

Obliczenia naukowe

Sprawozdanie 2

Józef Piechaczek

7 listopada 2019

1 Zadanie 1

Zadanie 1 polega na ponownym przeprowadzeniu eksperymentu z listy 1 dla zadania 5. Zadanie to polega na obliczeniu iloczynu skalarnego dwóch wektorów, przy czym obcięte są dwie ostatnie cyfry przy wartości x_4 oraz x_5 . Wektory po obcięciu wyglądają następująco:

$$x = [2.718281828, -3.141592654, 1.414213562, 0.577215664, 0.301029995]$$

$$y = [1486.2497, 878366.9879, -22.37492, 4773714.647, 0.000185049]$$

Iloczyn należy policzyć za pomocą czterech algorytmów

1. w przód
2. w tył
3. od największego do najmniejszego
4. od najmniejszego do największego

Wyniki eksperymentu przedstawia poniższa tabela:

	Float32		Float64	
	$x_1 \cdot y$	$x_2 \cdot y$	$x_1 \cdot y$	$x_2 \cdot y$
1	-0.4999443	-0.4999443	1.0251881368296672e-10	-0.004296342739891585
2	-0.4543457	-0.4543457	-1.5643308870494366e-10	-0.004296342998713953
3	-0.5	-0.5	0.0	-0.004296342842280865
4	-0.5	-0.5	0.0	-0.004296342842280865

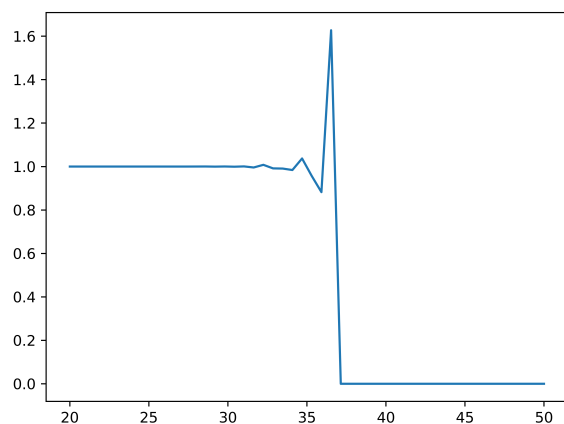
Wnioski:

Patrząc na tabelę możemy zauważyć, że wartości dla **Float32** nie uległy zmianie dla żadnego z algorytmów. Wynika to z faktu iż liczba pojedynczej precyzji zawiera około 7-8 cyfr znaczących, zatem zmiana na dziesiątej pozycji nie wpływa na wynik. Dla **Float64** możemy zauważyć znaczącą zmianę. Po pierwsze algorytmy **3** i **4** nie zwracają już wartości 0, która powiązana jest z prostopadłością wektorów. Możemy również dostrzec, że wyniki dla wszystkich czterech algorytmów są mocno zbliżone (dla **3** i **4** identyczne). Sugeruje to, że uzyskany wynik jest bliski poprawnego wyniku. Znacząca zmiana wyników jednak sugeruje, że zadanie obliczania iloczynu skalarnego jest źle uwarunkowane. W takim wypadku aby uzyskać dokładny wynik należy skorzystać z arytmetyki o jak najwyższej precyzji.

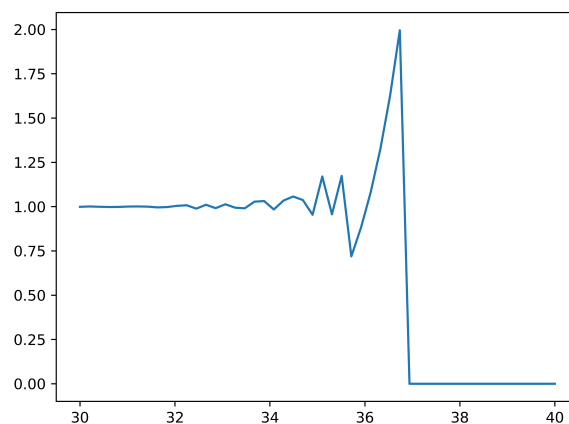
2 Zadanie 2

Zadanie 2 polega na narysowaniu wykresu funkcji $f(x) = e^x * \ln(1 + e^{-x})$ w co najmniej dwóch dowolnych programach do wizualizacji oraz na policzeniu granicy $\lim_{x \rightarrow \infty} f(x)$.

