Dokumentácia k projektu pre predmety IZP a IUS

Iteračné výpočty

projekt č. 2

5. december 2014

Autor: Michal Ondrejó, xondre08@stud.fit.vutbr.cz

Fakulta Informačních Technologií

Vysoké Učení Technické v Brně

Obsah

1. Úvod

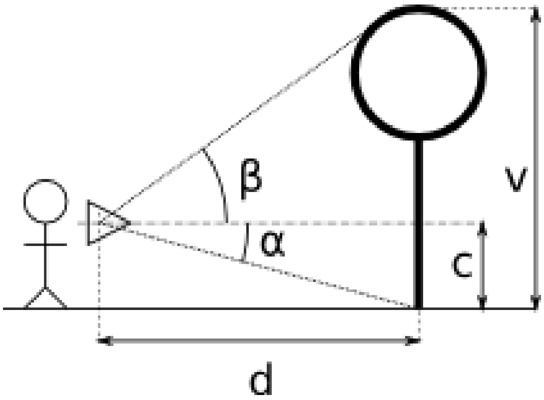
Tento dokument popisuje návrh a implementáciu aplikácie pre výpočet vzdialenosti od predmetu a výšky premetu len pomocou meracieho prístroja (napríklad mobilného telefónu s G-senzorom). Tento spôsob merania sa javí ako veľmi zaujímavý.

Dokument sa skladá z viacerých častí. V druhej kapitole sa dozvieme o analýze a princípe riešenia. V tretej časti s návrhom riešenia problému, v štvrtej časti s popisom riešenia a presnosť výpočtu v jednotlivých metódach a v piatej časti mám jednotlivé testy na chyby a očakávaný výstup programu, zo správne zadaných parametrov.

2. Analýza problému a popis princípu jeho riešenia

2.1. Zadanie projektu

Cieľom tohto projektu bolo vytvorenie programu, ktorý vypočíta vzdialenosť od predmetu a jeho výšku s pomocou prístroja, ktorý keď nahneme tak, aby ukazoval na spodok objektu (obr. 2.1), dostaneme vzdialenosť od predmetu a keď potom nahneme prístroj tak, aby ukazoval na vrch objektu (obr. 2.1), dostaneme aj výšku meraného objektu.



Obrázok 2.1

2.2. Podúlohy

V tomto projekte sme mali aj dve podúlohy, v ktorých som mal vytvoriť funkcie, ktoré by vypočítali tangens uhla „x“ pomocou:

1. Taylorovho polynómu

$$\tan(x) = x + \frac{x^3}{3} + \frac{2x^5}{15} + \frac{17x^7}{315} + \frac{62x^9}{2835} + \cdots \quad (1)$$

2. Zreťazených zlomkov

$$\tan(x) = \cfrac{1}{\cfrac{1}{x} - \cfrac{1}{\cfrac{3}{x} - \cfrac{1}{\cfrac{5}{x} - \cfrac{1}{\cfrac{7}{x} - \cdots}}}} \tag{2}$$

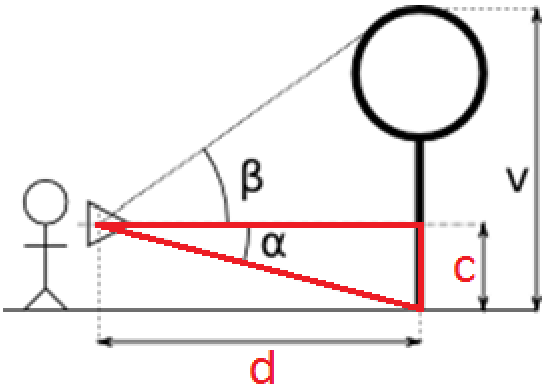
3. Návrh riešenia problému

Najprv som vypočítal vzdialenosť od meraného predmetu pomocou vzorca (3).

