

Dokumentace k projektu pro předměty IZP a IUS

Iterační výpočty

projekt č. 2

1. decembra 2014

Autor: Ján Mochňak, xmochn00@stud.fit.vutbr.cz
Fakulta Informačních Technologí
Vysoké Učení Technické v Brně

Obsah

1	Úvod	1
2	Analýza problému	2
2.1	Zadanie problému	2
2.2	Možnosti výpočtu funkcie \tan	2
2.2.1	Taylorov rozvoj	2
2.2.2	Zreťazenie zlomkov	3
3	Návrh riešenia problému	4
3.1	Porovnanie presností výpočtov tangens	4
3.2	Výpočet vzdialenosti a výšky meraného objektu	4
3.2.1	Výpočet vzdialenosti	4
3.2.2	Výpočet výšky	4
4	Špecifikácia testov	5
5	Popis riešenia	6
5.1	Analýza vstupných parametrov	6
5.2	Ovládanie programu	6
5.3	Implementácia	6
6	Záver	7
A	Metriky kódu	8

Kapitola 1

Úvod

Dokumentácia k druhému projektu do predmetov *základy programovania* a *softvérové inžinierstvo*, ktoré sa vyučujú na VUT v Brně.

Výsledkom projektu je konzolová aplikácia napísaná v jazyku *C*, splňujúca štandard *C99*. Aplikácia má dve hlavné úlohy, jednou z nich je overenie správnosti implementácie matematickej funkcie *tangens* pomocou *Taylorového rozvoja* a pomocou *zreťazených zlomkov*. Tieto funkcie sú implementované len pomocou základných matematických operácií. Druhá úloha tejto aplikácie spočíva vo výpočte vzdialenosti a výšky meraného objektu, v ktorej nájdeme využitie funkcie *tangens*.

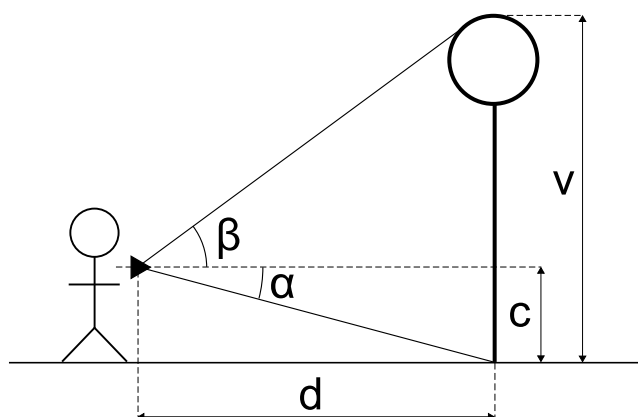
Kapitola 2

Analýza problému

Pre výpočet vzdialeností budeme potrebovať funkciu tangens, takže sa na možnosti implementácie tejto funkcie pozrieme bližšie.

2.1 Zadanie problému

Hlavnou úlohou je vypočítanie vzdialenosti a výšky meraného objektu, predpokladáme, že merací prístroj má možnosť nastavenia aj jeho výšky. Grafické zobrazenie tohoto problému je v obrázku 2.1.



Obr. 2.1: Náčrtok meracieho zariadenia a meraného objektu.

2.2 Možnosti výpočtu funkcie tan

Matematicky je funkcia tan definovaná ako:

$$\tan(x) = \frac{\sin(x)}{\cos(x)} \quad (2.1)$$

Ak by sme chceli teda vypočítať tangens uhla x , potrebovali by sme k tomu ešte aj funkcie sínus a cosínus. A preto sa pozrieme na ďalšie možné výpočty.

2.2.1 Taylorov rozvoj

Taylorov rozvoj by sme dokázali vyjadriť takto:

$$\tan(x) = x + \frac{x^3}{3} + \frac{2x^5}{15} + \frac{17x^7}{315} + \frac{62x^9}{2835} + \dots \quad (2.2)$$

Pričom v našom prípade rátame len s prvými 13 členmi Taylorovej rady, a to hlavne z dôvodu veľkosti dátového typu `long unsigned int` ($2^{63} - 1$). Maximálny počet iterácií pre spresnenie výsledku je v tomto prípade len 13.

2.2.2 Zreťazenie zlomkov

Funkciu tangens pomocou zreťazenia zlomkov, vyjadríme takto:

$$\tan(x) = \frac{1}{\frac{1}{x} - \frac{3}{\frac{1}{x} - \frac{5}{\frac{1}{x} - \frac{7}{\frac{1}{x} - \dots}}}} \quad (2.3)$$

Výhodou tejto možnosti je hlavne možnosť viacero iterácií a teda vyššej presnosti ako u Taylorového rozvoja. A z tohto dôvodu volíme, pre výpočet výšky a vzdialenosti meraného objektu túto funkciu.

Kapitola 3

Návrh riešenia problému

Po analýze funkcie tangens, som sa rozhodol akceptovať len hodnoty $0 < \alpha \leq 1.4 \text{ rad}$, ktoré sa nachádzajú v prvom kvadrante. Toto obmedzenie je pre naše účely dostačujúce, hodnota úhlov α a β by podľa obrázku 2.1 nemala presiahnuť 90° .

3.1 Porovnanie presností výpočtov tangens

Pri porovnávaní je potrebné zistiť ako sa veľmi sa odchyľuje výsledok iterácií z našich implementácií funkcie tangens oproti funkcii *tan* z matematickej knižnice `<math.h>`. Rozdielom výsledkov je vyjadrená absolútna odchýlka pre každú iteráciu, ale aj funkciu.

