odmocniny. Závěr kapitoly se zabývá výpočtem úhlů trojúhelníku zadaného souřadnicemi tří bodů.

3.1 Volba rozsahu

Ve všech výpočtech podle zadání používám datový typ double. Typ double je jeden z největších datových typů. To prakticky znamená, že dokáže pojmout obrovské čísla a tudíž i zobrazit čísla s velkou přesností. Typ double narozdíl od datového typu int dokáže zobrazit i čísla s desetinou čárkou. Jeho rozsah je cca <1.7 × 10^{308},1.7 × 10^{-308}>. Argumenty funkce arkus sinus jsou sice v intervalu <-1,1> ovšem jeho výpočet pracuje s přírůstky, kdy poslední přírůstky jsou právně kolem zadané přesnosti, což jsou čísla velice malé a právě double nám dovolí neztratit žádnou z hodnot při výpočtu.

3.2 Výpočet Arkus Sinus

Samotný výpočet funkce arkus sinus je realizován pomocí Taylorovy řady. Více o něm najdete v podkapitole 3.2.1. Také si v podkapitole 3.2.2 povíme něco o problému (a o řešeních tohoto problému) této řady.

3.2.1 Taylorova řada a výpočet arkus sinus

Pro výpočet arkus sinus, bylo zadáno, že je nutno použít Taylorovu řadu. Taylorova řada je typ mocinné řady, která nám dovoluje počítat podobné příklady. Pomocí Taylorovy řady se dá mimo jiné spočítat funkce sinus čísla x, cotg čísla x a jiné (pozor! každá z těchto funkcí má jiný způsob výpočtu, všechny ale vychází z Taylorovy řady).

Zápis Taylorovy řady pro arkus sinus je na obrázku 3.1.

$$\arcsin x = x + \frac{1}{2}\frac{x^3}{3} + \frac{1}{2}\frac{3}{4}\frac{x^5}{5} + \frac{1}{2}\frac{3}{4}\frac{5}{6}\frac{x^7}{7} + \cdots = x + \sum_{n=1}^{\infty} \frac{(2n-1)!!}{(2n)!!}\frac{x^{2n+1}}{2n+1} \quad \text{pro } x \in \langle -1, 1 \rangle$$

Obrázek 3.1: Výpočet arkus sinus z čísla x pomocí Taylorovy řady[1]

Tento zápis nám říká, že arkus sinus z čísla x se dá spočítat jako suma nekonečně mnoho přírůstků. Aby jich doopravdy nebylo nekonečně mnoho zadáváme právě onu zmíněnou přesnost, která výpočet ve vhodný okamžik zastaví. Výsledkem funkce arkus sinus z čísla x je úhel v radiánech. Což je důležité si uvědomit, protože by byla chyba předpokládat, že výsledek je ve stupních (vycházeli by nám neuvěřitelně malé úhly).

3.2.2 Optimalizace Taylorovy řady

Při výpočtu arkus sinus pomocí Taylorovy řady ovšem nastanou potíže, pokud budu chtít získat úhel z arkus sinus z čísla x v rozmezi <-1, -0.8> ⋃ <0.8, 1>. Taylorova řada totiž v tomto rozmezí konverguje velice pomalu a počet iterací značně narůstá. Výsledek je tedy zdlouhavý a ještě k tomu není ani zcela přesný. Proto je v tomto rozmezí vhodné zvolit optimalizaci.

Pro optimalizaci arkus sinus z čísla x jsem si zvolil převod sinus z na cosinus (převod je na obrázku 3.2), který lze aplikovat i na arkus sinus a arkus cosinus (konkrétně vztah je takový, že $\cos^{-1}$(x) = arcos (x). To samé platí i pro sinus a arkus sinus), a tudíž jsem ve výsledku počítal pomocí Taylorovy řady arkus cosinus z čísla x a né arkus sinus z čísla x.

$$\cos x = \sqrt{1 - \sin x^2}$$

Obrázek 3.2: Převod mezi sinus a cosinus[3]

Výsledek je arkus cosinus z čísla x, který samozřejmě potřebujeme převést na požadovaný arkus sinus. Pro tento převod se použije základní vztah těchto dvou funkcí, který je uveden na obrázku 3.3.