$$\tan\alpha = \frac{c}{d} \tag{3}$$

Z ktorého som pomocou úpravy dostal:

$$d = \tan\alpha * c \tag{4}$$



Obrázok 3.1:Výpočet vzdialenosti od objektu

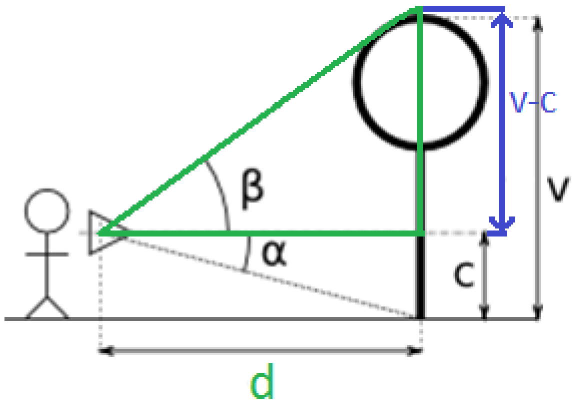
Keďže som už mal zadanú výšku, v ktorej sa nachádza merací prístroj, tak my stačilo vypočítať len $(v - c)$ (viď obrázok č. 3.2). Keďže som už mal aj vypočítanú vzdialenosť od predmetu, mohol som ju vypočítať pomocou vzorca (5).

$$\tan\beta = \frac{v - c}{d} \tag{5}$$

Z ktorého som pomocou úpravy dostal:

$$v - c = \tan\beta * d \tag{6}$$

Keď som už dostal výšku $(v - c)$, pripočítal som k nej výšku „c“, ktorá bola prednastavená na 1,5 metra alebo bola zadaná v argumentoch programu za parametrom „–c“ a tak som dostal výslednú výšku „V“.



Obrázok 3.2: Výpočet výšky objektu

Pri týchto výpočtoch som používal funkciu cfrac_tan, ktorá sa pri testovaný ukázala presnejšia už po pár iteráciách, ako počítanie tangensu pomocou Taylorovho polynómu.

3.1. Počet iterácií

Keďže máme počítať s presnosťou na 10 miest, tak som si musel zvoliť, koľko iterácií použijem. Ako maximálny počet iterácií som si zvolil číslo 8. Toto číslo som získal pomocou testovania, keď som zadal parametre „--tan 1.4 1 13,“ kde číslo „1.4“ reprezentuje maximálnu hodnotu akú môže „X“ nadobudnúť a čísla „1“ a „13“ reprezentujú počet iterácií, teda od hodnoty 1 až po 13. Výstup programu je vidieť na tabuľke č. 1.

Počet iterácii	tan x – z matematickej knižnici math.h	tan x – tangens, ktorý som dostal pomocou zreťazených zlomkov	Absolútny rozdiel medzi matematickou knižnicou a pomocou zreťazených zlomkov
1	5.79788371548288683499	-1.45833333333333348136	7.25621704881621987226
2	5.79788371548288683499	8.49655172413793557951	2.69866800865504874452
3	5.79788371548288683499	5.86430911761382667180	0.06642540213093983681
4	5.79788371548288683499	5.79943246311010174310	0.00154874762721490811
5	5.79788371548288683499	5.79790865214289663498	0.00002493666000979999
6	5.79788371548288683499	5.79788400459679031940	0.00000028911390348441
7	5.79788371548288683499	5.79788371800496715736	0.00000000252208032236
8	5.79788371548288683499	5.79788371550002690213	0.00000000001714006714
9	5.79788371548288683499	5.79788371548298275826	0.00000000000009592327
10	5.79788371548288683499	5.79788371548288949953	0.00000000000000266454
11	5.79788371548288683499	5.79788371548288949953	0.00000000000000266454
12	5.79788371548288683499	5.79788371548288949953	0.00000000000000266454
13	5.79788371548288683499	5.79788371548288949953	0.00000000000000266454

Tabuľka číslo 1: Zisťovanie počtu iterácií

3.2. Voľba dátového typu

Pri výpočte uhla som pracoval s desatinnými číslami, a preto som si zvolil dátový typ *double*[1]. Keďže počet iterácií môže byť len celočíselná hodnota od 1 až po 13, tak som preň zvolil dátový typ *int*[2].