3.2 Výpočet vzdialenosti a výšky meraného objektu

Ako implicitnú výšku zariadenia volíme 1.5, ktorá je definovaná zadáním. Pre výpočty použijeme definíciu *tan* v nasledujúcom tvare:

$$\tan(\alpha) = \frac{a}{b} = \frac{\text{protiľahlá}}{\text{priľahlá}} \quad (3.1)$$

Inými slovami $\tan(\alpha)$ je pomer dĺžok odvesny protiľahlej k tomuto uhlu a dĺžky odvesny k nemu priľahlej.

Presnosť, resp. počet iterácií sme si zvolili 11, pretože po tejto iterácii je výsledok pre datový typ `double` rovnaký. Tým sme dosiahli najvyššiu možnú presnosť.

3.2.1 Výpočet vzdialenosti

Vzdialenosť objektu od meracieho prístroja si z obrázku 2.1 vyjadríme takto:

$$\tan(\alpha) = \frac{c}{d} \Rightarrow d = \frac{c}{\tan(\alpha)} \quad (3.2)$$

Pri výpočte $\tan(\alpha)$ využijeme metódu zret'azených zlomkov (viz. 2.2.2).

3.2.2 Výpočet výšky

Výšku meraného objektu si vieme opäť odvodiť z obrázka 2.1:

$$\tan(\beta) = \frac{v_1}{d} \Rightarrow v = c + \tan(\beta) * d \quad (3.3)$$

K výslednej výške je potreba pripočítať výšku zariadenia *c*. Pri výpočte $\tan(\beta)$ využijeme metódu zret'azených zlomkov (viz. 2.2.2). Pre tento výpočet si musíme najprv vypočítať jeho vzdialenosť *d* pomocou 3.2.1.

Kapitola 4

Špecifikácia testov

Test 1: Chybná syntaxe → Detekce chyby.

```
./proj2                ; nedostatok parametrov
./proj2 -tan            ; správne -tan
./proj2 -m              ; je potrebné zadať hodnotu uhlu
./proj2 -m -c 2.2       ; výška musí byť nastavená pred uhlami
./proj2 -c 2.2          ; potreba nastaviť hodnotu uhlu -m
```

Test 2: Nesmyslná syntaxe → Detekce chyby.

```
./proj2 --tan 5 1 10    ; uhol nieje z intervalu <0,1.4>
./proj2 --tan 0.165461 1 52 ; maximálny počet iterácií je 13
./proj2 -c 233 -m 1.2    ; maximálna výška 100
```

Test 3: Porovnanie výpočtov → Predpokladaný výstup.

```
./proj2 --tan 1.024 10 10
10 1.642829e+00 1.642552e+00 2.773337e-04 1.642829e+00 0.000000e+00

./proj2 --tan 0.785398163 10 10
10 1.000000e+00 9.999992e-01 8.095039e-07 1.000000e+00 1.110223e-16
```

Test 4: Výpočet vzdialenosti a výšky → Predpokladaný výstup.

```
./proj2 -m 0.3
4.8490922156e+00
7.6106234032e+00

./proj2 -m 0.3 0.9
4.8490922156e+00
7.6106234032e+00

./proj2 -c 1.7 -m 0.15 1.3
1.1248205560e+01
4.2217188781e+01
```

Kapitola 5

Popis riešenia

5.1 Analýza vstupných parametrov

Pre overenie správnosti zadania číselných hodnôt sú použité funkcie `strtod` a `strtol`, ktoré viedlo k zjednodušeniu kódu.

5.2 Ovládanie programu

Program obsahuje nápovedu, ktorú vyvoláme parametrom `-help`. Nápoveda obsahuje presné informácie o vstupných parametroch, ktoré určujú funkciu programu.

Všetky chybné výstupy sa zobrazujú do štandardného chybového výstupu, validné výstupy sa zobrazujú do štandardného výstupu.

5.3 Implementácia

Parametry z príkazového riadka spracováva funkcia `parse_args`, ktorá naplní štruktúru `params`. Následne podľa tejto štruktúry rozvetvíme program na dve časti, jedna z nich spracováva parameter `-tan` a druhá `-m`.

Pri výpočtoch odchýlky používame funkciu `show_tan_diff_table`, ktorá postupne iteruje a zobrazuje výsledky na štandardný výstup. Implementáciu Tayloroveho rozvoja nájdeme vo funkcii `taylor_tan` a metódu zret'azených zlomkov `cfrac_tan`.

Pre výpočet vzdialenosti používame funkciu `calculate_distance` a výsledok z nej zobrazíme na štandardný výstup. Pre výpočet výšky objektu je použitá funkcia `calculate_height`, predtým je však potreba overiť, či bol zadán aj úhol β , keďže je tento parameter voliteľný. Výsledok je následne vypísaný na štandardný výstup.

Ak v tomto priebehu nastane chyba, (napr. nevalidné hodnoty parametrov) nastavíme hodnotu premennej `err` a následne zavoláme funkciu `show_error_and_halt`, ktorá program ukončí s chybovou hláškou.

Kapitola 6

Záver

toto je zaver

Dodatok A

Metriky kódu

Počet súborov: 1 súbor

Počet riadkov zdrojového textu: 238 riadkov

Veľkosť statických dát: 312B

Veľkosť spustiteľného súbora: 9753B (Ubuntu x86, pri preklade bez ladiacich informácií)