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

Obrázek 3.3: Převod mezi arkus sinus a arkus cosinus[3]

Pomocí této optimalizace jsme schopni efektivně, rychle a hlavně přesně počítat hodnoty arkus sinu z intervalu <-1, -0.5> ∪ <0.5, 1>, u kterých nepřesnost neoptimalizovaného výpočtu nabývá s roustoucím argumentem vyšších a vyšších časů a odchylek.

3.3 Výpočet odmocniny

Výpočet odmocniny v tomto projektu je realizován pomocí newtonovi metody. O Newtonově metodě se více zmiňuji v podkapitole 3.3.1. Nezapomeňme ma základní podmínku: argument odmocniny nesmí být záporný, protože odmocnina ze záporného čísla není definována.

3.3.1 Newtonova metoda

Newtonova metoda (Newtonova proto, že jí dal základ, i když ji uplně nedopracoval), nebo také metoda tečen, je iterační metoda, která nám dovoluje numericky počítat soustavy nelineární rovnic. Mimo jiné nám také dovolí počítat odmocninu, kterou potřebujeme pro výpočet strany trojúhelníku, pokud máme zadané body v soustavě souřadnic (více o tom proč se dovíme v kapitole 3.4 Výpočet úhlu rovinného trojúhelníku).

Newtonova metoda funguje, zjednodušenně řečeno, tak, že pomocí vzorce z obrázku 3.4 při každé iteraci vypočítám nový člen, který se počítá za pomocí členu starého. Po jedné iteraci je vždy důležité porovnat starý s výsledek, který je nutné si uložit, s výsledkem novým, zda-li již podíl těchto dvou výsledků není menší než přesnost.

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

Obrázek 3.4: Newtonova metoda pro výpočet odmocniny[4]

Důležité je upozornit také na to, že Newtonova metoda při relativní přesnosti nedokáže spočítat odmocninu z nuly. Proto je nutno tuto hodnotu nastavit napevno. Proč to tak je, lze vyčíst z obrázku 3.4

3.4 Výpočet úhlu rovinného trojúhelníku

Abychom vypočítali úhly trojúhelníku za pomocí cosinovi věty, potřebujeme znát strany daného trojúhelníku. Ty je samozřejmě nutné vypočítat. Výpočtu stran při známích souřadnicích se věnuje kapitola 3.4.1.

3.4.1 Základní práce s vektory - vektor a pythagorova věta

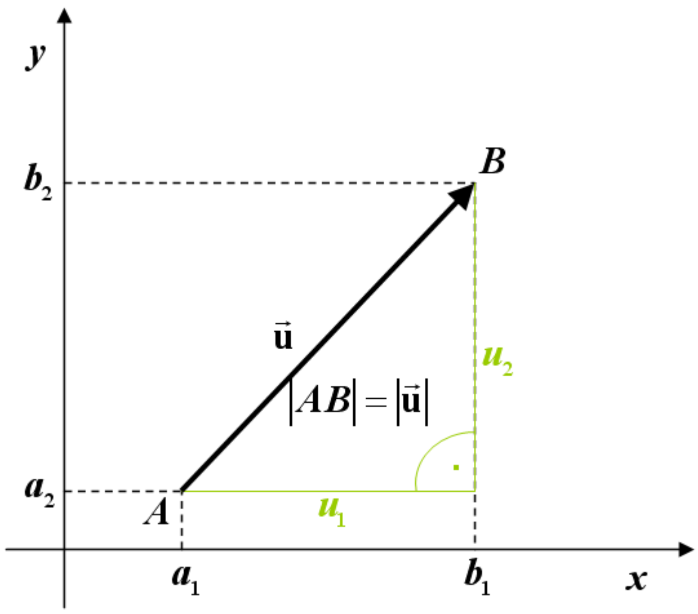
Základní myšlenkou použití vektorů při tomto projektu je to, že pomocí dvou bodů dokážu zjistit vektor u, jehož velikost je rovna právě vzdálenosti těchto dvou bodů. Vektor pomocí dvou bodů, například pomocí bodu 'A' a 'B', se dá určit tak, že provedu B-A, což v praxi znamená, že provedu tento výpočet: $(B_x - A_x, B_y - A_y)$. Čímž získám vektor $u = (u_1, u_2)$.

Jedna ze základních operací s vektorem je právě určení jeho velikosti. Tato operace se provede pomocí pythagorovi věty. Výpočet je na obrázku 3.5.

$$|u| = \sqrt{u_1^2 + u_2^2}$$

Obrázek 3.5: Výpočet délky vektoru[5]

Na obrázku 3.6 je patrné, kde se Pythagora věta při výpočtu délky vektoru „nalézá".