4. Popis vlastného riešenia

4.1. Výpočet tangensu

Tangens som počítal pomocou dvoch metód a to:

- Taylorovým polynómom
- Zreťazenými zlomkami

4.1.1. Výpočet pomocou metódy Taylorovým polynómom

Na výpočet tangensu pomocou použitia Taylorových polynómov sa používa vzorec (č. 1). V princípe ide o to, že s pribúdajúcim počtom iterácií (obrázok č. 4.1) sa pomaly približujeme k skutočnej hodnote tan(x). V čitateli mám hodnotu „X“, ktorá sa každou ďalšou iteráciou vynásobí *x^2. Táto hodnota sa ešte musí vynásobiť s koeficientom (*1, 1, 2, 17, 62, 1382, 21844, 929569, 6404582, 443861162, 18888466084, 113927491862, 58870668456604*). Tento koeficient veľmi rýchlo narastá a prekračuje hodnotu *int*. Kvôli tomu som musel preň alokovať pole, s dátovým typom long long, o veľkosti trinástich prvkov. Pre menovateľa mám podobné pole, pri tomto poli som ale našiel určitý vzťah(č.7), kde „K“ reprezentuje novovytvorený koeficient *{1,3,5,21,9,55,39,105,17,171,105,253,75,351}* a kvôli tomu som nemusel použiť pole s dátovým typom long long ale stačilo mi použiť menšie pole s int, ktorý zaberá menej pamäte.

$$menovateľ_n = menovateľ_{n-1} * K \qquad \text{,kde } menovateľ_0 = 1 \qquad (7)$$

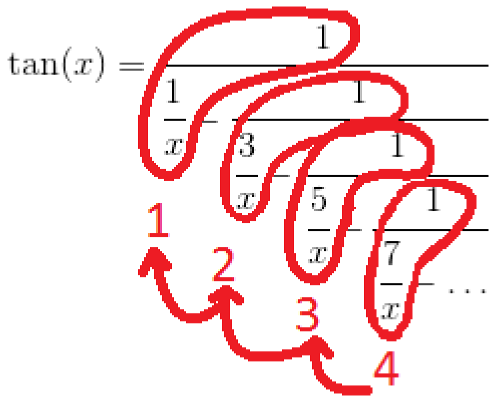
$$\tan(x) = \overset{1.}{\boxed{x}} + \overset{2.}{\boxed{\frac{x^3}{3}}} + \overset{3.}{\boxed{\frac{2x^5}{15}}} + \overset{4.}{\boxed{\frac{17x^7}{315}}} + \overset{5.}{\boxed{\frac{62x^9}{2835}}} + \cdots$$

Obrázok 4.1 Výpočet pomocou Taylorových polynómov

[1] Desatinná hodnota s presnosťou na približne 20 desatinných miest
[2] Zväčša 32-bitové celé číslo v dvojkovom doplnkovom kóde. -Integer

4.1.2. Výpočet pomocou metódy zreťazených zlomkov

Pri použití zreťazených zlomkov sa používa vzorec (č. 2).Najprv som chcel riešiť tento príklad pomocou rekurzívnej[3] funkcie, ale ako som zistil neskôr, tak tento spôsob nebol najideálnejší. Potom som prišiel na spôsob, kde by sa tangens počítal od konca. Tento spôsob riešenia je lepšie vidieť na obrázku číslo 4.2.