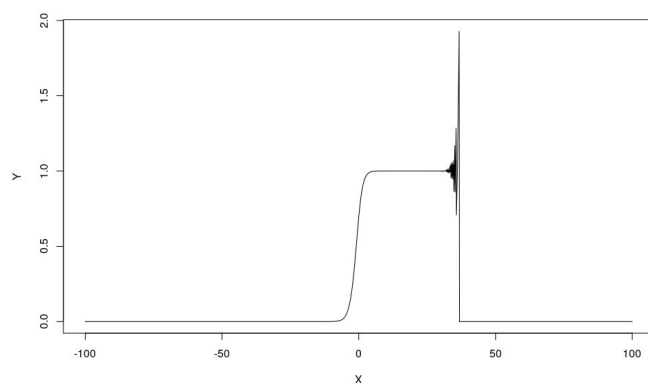
Wykresy:



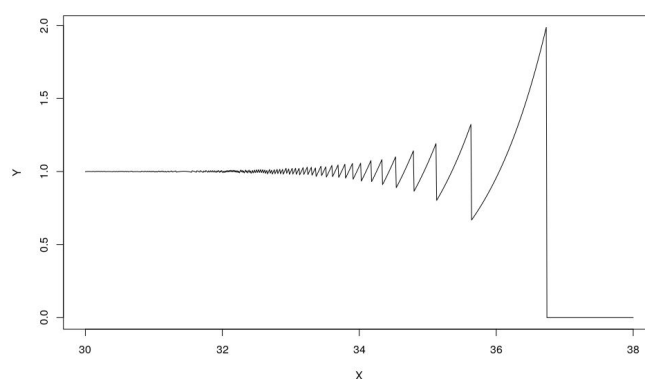
Rysunek 1: PyPlot od 20 do 50



Rysunek 2: PyPlot od 30 do 40



Rysunek 3: R od -100 do 100



Rysunek 4: R od 30 do 36

Granica:

$$\begin{aligned} \lim_{x \rightarrow \infty} e^x \ln(1 + e^{-x}) &= \lim_{x \rightarrow \infty} \frac{\ln(1 + e^{-x})}{e^{-x}} \stackrel{\text{z l'Hospitala}}{=} \lim_{x \rightarrow \infty} \frac{-\frac{1}{e^x + 1}}{-e^{-x}} \\ &= \lim_{x \rightarrow \infty} \frac{1}{(e^x + 1)(e^{-x})} = \lim_{x \rightarrow \infty} \frac{1}{1 + e^{-x}} = 1 \end{aligned}$$

Wnioski:

Po obliczeniu granicy dla funkcji f wiemy, że wykres powinien dążyć do wartości 1. Patrząc na otrzymane wykresy możemy zauważyć następujące obserwacje. (1) Dla około $x = 32$ możemy zobaczyć pojawiające się zaburzenie, które szybko zmienia wartość, a następnie wykres przyjmuje już do końca wartość 0. (2) Przy próbie wygenerowania wartości dla dużych wartości x , przy około $x = 710$ w bibliotece PyPlot pojawił się błąd `overflow encountered in power` oraz `invalid value encountered in multiply`. Obserwację 1 można wyjaśnić patrząc na tabelę wartości e^{-x} oraz $1 + e^{-x}$:

n	e^{-x}	$1 + e^{-x}$
30	9.35762297e-14	1.0000000000000935e+00
31	3.44247711e-14	1.0000000000000344e+00
32	1.26641655e-14	1.0000000000000127e+00
33	4.65888615e-15	1.0000000000000047e+00
34	1.71390843e-15	1.0000000000000018e+00
35	6.30511676e-16	1.0000000000000007e+00
36	2.31952283e-16	1.0000000000000002e+00
37	8.53304763e-17	1.0000000000000000e+00

Dla rosnącego x wartość wyrażenia e^{-x} zbliża się do wartości 0. Wyrażenie to po dodaniu do wartości 1 traci coraz to więcej informacji, a pojawiający się błąd rośnie również wraz z wzrostem wartości e^x . W momencie gdy 1 całkowicie "pochlania" wartość wyrażenia, logarytm osiąga wartość 0, czego skutkiem jest wartość 0 dla $x < 710$. Aby dowiedzieć się co się dzieje dla wartości $x = 710$ oraz wyższych posłuże się następującą tabelą:

n	e^x
707	1.11224050e+307
708	3.02338314e+307
709	8.21840746e+307
710	Inf
711	Inf

Dla odpowiednio wysokich x wartość wyrażenia e^x jest na tyle duża, że wykracza poza wartość maksymalną dla arytmetyki. Wyrażenie wtedy przyjmuje postać $\infty \cdot 0$ co sprawia że nie da się go policzyć.

