MOWNiT Laboratorium 1
Precyzja zmiennoprzecinkowa

Hubert Miklas

11-03-2025

1 Wprowadzenie

Laboratorium polega na wprowadzeniu do precyzji obliczeń w operacjach zmiennoprzecinkowych w komputerze. Wszystkie zadania zostały wykonane w języku Python 3.10.12.

2 Zadanie 1

Szukamy precyzji komputerowej ϵ, czyli najmniejszej liczby, dla której komputer uznaje $1 + \epsilon > 1$ za prawdziwe.

2.1 Kod implementujący obliczenia

```
epsilon = 1
const = 1
while const + epsilon > const:
epsilon /= 2
epsilon *= 2
print(epsilon)
```

To daje wynik $\epsilon \approx 2.2 \times 10^{-16}$, co jest oczekiwane dla liczb zmiennoprzecinkowych z podwójną precyzją (float64).

3 Zadanie 2

Rozważamy problem ewaluacji funkcji $\sin(x)$, przy czym występuje propagacja błędu danych wejściowych (zakłócenie h w argumencie x). Zakładamy $h = 10^{-5}$.

3.1 Błąd bezwzględny

$$\text{bb} = |\sin(x + h) - \sin(x)| \approx |\cos(x)|\, h.$$

Maksymalny błąd bezwzględny wynosi zatem około 10^{-5} (dla $|\cos(x)| = 1$).

3.2 Błąd względny

$$\text{bw} = \frac{|\sin(x + h) - \sin(x)|}{|\sin(x)|} \approx h\,|\cot(x)|.$$

Błąd względny rośnie, gdy $\sin(x)$ jest bliskie zeru, czyli dla $x = k\pi$, $k \in \mathbb{Z}$.

3.3 Uwarunkowanie problemu

$$\text{cond} \approx \left| \frac{x\cos(x)}{\sin(x)} \right| = |x\ \cot(x)|.$$

Problem jest bardzo czuły dla argumentów x bliskich wielokrotności π, gdzie $\sin(x) \approx 0$, natomiast lepiej uwarunkowany dla $x \approx \pi/2 + k\pi$. W szczególności, dla $x \to 0$ mamy

$$\lim_{x\to 0} \frac{\sin(x)}{x} = 1,$$

co zapewnia dobre uwarunkowanie w pobliżu zera.

3.4 Kod implementujący obliczenia

```
import math from sin, cos

def find_machine_epsilon():
    eps = 1.0
    while (1.0 + eps) > 1.0:
        eps /= 2
    return eps * 2

def absolute_error(x, h):
    return abs(sin(x + h) - sin(x))

def relative_error(x, h):
    if sin(x) == 0:
        return float('inf')  # unbounded error
    return abs(h * cos(x) / sin(x))

def condition_number(x):
    if sin(x) == 0:
        return float('inf')
    return abs(x * cos(x) / sin(x))

h = 1e-5  # małe zakłócenie

precision = 4

for x in test_values:
    abs_err = absolute_error(x, h)
    rel_err = relative_error(x, h)
    cond_num = condition_number(x)
    print(f"x = {x}")
    print(f"  Absolute error: {round(abs_err,precision)}")
    print(f"  Relative error: {round(rel_err,precision)}")
    print(f"  Condition number: {round(cond_num,precision)}")
```

3.5 Wyniki obliczeń

Poniżej przedstawiamy wartości błędów i liczby uwarunkowania dla wybranych wartości x:

x	Błąd bezwzględny	Błąd względny	Liczba uwarunkowania
0.1	9.95×10^{-6}	9.967×10^{-5}	0.9967
0.5	8.78×10^{-6}	1.83×10^{-5}	0.9152
1.0	5.40×10^{-6}	6.42×10^{-6}	0.6421

4 Zadanie 3

Funkcję $\sin(x)$ można rozwijać w szereg Maclaurina (szczególny przypadek szeregu Taylora):

$$\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots \tag{1}$$

4.1 Przybliżenie pierwszym składnikiem ($\sin(x) \approx x$)

Przyjmujemy $\hat{y} = x$. Wyznaczamy błędy:

$$\Delta y = \hat{y} - \sin(x), \quad \Delta x = \arcsin(\hat{y}) - x. \tag{2}$$

4.2 Kod implementujący obliczenia dla obu przypadków

```
from math import sin, asin

def progressive_error_approx1(x):
    return abs(sin(x) - x)

def progressive_error_approx2(x):
    return abs(sin(x) - (x - x**3 / 6))

def backward_error_approx1(x):
    return abs(x - asin(x))

def backward_error_approx2(x):
    return abs(x - asin(x - x**3 / 6))

def underflow_level(beta, L):
    return beta**L

test_values = [0.1, 0.5, 1.0]
precision = 4

print("Approximation 1")

for value in test_values:
    print(f"{value} & {round(backward_error_approx1(value),precision)} & {round(progressive_error_appro

print("Approximation 2")

for value in test_values:
    print(f"{value} & {round(backward_error_approx2(value),precision)} & {round(progressive_error_appro
```

4.3 Przybliżenie jednym składnikiem ($\sin(x) \approx x$)

Dla wybranych wartości x otrzymujemy:

x	Δy	Δx
0.1	-1.6658×10^{-4}	1.6742×10^{-4}
0.5	-2.0574×10^{-2}	2.3599×10^{-2}
1.0	-1.5853×10^{-1}	5.70796×10^{-1}

Table 1: Błędy progresywny i wsteczny dla przybliżenia $\sin(x) \approx x$

4.4 Przybliżenie dwoma składnikami ($\sin(x) \approx x - \frac{x^3}{6}$)

Przyjmujemy $\hat{y} = x - \frac{x^3}{6}$. Otrzymujemy:

$$\Delta y = \hat{y} - \sin(x), \quad \Delta x = \arcsin(\hat{y}) - x. \tag{3}$$

Dla wybranych wartości x otrzymujemy:

x	Δy	Δx
0.1	8.33×10^{-8}	-8.37×10^{-8}
0.5	2.5887×10^{-4}	-2.9496×10^{-4}
1.0	8.1377×10^{-3}	-1.4889×10^{-2}

Table 2: Błędy progresywny i wsteczny dla przybliżenia $\sin(x) \approx x - \frac{x^3}{6}$

5 Zadanie 4

Zakładamy, że mamy znormalizowany system zmiennoprzecinkowy o parametrach: $\beta = 10$, $p = 3$, $L = -98$.

Poziom niedomiaru (ang. Underflow level - UFL): Najmniejsza dodatnia liczba znormalizowana ma postać

$$\text{UFL} = 1.00 \times 10^{-98}.$$

Operacja: Dla $x = 6.87 \times 10^{-97}$ oraz $y = 6.81 \times 10^{-97}$:

$$x - y = (6.87 - 6.81) \times 10^{-97} = 0.06 \times 10^{-97} = 6.0 \times 10^{-99}.$$

Ponieważ $6.0 \times 10^{-99} < \text{UFL}$, wynik operacji ulega underflow i jest reprezentowany jako 0.