Sprawozdanie Obliczenia Naukowe Lista 1

Kamil Włodarski, 261739

23 października 2022

1 Zadanie 1

1.1 wstęp

Epsilonem maszynowym *macheps* (ang. machine epsilon) nazywamy najmniejszą liczbę $macheps > 0$ taką, że $fl(1.0 + macheps) > 1.0$ i $fl(1.0 + macheps) = 1 + macheps$. Innymi słowy macheps jest odległością 1.0 od kolejnej liczby $x, x > 1$, reprezentowanej w arytmetyce zmiennopozycyjnej (kolejnej liczby maszynowej). Liczbą maszynową *eta* nazywamy liczbę maszynową taką, że $eta > 0.0$.

1.2 wyniki

źródło	Float 16	Float 32	Float 64
Obliczone	0.000977	1.1920929e-7	2.220446049250313e-16
Julia	0.000977	1.1920929e-7	2.220446049250313e-16
C	—	1.1920929e-07	2.2204460492503131e-16

Tabela 1: Epsilon maszynowy.

źródło	Float 16	Float 32	Float 64
Obliczone	6.0e-8	1.0e-45	5.0e-324
Julia	6.0e-8	1.0e-45	5.0e-324

Tabela 2: Liczba Eta.

Float 32	Float 64
1.1754944e-38	2.2250738585072014e-308

Tabela 3: Wyniki floatmin().

źródło	Float 16	Float 32	Float 64
Obliczone	6.55e4	3.4028235e38	1.7976931348623157e308
Julia	6.55e4	3.4028235e38	1.7976931348623157e308
Z raportu	–	3.4e38	1.8e308

Tabela 4: Wartości floatmax.

1.3 wnioski

Obliczony epsilon maszynowy (tabela 1) pokrywa się z epsilonem maszynowym zawartym w bibliotekach standardowych julii i c (za wyjątkiem float 16 w przypadku c który nie jest obsługiwany w moim systemie).

Liczba *eta* wyliczona przeze mnie (tabela 2) odpowiada wartością wskazywanym przez julię z po użyciu funkcji: **nextfloat(Float16(0.0)), nextfloat(Float32(0.0)), nextfloat(Float64(0.0))**.

Epsilon maszynowy to podwojona wartość precyzji arytmetyki.

Liczba eta odpowiada liczbie MIN_{sub} omawianej na wykładzie czyli najmniejszej zdenormalizowanej liczbie dodatniej reprezentowanej w danej arytmetyce zmiennopozycyjnej.

Wartości zwracane przez funkcję **floatmin(Float32), floatmin(Float64)** (tabela 3) odpowiadają wartości MIN_{nor} omawianej na wykładzie czyli najmniejszej dodatniej liczbie znormalizowanej reprezentowanej w danej arytmetyce zmiennopozycyjnej.

Liczba MAX wyliczona przeze mnie (tabela 4) odpowiada wartością z biblioteki standardowej julii oraz z grubsza odpowiada wartością teoretycznym zawartym w raporcie dołączonym do listy zadań.

2 Zadanie 2

2.1 Wstęp

Kahan stwierdził, że epsilon maszynowy można otrzymać obliczając wyrażenie $3 * (\frac{4}{3} - 1) - 1$. Celem zadania jest sprawdzenie eksperymentalnie tego twierdzenia.

2.2 Wyniki

źródło	Float 16	Float 32	Float 64
Kahan	-0.000977	1.1920929e-7	-2.220446049250313e-16
julia	0.000977	1.1920929e-7	2.220446049250313e-16

Tabela 5: Porównanie wartości *macheps* z przybliżeniem otrzymanym metodą Kahan'a

2.3 Wnioski

Możemy zauważyć, że wartości otrzymane tą metodą są zgodne z wartościami z biblioteki standardowej julii z dokładnością do znaku. Przeciwny znak w przypadku 16 i 64 bitów może wynikać z parzystej ilości bitów mantysy oraz z tego że rozwinięciem ułamka $\frac{4}{3}$ jest 1.(10).

