Metody obliczeniowe w nauce i technice
Labolatorium 1
Arytmetyka Komputerowa

Mateusz Kowalski

1. Wstęp

Na poprzednim laboratorium zajmowaliśmy się różnymi sposobami sumowania liczb zmiennoprzecinkowych, badaniem ich błędów oraz mierzeniem czasu ich wykonania. Wszystkie obliczenia zostały wykonane w C++ z wykorzystaniem zmiennych typu *float* oraz *double.*

2. Sumowanie liczb pojedynczej precyzji

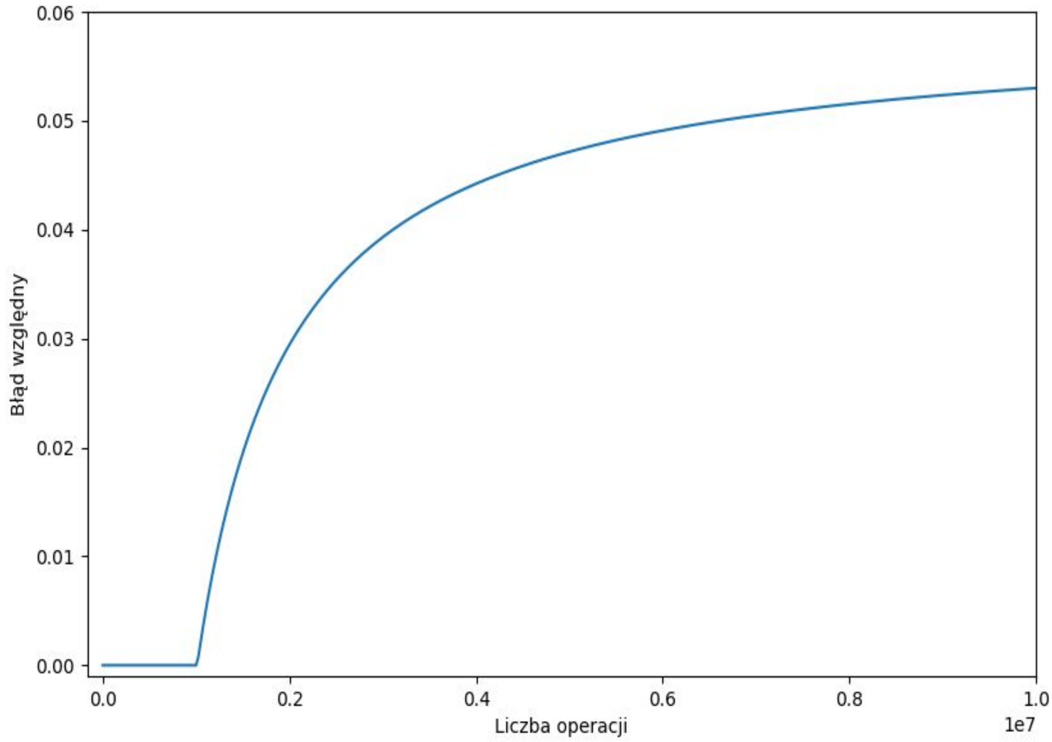
2.1. Sumowanie N pojedynczych liczb pojedynczej precyzji

Dla tablicy zmiennych typu *float* o wielkości N = 10^7 wypełnioną wartością V = 0.53125 suma tych wartości, przy dokładności naszego typu wynosi 5030840.5 kiedy prawdziwa wartość takiego działania wynosi 5312500.

Błąd bezwzględny: 281659.5
Błąd względny: 0.05301825882

Nasze błędy są stosunkowo duże, wynika to z użycia najprostszego algorytmu sumowania który przy każdej następnej operacji “gubi” kolejne realne wartości, przez co wynik operacji różni się od realnej wartości.

Dla zwizualizowania procesu rośnięcia błędu względnego, obserwujemy jego wartość co 25000 kroków, co później pozwala nam skonstruować odpowiedni wykres. Jak możemy zauważyć, przy początkowych krokach, nasz błąd jest bliski 0, lecz wraz ze wzrostem liczby operacji nasz błąd znacznie rośnie.



2.2. Rekurencyjny algorytm sumowania

Przy użyciu rekurencyjnego algorytmu sumowania otrzymujemy w większości przypadków zerowy błąd, co powoduje otrzymanie dokładnej sumy. Powodem tego jest to, że w każdym kroku dodajemy do siebie takie same wartości, przez to też suma jest stopniowo i stabilnie wyliczana.

Dla wartości V = 0.666666 błąd jest różny od zera

2.3. Algorytm Kahana

Dzięki temu algorytmowi, oba nasze błędy wynoszą 0. Zawdzięczamy to zmiennej *err*, która z każdym krokiem kompensuje nasz błąd operacji dodawania wartości zmiennoprzecinkowych.

3. Czasy działania

Poniżej przedstawione czasy działania algorytmów sumowania dla danych:

V = 0.53125
N = 10^7

Standardowy algorytm sumowania: 0.030293 s
Algorytm rekurencyjny: 0.057824 s
Algorytm Kahana: 0.080313 s

Jak możemy zauważyć, ceną za dokładne obliczenia jest czas realizacji zadania.

4. Sumy częściowe

Dla każdej kombinacji zmiennych S = 2, 3.6667, 5, 7.2, 10 oraz też N = 50, 100, 200, 500, 1000 przeprowadziłem obliczenie sum szeregu definiujących funkcje Riemanna oraz Dirichleta. Wszystkie operacje wykonane zostały sumując w przód jak i wstecz oraz z wykorzystaniem obu rodzajów zmiennych (pojedynczej i podwójnej precyzji).

Jak możemy zauważyć, liczba N jest decydująca w kwestii odchyłu od naszej realnej wartości. Im większa, tym nasze pomiary bardziej precyzyjne. Również kierunek w którym zliczamy naszą sumę też gra rolę, ponieważ podczas działania kumulowany jest inny błąd.