3 Zadanie 3

Zadanie 3 polega na rozwiązaniu układu równań liniowych $Ax = b$ dla danej macierzy współczynników $A \in \mathbf{R}^{n \times n}$ i wektora prawych stron $b \in \mathbf{R}^n$.

Macierz A generować w następujący sposób:

1. $A = H_n$, gdzie H_n jest macierzą Hilberta stopnia n
2. $A = R_n$, gdzie R_n jest losową macierzą stopnia n z zadanyim wskaźnikiem uwarunkowania c

Wektor b zadany jest przez $b = Ax$, gdzie A jest wygenerowaną macierzą, a $x = (1, \dots, 1)^T$

Układ równań należy rozwiązać za pomocą dwóch algorytmów:

1. eliminacji Gaussa $x = A \backslash b$
2. $x = A^{-1}b$

Eksperymenty należy wykonać dla macierzy Hilberta H_n z rosnącym stopniem $n > 1$ oraz dla macierzy losowej $R_n, n = 5, 10, 20$ z rosnącym wskaźnikiem uwarunkowania $c = 1, 10, 10^3, 10^7, 10^{12}, 10^{16}$. Obliczony \hat{x} należy porównać z rozwiązaniem dokładnym $x = (1, \dots, 1)^T$ tj. policzyć błędy względne.

Wyniki przedstawiają następujące tabele:

			Błąd względny	
Size	Rank	Cond	Gauss	Inversion
2x2	2	1.928147e+01	5.661049e-16	1.404333e-15
3x3	3	5.240568e+02	8.022594e-15	0.000000e+00
4x4	4	1.551374e+04	4.137410e-14	0.000000e+00
5x5	5	4.766073e+05	1.682843e-12	3.354436e-12
6x6	6	1.495106e+07	2.618913e-10	2.016376e-10
7x7	7	4.753674e+08	1.260687e-08	4.713280e-09
8x8	8	1.525758e+10	6.124090e-08	3.077484e-07
9x9	9	4.931538e+11	3.875163e-06	4.541268e-06
10x10	10	1.602442e+13	8.670390e-05	2.501493e-04
11x11	10	5.222678e+14	1.582781e-04	7.618304e-03
12x12	11	1.751473e+16	1.339621e-01	2.589941e-01
13x13	11	3.344143e+18	1.103970e-01	5.331276e+00
14x14	11	6.200786e+17	1.455409e+00	8.714993e+00
15x15	12	3.674393e+17	4.696668e+00	7.344641e+00
16x16	12	7.865468e+17	5.415519e+01	2.984884e+01
17x17	12	1.263684e+18	1.370724e+01	1.051694e+01
18x18	12	2.244631e+18	9.134135e+00	7.575476e+00
19x19	13	6.471954e+18	9.720590e+00	1.223376e+01
20x20	13	1.355366e+18	7.549915e+00	2.206270e+01

Tabela 1: Wartości dla macierzy Hilberta

			Błąd względny	
Size	Rank	Cond	Gauss	Inversion
5x5	5	1	1.719950e-16	1.216188e-16
10x10	10	1	1.683737e-16	2.482534e-16
20x20	20	1	6.030055e-16	3.358310e-16
5x5	5	10	7.021667e-17	2.275280e-16
10x10	10	10	5.790215e-16	5.135907e-16
20x20	20	10	7.739776e-16	5.628294e-16
5x5	5	1e3	5.248902e-15	7.399118e-15
10x10	10	1e3	2.742085e-14	2.134402e-14
20x20	20	1e3	2.057420e-14	2.385714e-14
5x5	5	1e7	4.948064e-10	2.472969e-10
10x10	10	1e7	3.051511e-10	2.871953e-10
20x20	20	1e7	2.145808e-10	2.235508e-10
5x5	5	1e12	1.729134e-05	1.722960e-05
10x10	10	1e12	3.243992e-05	3.472999e-05
20x20	20	1e12	1.166862e-06	3.780269e-06
5x5	4	1e16	2.092372e-01	2.078771e-01
10x10	9	1e16	1.228583e+00	1.310098e+00
20x20	19	1e16	4.168338e-01	4.414028e-01