3 Zadanie 3

3.1 Wstęp

Sprawdź eksperymentalnie że w arytmetyce Float64 liczby zmiennopozycyjne są równomiernie rozmieszczone w przedziale [1,2]. Jak rozmieszczone są liczby zmiennopozycyjne w przedziale [$\frac{1}{2}$,1], jak w przedziale[2,4] i jak mogą być przedstawione dla rozpatrywanego przedziału?

3.2 Rozwiązanie

Do rozwiązania tego zadnia możemy albo wykorzystać metodę manualną (brute-force) lub spojrzeć na eksponentę reprezentacji bitowej liczb z przedziału. Jeśli najmniejsza i największa liczba z przedziału mają taka samą eksponentę to znaczy że wszystkie liczby w przedziale są rozłożone równomiernie.

3.3 Wnioski

Nasze sprawdzenie potwierdza że liczby w przedziale [1, 2] są rozłożone równomiernie ze skokiem 2^{-52}. W pozostałych przedziałach liczby również są rozłożone równomiernie ze skokami odpowiednio $1.110e - 16$ dla $[\frac{1}{2}, 1]$ oraz $4.441e - 16$ dla przedziału [2, 4].

4 Zadanie 4

4.1 Wstęp

Szukamy liczby x takiej że $x * (\frac{1}{x}) \neq 1$. Znajdziemy od razu najmniejszą taką liczbę idąc od dołu przedziału czyli od 1.

4.2 Wyniki

Znaleziona przeze mnie liczba to: 1.000000057228997.

4.3 Wnioski

Jak możemy zauważyć liczba taka istnieje co świadczy o tym że należy być ostrożnym wykonując działania w arytmetyce zmiennopozycyjnej gdyż każde działanie może wprowadzić błąd nawet gdy w zwykłej arytmetyce nie miało by one żadnego wpływu na wynik.

5 Zadanie 5

5.1 Wstęp

Zadanie to polega na mnożeniu dwóch wektorów x i y różnymi algorytmami i porównanie otrzymanych wyników.

5.2 Wyniki

algorytm	Float 32	Float 64
a)	-0.4999443	1.0251881368296672e-10
b)	-0.4543457	-1.5643308870494366e-10
c)	-0.5	0.0
d)	-0.5	0.0
biblioteka	-0.34720382	1.0251881368296672e-10

Tabela 6: Porównanie wyników otrzymanych różnymi algorytmami

5.3 Wnioski

Poprawna wartość wynosi $1.00657107000000 \cdot 10^{-11}$. Żaden z algorytmów nie oblicza dokładnie wartości. Dodatkowo sprawdziłem obliczenia wykonywane przez bibliotekę LinearAlgebra w julii która otrzymała wyniki lepsze dla 32 bitowej precyzji (możliwe że w procesie obliczeniowym wykorzystywała rozszerzenie do 64 bitowej precyzji) natomiast takie same jak algorytm a dla Float 64.

6 Zadanie 6

6.1 Wstęp

Zadanie polega na porównaniu dwóch funkcji $g(x)$ i $f(x)$ które są sobie równe w sensie matematycznym ale wykonują różne działania podczas obliczeń na komputerze.

6.2 Wyniki

patrz tabela 7.

6.3 Wnioski

Obie funkcje dają zbliżone do siebie wyniki natomiast jednak funkcja f bardzo szybko zbiega do 0. Funkcja g radzi więc sobie lepiej podając dokładniejsze wyniki.

7 Zadanie 7

7.1 Wstęp

W zadaniu 7 wyliczamy przybliżoną wartość pochodnej w sposób numeryczny i porównujemy ją z wartością oczekiwaną.

7.2 Wyniki

patrz tabela 8.