Obrázek 3.6: Pythagorova věta při výpočtu délky vektoru [5]

Pokud již známe délky stran trojúhelníku, stačí aplikovat cosinovu větu, kterou je nutné si ovšem prvně správně upravit do tvaru: $\alpha = \cos^{-1} [(b^2 + c^2 - a^2)/ 2\times b\times c]$. Jelikož ovšem není možné použít cosinus, je otázka jak vyřešit $\cos^{-1}$. Řešení se nachází ve vztahu mezi cosinem a arkus kosinem, které je takové: $\cos^{-1}(x) = \arccos(x)$. Jelikož $\cos^{-1}$ je jediné co potřebujeme, tento problém se skoro vyřešil. Poslední otázka je, jak získat arkus kosinus, když můžeme používat pouze arkus sinus? Odpověď na tuto otázku již byla zmíněna. Je to vztah, který je právě na obrázku 2.2. Díky tomuto vztahu již známe vše potřebné pro výpočet jakéhokoliv vnitřního úhlu trojúhelníku zadaného třemi body v podobě souřadnic.

3.4.2 Ověření správnosti trojúhelníku

Důležitým krokem pro výpočet úhlů v trojúhelníku je také zjistit to, zda-li uživatel trojúhelník vůbec zadal. Pro ověření tohoto faktu používáme jednoduchý výpočet, který porovnává délky stran podle pravidla: Součet dvou libovolných stran musí být vždy větší, než strana třetí. Je tedy nutné vyzkoušet všechny tyto kombinace a zjistit, zda-li sme byli všude úspěšní (jedná se o trojúhelník), či nikoliv (a tudíž se o trojúhelník nejedná.

3.5 Absolutní hodnota čísla

Pro počítání s relativní přesností se používá absolutní hodnota z nějakého čísla x. Matematicky je absolutní hodnota z čísla x velice jednoduchou záležitostí. Prakticky jde o kontrolu, zda-li je číslo záporné, pokud ano, absolutní hodnota vrátí x*(-1). Pokud je kladné nebo rovné nule, nestane se nic.

3.6 Specifikace testů

Test č. 1: Nesmyslná data --> detekce chyby

--asin blablabla (chyný argument)

--sqrt (chybí z čeho se má počítat odmocnina)

--triangle 0 0 0 0 0 0 (zadané souřadnice neoznačují trojúhelník)

Test č. 2: Chybná syntaxe --> detekce chyby

--triangle 1 2 3 4 5 (nedostatečný počet souřadnic)

--triangle 123456 (chybi mezera mezi souřadnicemi)

--asin1 (opět chybí mezera)

--asin 0,5 (čárka místo tečky - špatně označena desetiná čárka)

Test č. 3: Data mimo povolený rozsah hodnot --> detekce chyby

--asin 4 (argument pro arkus sinus má rozsah <-1,1>)

--sqrt -1 (argument odmocnina musí být kladný, popřípadně nula)

Test č. 4: Správná data --> předpokládaný správný výsledek

--sqrt 1 -> 1

--asin 1 -> 1.57079632679

--triangle 0 0 1 0 0 2 -> 1.5707963268e+00, 1.1071487178e+00, 4.6364760900e-01

--asin 0.5 -> 0.52359877560

--sqrt 4 -> 2

Kapitola 4

4. Popis řešení

Při implementaci jsem vycházel z poznatků získaných při analýze popsané v předchozích kapitolách. Odmocnina je implementovaná podle Newtonovy metody. Výpočet arkus sinus z čísla x podle Taylorovy řady. Výpočet úhlů trojúhelníku podle cosinovy věty.

4.1 Ovládání programu

Program je konzolová aplikace, jehož vstup se zadává (a tedy i aplikaci se ovládá) pouze pomocí parametrů. Program může uživatel zadat se čtyřmi parametry. Parametr "--help" neprovádí žádný výpočet pouze vypíše nápovědu k užívání programu. Parametr "--sqrt x" vypíše odmocninu z čísla x. Parametr "--asin x" vypíše arkus sinus z čísla x. V obou případech je nutné místo x dát reálné číslo v požadovaném intervalu. Parametr "--triangle Ax Ay Bx By Cx Cy" vypise na obrazovku úhly trojúhelníku zadanými souřadnicemi tří bodů A, B a C, kdy pořadí je pevně určeno. Program vypíše úhly v pořadí: alpha, beta, gama.