Obrázok 4.2 Výpočet pomocou zreťazených zlomkov

4.2. Presnosť

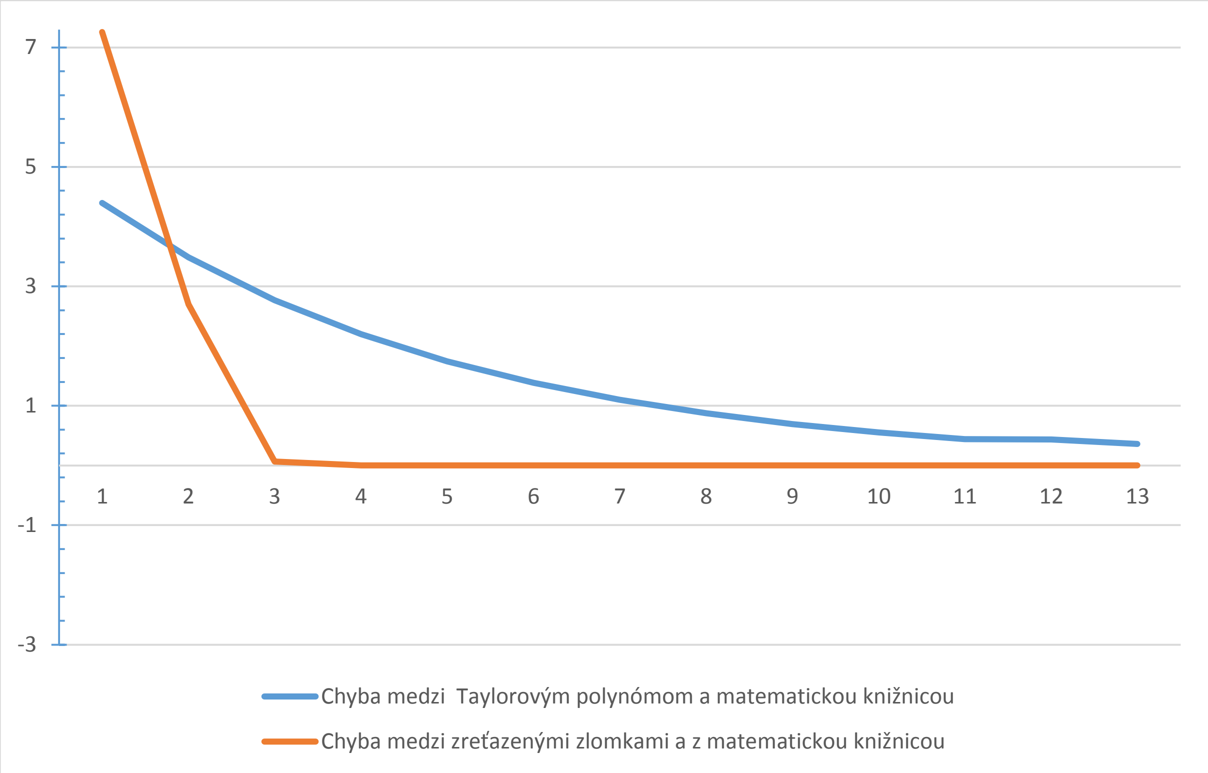
Presnosť počítania tangensu, pomocou metódy Taylorovým polynómom a metódou s použitím zreťazených zlomkov, je dosť rozdielna. Pri počítaní s Taylorovým polynómom je presnosť na jednu číslicu až pri ôsmej iterácii, zatiaľ čo pri metóde so zreťazenými zlomkami je presnosť pri ôsmej iterácii už až na 11 platných číslic (tabuľka č. 2).

Počet iterácií	Chyba medzi Taylorovým polynómom a matematickou knižnicou	Chyba medzi zreťazenými zlomkami a matematickou knižnicou
1	4.397883715482887367898001684807	7.256217048816219872264809964690
2	3.483217048816220628992823549197	-2.698668008655048744515170255909
3	2.766118382149554300042382237734	-0.066425402130939836808920517797
4	2.197220106593999133792749489658	-0.001548747627214908106907387264
5	1.745373580806345081839481281349	-0.000024936660009799993531487416
6	1.386451081882141167511690582614	-0.000000289113903484405909694033
7	1.101338572877343047196063707815	-0.000000002522080322364672611002
8	0.874857189836195203724855673499	-0.000000000017140067143373016734
9	0.694949877019277018064258300001	-0.000000000000095923269327613525
10	0.552039049803908277169739449164	-0.000000000000002664535259100376
11	0.438516679549718624286924750777	-0.000000000000002664535259100376
12	0.433212125900194422456479514949	-0.000000000000002664535259100376
13	0.361578970033452407051299815066	-0.000000000000002664535259100376