Tabela 2: Wartości dla losowej macierzy

Wnioski:

Patrząc na wyniki z zadania 3 możemy zaobserwować, że błąd względny bezpośrednio zależy od wskaźnika uwarunkowania macierzy. Patrząc na przykładowe macierze Hilberta oraz losowe o podobnym współczynniku uwarunkowania, możemy zauważyć, że błędy względne są podobnego rzędu. Macierz Hilberta jest bardzo źle uwarunkowana, co skutkuje faktem, że obliczenia z nią wykonane obarczone są dużym błędem. Patrząc na wyniki możemy również zauważyć, że metoda Gaussa w większości przypadków sprawdza się lepiej niż metoda inwersji.

4 Zadanie 4

Zadanie 4 polega na obliczeniu 20 zer wielomianu Wilkinsona $p = (x - 20)(x - 19)\dots(x - 1)$ w postaci naturalnej P i sprawdzeniu otrzymanych wyników z_k obliczając wartości $|P(z_k)|$, $|p(z_k)|$ i $|z_k - k|$ dla $1 \leq k \leq 20$. Eksperyment należy powtórzyć po zaburzeniu współczynnika -210 na $-210 - 2^{-23}$. W celu rozwiązania zadanie należy użyć pakietu **Polynomials**.

Eksperyment 1

Otrzymane wyniki:

k	z_k	$P(z_k)$	$p(z_k)$	$ z_k - k $
1	1.000000e+00	3.635200e+04	5.517824e+06	3.010925e-13
2	2.000000e+00	1.817600e+05	7.378698e+19	2.831824e-11
3	3.000000e+00	2.094080e+05	3.320414e+20	4.079035e-10
4	4.000000e+00	3.106816e+06	8.854437e+20	1.626247e-08
5	5.000001e+00	2.411469e+07	1.844675e+21	6.657698e-07
6	5.999989e+00	1.201521e+08	3.320395e+21	1.075418e-05
7	7.000102e+00	4.803983e+08	5.423593e+21	1.020028e-04
8	7.999356e+00	1.682691e+09	8.262050e+21	6.441704e-04
9	9.002915e+00	4.465327e+09	1.196559e+22	2.915294e-03
10	9.990413e+00	1.270713e+10	1.655260e+22	9.586958e-03
11	1.102502e+01	3.575990e+10	2.247833e+22	2.502293e-02
12	1.195328e+01	7.216772e+10	2.886945e+22	4.671675e-02
13	1.307431e+01	2.157236e+11	3.807326e+22	7.431403e-02
14	1.391476e+01	3.653833e+11	4.612720e+22	8.524441e-02
15	1.507549e+01	6.139878e+11	5.901011e+22	7.549380e-02
16	1.594629e+01	1.555028e+12	7.010874e+22	5.371328e-02
17	1.702543e+01	3.777624e+12	8.568906e+22	2.542715e-02
18	1.799092e+01	7.199555e+12	1.014480e+23	9.078647e-03
19	1.900191e+01	1.027838e+13	1.199038e+23	1.909818e-03
20	1.999981e+01	2.746295e+13	1.401912e+23	1.907088e-04