4.2 Volba datových typů

Volba datových typů byla pevně daná, jelikož byla omezená zadáním. Ve všech výpočtech je zadáno použití typu double. Typ double je zároveň pro tento program vhodný, protože disponuje dostatečnou přesností, aby nevznikali přílišné odchylky, které mohou být pro uživatele značně nepříjemné (například pokud chce s odmocninou počítat v dalších výpočtech)

4.3 Vlastní implementace

O zpracování parametrů programu se stará funkce parametrCheck, která se volá jako první funkce ve funkci main. Funkce parametry zpracuje a nastaví příslušný mód, který uživatel chce použít. Módy jsou 3 a jsou definovány proměnnými: sqrtMode, asinMode, triangleMode. Všechny jsou datového typu int a nabívají buď hodnot 1 nebo 0 (1 - mod je aktivni, 0 - mod neni aktivní).

Následně se volá funkce modeSelect, která nazákladě zapnutého módu provede danou akci. Pokud je zaplý mód pro odmocninu, funkce zavolá funkci my_sqrt, která vrátí hodnotu odmocniny. Hodnota je poté funkcí modeSelect vypsána na standartní výstup.

Pokud je zaplý mód pro arkus sinus, funkce zavolá funkci my_asin, která vrátí hodnotu arkus sinus zadaného čísla (pokud je číslo správné). Hodnota je poté vypsána na standartní výstup.

Pokud je zaplý mód pro výpočet úhlů trojúhelníku, funkce nejdřív zavolá funkci vypoctiStrany, která provede výpočet stran ze zadaných souřadnic. Poté se jednoduchou podmínkou zjistí, jestli zadané souřadnice vůbec označují trojúhelník. Pokud ne, vypíše se chybové hlášení a program se ukončí. Pokud ano, tak se následně volá pro každý úhel funkci vypoctiUhel, která provede výpočet pomocí kosinovy věty. Následné 3 úhly, každý na jeden řádek, vypíše na standartní výstup.

Kapitola 5

5. Závěr

Program dokáže provádět 3 samostatné operace. Výpočet úhlů trojúhelníku, výpočet odmocniny a výpočet arkus sinus z čísla x. Výpočet arkus sinus, je díky optimalizaci přesný a rychlý, ovšem bez této optimalizace, díky pomalu konvergující Taylorově řadě, by výsledky byly nepřesný a program by mohl počítat i několik minut. Výpočet Newtonovy metody je přesný, ovšem obsahuje určité nedostatky, třeba výpočet odmocniny z čísla nula. Tuto hodnotu je teda nutné ošetřit.

V programu jsou používané pouze základní matematické operace +, -, ×, ÷. Program počítá v zadání určené kontrolní parametry správně a hodnoty vychází stejně jako v kontrole. Navržené řešení, je přenositelné na všechny platformy. Program byl úspěšně testován v prostředí operačních systému Linux a Microsoft Windows.

Zdroje

[1] WIKIMEDIA FOUNDATION. *Wikipedie: Otevřená encyklopedie* [online]. 2001, 2013 [cit. 2013-12-03]. Dostupné z: http://cs.wikipedia.org/wiki/Hlavn%C3%AD_strana

[2] Řešení rekurentních problémů. *Fakulta informačních technologií, VUT Brno: Informační systém*[online]. 2013 [cit. 2013-11-27]. Dostupné z: https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/IZP-IT/lectures/6s_rek-problemy.pdf?cid=9416

[3] *Matematické Fórum* [online]. 2002, 2005 [cit. 2013-12-03]. Dostupné z: http://forum.matematika.cz/

[4] FAKULTA INFORMAČNÍCH TECHNOLOGIÍ, Vysoké učení technické v Brně. *Informační systém: Wiki stránky předmětu IZP* [online]. 2013 [cit. 2013-12-03]. Dostupné z: https://wis.fit.vutbr.cz/FIT/st/cwk.php?id=9416&csid=538050

[5] SCHUHMEIER, Petr. *Artistoteles: náš online učitel* [online]. 2006, 21. listopadu 2011 [cit. 2013-12-03]. Dostupné z: http://www.aristoteles.cz/

Metriky kódu

Počet souborů : 1 soubor

Počet řádků zdrojového textu: 228 řádků

Velikost statických dat: 8 719 B

Velikost spustitelného souboru: 34 981 B (systém Windows, 32 bitová architektura)