Tabuľka číslo 2: Presnosť funkcii voči matematickej knižnici

[3] Funkcia, ktorá vo svojom tele obsahuje príkaz na volanie samej seba

Rozdiel medzi týmito dvomi metódami je ten, že pri počítaní tangensu pomocou Taylorovho polynómu, sa približujeme k výsledku zdola (výsledná hodnota sa stále zväčšuje), zatiaľ čo pri metóde pomocou zreťazených zlomkov sa približujeme zhora(výsledná hodnota sa stále zmenšuje).



Obrázok 4.3:Chyba medzi funkciami a mat. knižnicou

5. Špecifikácia testov

Z návrhu riešenia vyplýva niekoľko rizikových oblastí, ktoré som musel otestovať, napríklad zlý počet argumentov, v argumentoch ktoré mali obsahovať len čísla, obsahovali aj nepovolené znaky, počet iterácií bol zadaný menší ako 1 alebo presiahol hodnotu 13 alebo uhol nebol zadaný v intervale ⟨0; 1.4⟩.

Test 1: chybný uhol−→ Detekcia Chyby

--tan 0 1 2

--tan 1.6 1 10

Test 2: chybná syntax−→ Detekcia Chyby

--tan 1.3p 1 13

--tan 1.3 p 13

--chyba

50 -c –m 1.k3 0.2

Test 3: chyba iterácii−→ Detekcia Chyby

--tan 1.3 -5 13

--tan 1.3 13 5

--tan 1.3 1 20

Test 4: nesprávny počet argumentov−→ Detekcia Chyby

--tan

--tan 1 2 5 8

-c –m 1.3 0.2

-c 50 –m 1.3 0.2 20

Test 5 správne argumenty−→ Predpokladané správne hodnoty

vstup	**Očakávaný výstup**
--help	Nápoveda k programu
--tan 0.123 5 5	5 1.236241e-001 1.236241e-001 8.694018e-013 1.236241e-001 1.387779e-017
--tan 1.36 10 10	10 4.673441e+000 4.426780e+000 2.466612e-001 4.673441e+000 8.881784e-016
--tan 1.39999 13 13	13 5.797538e+000 5.436041e+000 3.614964e-001 5.797538e+000 0.000000e+000
--tan 1.024 2 4	2 1.642829e+000 1.381914e+000 2.609155e-001 1.694293e+000 5.146377e-002 3 1.642829e+000 1.532034e+000 1.107955e-001 1.643755e+000 9.250672e-004 4 1.642829e+000 1.595748e+000 4.708106e-002 1.642841e+000 1.133960e-005
-m 0.365	3.9254472930e+000
-m 0.365 1.19	3.9254472930e+000 1.1305372389e+001
-c 100 -m 0.365 1.19	2.6169648620e+002 7.5369149257e+002
-c 100 -m 0.365	2.6169648620e+002
-c 2 -m 0.365	5.2339297240e+000

Tabuľka číslo 3: Vstupy a výstupy programu

6. Záver

Zistil som, že počítanie tangens pomocou Taylorových polynómov je neefektívne , keďže má oveľa vyššiu nepresnosť, ako keď som použil výpočet, pomocou zreťazených zlomkov (obr. 4.1).

Tento program splňuje požiadavky štandardu c99 a bol otestovaný na operačnom systéme Linux a aj v operačnom systéme Windows, na ktorých pracoval podľa očakávania.

A. Metriky kódu

Počet súborov: 1 súbor

Počet funkcií: 12

Počet riadkov zdrojového textu: 383 riadkov

Veľkosť statických dát: 552B

Veľkosť spustiteľného súboru: 12662B