7.3 Wnioski

Możemy zauważyć że najlepsze przybliżenie otrzymujemy dla h = 1.4901161193847656e-8 a więc nie dla minimalnej wielkości h. Wynika to z tego że wraz z tym jak maleje h maleje też liczba cyfr znaczących w zapisie liczby zmiennoprzecinkowej co wpływa na to że obliczenia wykonywane na takich liczbach szybko tracą precyzję.

x	$f(x)$	$g(x)$
0.125	0.0077822185373186414	0.0077822185373187065
0.015625	0.00012206286282867573	0.00012206286282875901
0.001953125	1.9073468138230965e-6	1.907346813826566e-6
0.000244140625	2.9802321943606103e-8	2.9802321943606116e-8
3.0517578125e-5	4.656612873077393e-10	4.6566128719931904e-10
3.814697265625e-6	7.275957614183426e-12	7.275957614156956e-12
4.76837158203125e-7	1.1368683772161603e-13	1.1368683772160957e-13
5.960464477539063e-8	1.7763568394002505e-15	1.7763568394002489e-15
7.450580596923828e-9	0.0	2.7755575615628914e-17
9.313225746154785e-10	0.0	4.336808689942018e-19
1.1641532182693481e-10	0.0	6.776263578034403e-21
1.4551915228366852e-11	0.0	1.0587911840678754e-22
1.8189894035458565e-12	0.0	1.6543612251060553e-24
2.2737367544323206e-13	0.0	2.5849394142282115e-26
2.842170943040401e-14	0.0	4.0389678347315804e-28
3.552713678800501e-15	0.0	6.310887241768095e-30
4.440892098500626e-16	0.0	9.860761315262648e-32
5.551115123125783e-17	0.0	1.5407439555097887e-33
6.938893903907228e-18	0.0	2.407412430484045e-35
8.673617379884035e-19	0.0	3.76158192263132e-37
1.0842021724855044e-19	0.0	5.877471754111438e-39
1.3552527156068805e-20	0.0	9.183549615799121e-41
1.6940658945086007e-21	0.0	1.4349296274686127e-42
2.117582368135751e-22	0.0	2.2420775429197073e-44
2.6469779601696886e-23	0.0	3.503246160812043e-46
3.308722450212111e-24	0.0	5.473822126268817e-48
4.1359030627651384e-25	0.0	8.552847072295026e-50
5.169878828456423e-26	0.0	1.3363823550460978e-51
6.462348535570529e-27	0.0	2.088097429759528e-53
8.077935669463161e-28	0.0	3.2626522339992623e-55
1.0097419586828951e-28	0.0	5.0978941156238473e-57
1.262177448353619e-29	0.0	7.965459555662261e-59
1.5777218104420236e-30	0.0	1.2446030555722283e-60
1.9721522630525295e-31	0.0	1.9446922743316068e-62
2.465190328815662e-32	0.0	3.0385816786431356e-64
3.0814879110195774e-33	0.0	4.7477838728798994e-66
3.851859888774472e-34	0.0	7.418412301374843e-68
4.81482486096809e-35	0.0	1.1591269220898192e-69
6.018531076210112e-36	0.0	1.8111358157653425e-71
7.52316384526264e-37	0.0	2.8298997121333476e-73
9.4039548065783e-38	0.0	4.421718300208356e-75
1.1754943508222875e-38	0.0	6.908934844075556e-77
1.4693679385278594e-39	0.0	1.0795210693868056e-78
1.8367099231598242e-40	0.0	1.6867516709168837e-80
2.2958874039497803e-41	0.0	2.635549485807631e-82
2.8698592549372254e-42	0.0	4.118046071574423e-84
3.587324068671532e-43	0.0	6.434446986835036e-86
4.484155085839415e-44	0.0	1.0053823416929744e-87
5.605193857299268e-45	0.0	1.5709099088952725e-89
7.006492321624085e-46	0.0	2.4545467326488633e-91
8.758115402030107e-47	0.0	3.835229269763849e-93
1.0947644252537633e-47	0.0	5.992545734006014e-95

