Arytmetyka komputerowa

ćwiczenie 3

Opracowanie: Maksymilian Sulima

Wstęp

System operacyjny:

- Manjaro linux 22.0.4

Język:

- C++, kompilator g++ 12.2.1 20230201

Procesor:

- AMD Ryzen 7 4700U

Treść zadania

Wyznaczyć wartości funkcji f(x) = sqrt(x2 + 1) - 1, g(x) = x2 / (sqrt(x2 + 1) + 1), dla argumentu x = 8^(-1), 8^(-2), 8^(-3), ... Sprawdzić, czy wyznaczone wartości dla obu funkcji (matematycznie tożsamych) są takie same i spróbować uzasadnić ewentualne różnice.

Jak obliczać z kolei wartości dla dużych argumentów (np. x bliskiego największej liczbie typu double)?

Obliczenia wykonać dla zmiennych typu float, double, long double.

Kod

```
float f(float x) {
   return sqrt(x * x + 1) - 1;
}
```

```
float g(float x) {
   return (x * x) / (sqrt(x * x + 1) + 1);
}
```

Wyniki

Argument	f(x)	g(x)	różnica
0.125	0.00778222	0.00778222	2.32831e-09
0.015625	0.00012207	0.000122063	7.45058e-09
0.00195312	1.90735e-06	1.90735e-06	1.81899e-12
0.000244141	0	2.98023e-08	2.98023e-08
3.05176e-05	0	4.65661e-10	4.65661e-10
3.8147e-06	0	7.27596e-12	7.27596e-12

Tabela 1. Wyniki funkcji dla operacji na typie float

Argument	f(x)	g(x)	różnica
0.125	0.00778222	0.00778222	6.50521e-17
0.015625	0.000122063	0.000122063	8.32803e-17
0.00195312	1.90735e-06	1.90735e-06	3.46945e-18
0.000244141	2.98023e-08	2.98023e-08	1.32349e-23
3.05176e-05	4.65661e-10	4.65661e-10	1.0842e-19
3.8147e-06	7.27596e-12	7.27596e-12	2.64698e-23

Tabela 2. Wyniki funkcji dla operacji na typie double

Argument	f(x)	g(x)	różnica
0.125	0.00778222	0.00778222	5.2516e-20
0.015625	0.000122063	0.000122063	2.37169e-20
0.00195312	1.90735e-06	1.90735e-06	8.27181e-24
0.000244141	2.98023e-08	2.98023e-08	1.32349e-23
3.05176e-05	4.65661e-10	4.65661e-10	2.52435e-29
3.8147e-06	7.27596e-12	7.27596e-12	2.64698e-23

Tabela 3. Wyniki funkcji dla operacji na typie long double

Argument	Float	Double	Long double
0.125	2.32831e-09	6.50521e-17	5.2516e-20
0.015625	7.45058e-09	8.32803e-17	2.37169e-20
0.00195312	1.81899e-12	3.46945e-18	8.27181e-24
0.000244141	2.98023e-08	1.32349e-23	1.32349e-23
3.05176e-05	4.65661e-10	1.0842e-19	2.52435e-29
3.8147e-06	7.27596e-12	2.64698e-23	2.64698e-23

Tabela 4. Różnice w wynikach dla różnych typów

Wnioski

- Różnice w wynikach są związane z faktem dodawania 1, która jest liczbą o rzędy wielkości większą od argumentów podawanych do funkcji f i g.
- W szczególności uwidocznione jest to dla funkcji f, na floatach gdzie dla argumentów mniejszych lub równych 8^(-4) wynik jest już zaokrąglany do 0. Mała liczba jest gubiona w momencie dodania 1 pod pierwiastkiem.
- Funkcja g unika tego problemu pomimo tracenia tej wartości w mianowniku ze względu na brak dodawania 1 w liczniku.

Obliczenia dla dużych liczb

Problem:

- problem podnoszenia do kwadratu

```
cout << "duży double, x = 1.5e308\n";
double x = 1.5e308;
cout << f(x) << "\t" << g(x) << "\n";
```

```
duży double, x = 1.5e308
inf   -nan
```

Rozwiązanie należy unikać działań, które w znaczny sposób zwiększają duże liczby lub zmniejszą liczby małe aby to osiągnąć należy przekształcić odpowiednio wzór funkcji.

Np sqrt(x * y) => sqrt(x) * sqrt(y).

Jednak w przypadku funkcji danych w zadaniu nie udało mi się dokonać podobnego przekształcenia.

KONIEC