Tabela 7: Porównanie funkcji f i g.

h	pochodna	różnica
1.0	2.0179892252685967	1.9010469435800585
0.5	1.8704413979316472	1.753499116243109
0.25	1.1077870952342974	0.9908448135457593
0.125	0.6232412792975817	0.5062989976090435
0.0625	0.3704000662035192	0.253457784514981
0.03125	0.24344307439754687	0.1265007927090087
0.015625	0.18009756330732785	0.0631552816187897
0.0078125	0.1484913953710958	0.03154911368255764
0.00390625	0.1327091142805159	0.015766832591977753
0.001953125	0.1248236929407085	0.007881411252170345
0.0009765625	0.12088247681106168	0.0039401951225235265
0.00048828125	0.11891225046883847	0.001969968780300313
0.000244140625	0.11792723373901026	0.0009849520504721099
0.0001220703125	0.11743474961076572	0.0004924679222275685
6.103515625e-5	0.11718851362093119	0.0002462319323930373
3.0517578125e-5	0.11706539714577957	0.00012311545724141837
1.52587890625e-5	0.11700383928837255	6.155759983439424e-5
7.62939453125e-6	0.11697306045971345	3.077877117529937e-5
3.814697265625e-6	0.11695767106721178	1.5389378673624776e-5
1.9073486328125e-6	0.11694997636368498	7.694675146829866e-6
9.5367431640625e-7	0.11694612901192158	3.8473233834324105e-6
4.76837158203125e-7	0.1169442052487284	1.9235601902423127e-6
2.384185791015625e-7	0.11694324295967817	9.612711400208696e-7
1.1920928955078125e-7	0.11694276239722967	4.807086915192826e-7
5.960464477539063e-8	0.11694252118468285	2.394961446938737e-7
2.9802322387695312e-8	0.116942398250103	1.1656156484463054e-7
1.4901161193847656e-8	0.11694233864545822	5.6956920069239914e-8
7.450580596923828e-9	0.11694231629371643	3.460517827846843e-8
3.725290298461914e-9	0.11694228649139404	4.802855890773117e-9
1.862645149230957e-9	0.11694222688674927	5.480178888461751e-8
9.313225746154785e-10	0.11694216728210449	1.1440643366000813e-7
4.656612873077393e-10	0.11694216728210449	1.1440643366000813e-7
2.3283064365386963e-10	0.11694192886352539	3.5282501276157063e-7
1.1641532182693481e-10	0.11694145202636719	8.296621709646956e-7
5.820766091346741e-11	0.11694145202636719	8.296621709646956e-7
2.9103830456733704e-11	0.11693954467773438	2.7370108037771956e-6
1.4551915228366852e-11	0.116943359375	1.0776864618478044e-6
7.275957614183426e-12	0.1169281005859375	1.4181102600652196e-5
3.637978807091713e-12	0.116943359375	1.0776864618478044e-6
1.8189894035458565e-12	0.11688232421875	5.9957469788152196e-5
9.094947017729282e-13	0.1168212890625	0.0001209926260381522
4.547473508864641e-13	0.116943359375	1.0776864618478044e-6
2.2737367544323206e-13	0.11669921875	0.0002430629385381522
1.1368683772161603e-13	0.1162109375	0.0007313441885381522
5.684341886080802e-14	0.1171875	0.0002452183114618478
2.842170943040401e-14	0.11328125	0.003661031688538152
1.4210854715202004e-14	0.109375	0.007567281688538152
7.105427357601002e-15	0.109375	0.007567281688538152
3.552713678800501e-15	0.09375	0.023192281688538152
1.7763568394002505e-15	0.125	0.008057718311461848
8.881784197001252e-16	0.0	0.11694228168853815
4.440892098500626e-16	0.0	0.11694228168853815

Tabela 8: Wyniki zadanie 7