Tabela 3: Wartości dla oryginalnego wielomianu

Eksperyment 2

k	z_k	$P(z_k)$	$p(z_k)$	$ z_k - k $
1	1.000000e+00 + 0.000000e+00i	2.099200e+04	3.012096e+06	1.643130e-13
2	2.000000e+00 + 0.000000e+00i	3.491840e+05	7.378698e+19	5.503731e-11
3	3.000000e+00 + 0.000000e+00i	2.221568e+06	3.320414e+20	3.396580e-09
4	4.000000e+00 + 0.000000e+00i	1.046784e+07	8.854438e+20	8.972436e-08
5	4.999999e+00 + 0.000000e+00i	3.946394e+07	1.844673e+21	1.426112e-06
6	6.000020e+00 + 0.000000e+00i	1.291484e+08	3.320450e+21	2.047667e-05
7	6.999602e+00 + 0.000000e+00i	3.881231e+08	5.422367e+21	3.979296e-04
8	8.007772e+00 + 0.000000e+00i	1.072547e+09	8.289400e+21	7.772029e-03
9	8.915816e+00 + 0.000000e+00i	3.065575e+09	1.160747e+22	8.418363e-02
10	1.009546e+01 + -6.449328e-01i	7.143114e+09	1.721289e+22	6.519587e-01
11	1.009546e+01 + 6.449328e-01i	7.143114e+09	1.721289e+22	1.110918e+00
12	1.179389e+01 + -1.652477e+00i	3.357756e+10	2.856840e+22	1.665281e+00
13	1.179389e+01 + 1.652477e+00i	3.357756e+10	2.856840e+22	2.045820e+00
14	1.399241e+01 + -2.518824e+00i	1.061206e+11	4.934647e+22	2.518836e+00
15	1.399241e+01 + 2.518824e+00i	1.061206e+11	4.934647e+22	2.712881e+00
16	1.673074e+01 + -2.812625e+00i	3.315103e+11	8.484695e+22	2.906002e+00
17	1.673074e+01 + 2.812625e+00i	3.315103e+11	8.484695e+22	2.825484e+00
18	1.950244e+01 + -1.940332e+00i	9.539425e+12	1.318195e+23	2.454021e+00
19	1.950244e+01 + 1.940332e+00i	9.539425e+12	1.318195e+23	2.004329e+00
20	2.084691e+01 + 0.000000e+00i	1.114454e+13	1.591108e+23	8.469102e-01

Tabela 4: Wartości dla zmodyfikowanego wielomianu

Wnioski:

W zadaniu 4 możemy zauważyć, że pierwiastki policzone w eksperymencie 1 są bardzo bliskie rzeczywistym wartościom. Po podstawieniu obliczonych pierwiastków do wielomianu uzyskujemy bardzo duże wartości, sięgające nawet rzędu 10^{23} . Wynika to z faktu, iż pierwiastki z niewielkim błędem obliczeń po podstawieniu do wielomianu są potęgowane i przemnażane przez ogromne wartości sięgające rzędu aż do $1e18$.

Dodatkowo po wprowadzeniu niewielkiej zmiany współczynnika wynoszącej 2^{-23} uzyskujemy zera zespolone w eksperymencie 2.

Wnioskiem z powyższych eksperymentów jest fakt, iż zadanie wyznaczania pierwiastków wielomianu jest źle uwarunkowane ze względu na zaburzenia współczynników.

5 Zadanie 5

W zadaniu 5 rozważamy równanie rekurencyjne przedstawiające model wzrostu populacji:

$$p_{n+1} := p_n + rp_n(1 - p_n)$$

gdzie r jest pewną daną stałą, $r(1 - p_n)$ jest czynnikiem wzrostu populacji, a p_0 jest wielkością populacji stanowiącą procent maksymalnej wielkości populacji dla danego stanu środowiska.

Należy przeprowadzić następujące eksperymenty:

1. Dla danych $p_0 = 0.01$ i $r = 3$ wykonać 40 iteracji wyrażenia, a następnie wykonać ponownie 40 iteracji wyrażenia z niewielką modyfikacją tj. wykonać 10 iteracji, zatrzymać, zastosować obcię-

cie wyniku odrzucając cyfry po trzecim miejscu po przecinku i kontynuować dalej obliczenia. Porównać otrzymane wyniki.

2. Dla danych $p_0 = 0.01$ i $r = 3$ wykonać 40 iteracji wyrażenia w arytmetyce **Float32** i **Float64**. Porównać otrzymane wyniki.

Wyniki przedstawiam w poniższych tabelach:

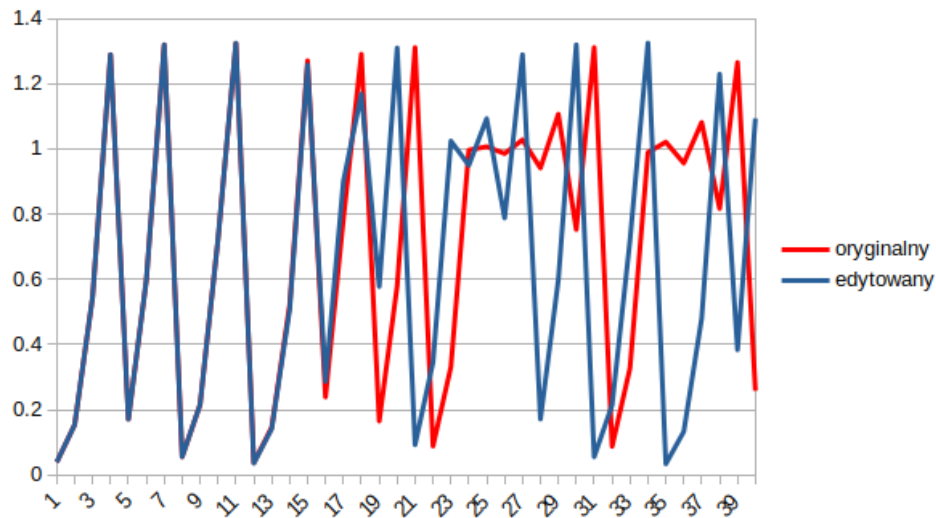
	p_n	
n	przed mod.	po mod.
1	0.0397	0.0397
2	0.15407173	0.15407173
3	0.5450726	0.5450726
4	1.2889781	1.2889781
5	0.1715188	0.1715188
10	0.7229306	0.722
15	1.2704837	1.2572169
20	0.5799036	1.3096911
25	1.0070806	1.0929108
30	0.7529209	1.3191822
35	1.021099	0.034241438
36	0.95646656	0.13344833
37	1.0813814	0.48036796
38	0.81736827	1.2292118
39	1.2652004	0.3839622
40	0.25860548	1.093568

Tabela 5: Wartości dla eksperymentu 1

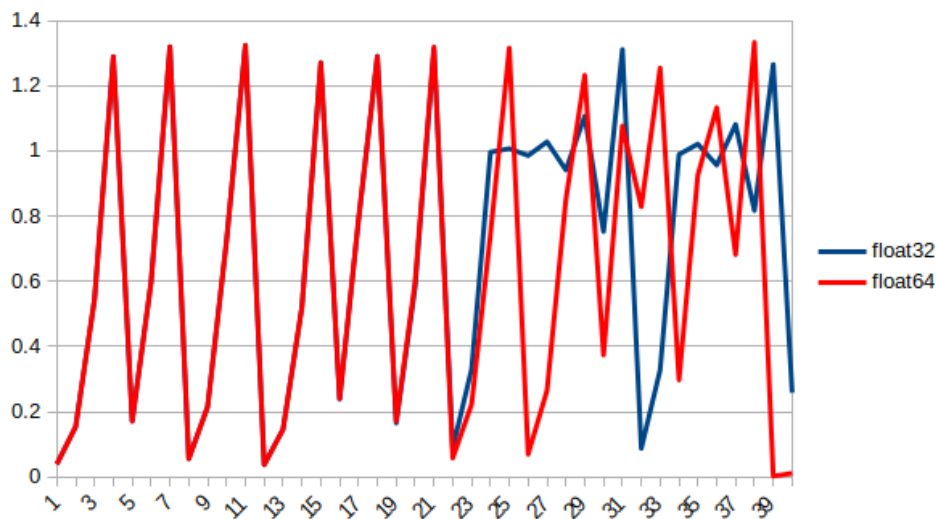
	p_n	
n	Float32	Float64
1	0.0397	0.0397
2	0.15407173	0.15407173000000002
3	0.5450726	0.5450726260444213
4	1.2889781	1.2889780011888006
5	0.1715188	0.17151914210917552
10	0.7229306	0.722914301179573
15	1.2704837	1.2702617739350768
20	0.5799036	0.5965293124946907
25	1.0070806	1.315588346001072
30	0.7529209	0.37414648963928676
35	1.021099	0.9253821285571046
36	0.95646656	1.1325322626697856
37	1.0813814	0.6822410727153098
38	0.81736827	1.3326056469620293
39	1.2652004	0.0029091569028512065
40	0.25860548	0.011611238029748606

Tabela 6: Wartości dla eksperymentu 2

Powyższe dane w formie wykresów:



Rysunek 5: Eksperyment 1



Rysunek 6: Eksperyment 2

Wnioski:

W zadaniu 5 możemy dostrzec, że wprowadzenie niewielkiej zmiany danych dla $n = 10$ spowodowało całkowitą rozbieżność danych w pozostałych iteracjach. Podobne zjawisko miało miejsce dla eksperymentu 2, gdzie od pewnej iteracji wartości dla Float32 i Float64 stały się całkowicie rozbieżne. Niewielkie błędy popełnione w początkowym studium procesu kumulują się w kolejnych stadiach, co pozwala nam uznać proces numeryczny za **niestabilny**. Zatem obliczone wartości od pewnego n są całkowicie niezgodne z rzeczywistymi. Aby uzyskać dokładniejsze wyniki możemy zwiększać precyzję arytmetyki, jednak dla odpowiednio dużych wartości n zawsze pojawiają się rozbieżności. Wynika to z

faktu, iż liczba z każdą iteracją zawiera coraz więcej cyfr znaczących, więc zawsze pojawiają się błędy zapisu, które następnie będą kumulowane.

6 Zadanie 6

W zadaniu 6 rozważamy równanie rekurencyjne:

$$x_{n+1} := x_n^2 + c$$

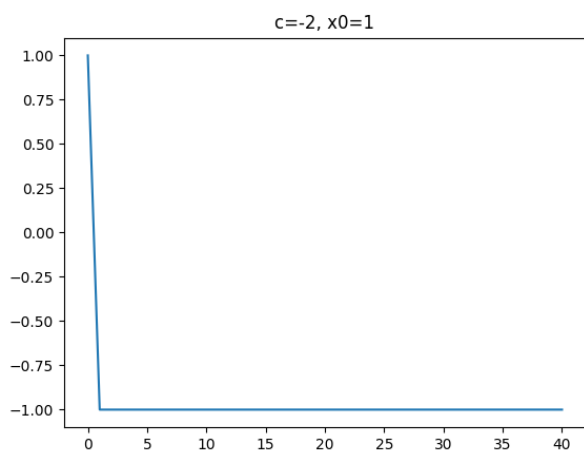
gdzie c jest pewną stałą.

Należy przeprowadzić następujące eksperymenty. Dla danych:

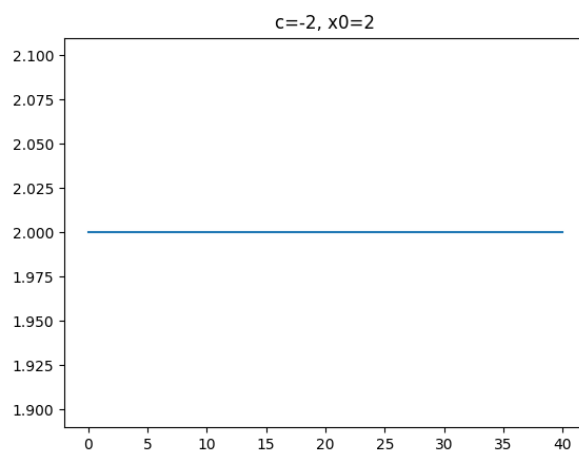
1. $c = -2$ i $x_0 = 1$
2. $c = -2$ i $x_0 = 2$
3. $c = -2$ i $x_0 = 1.9999999999999999$
4. $c = -1$ i $x_0 = 1$
5. $c = -1$ i $x_0 = -1$
6. $c = -1$ i $x_0 = 0.75$
7. $c = -1$ i $x_0 = 0.25$

wykonać w języku **Julia** w arytmetyce **Float64** 40 iteracji wyrażania oraz zaobserwować zachowanie generowanych ciągów.

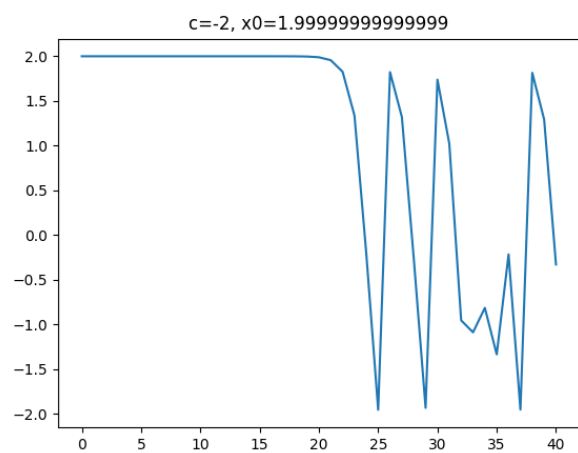
Uzyskane wartości przedstawiają następujące wykresy:



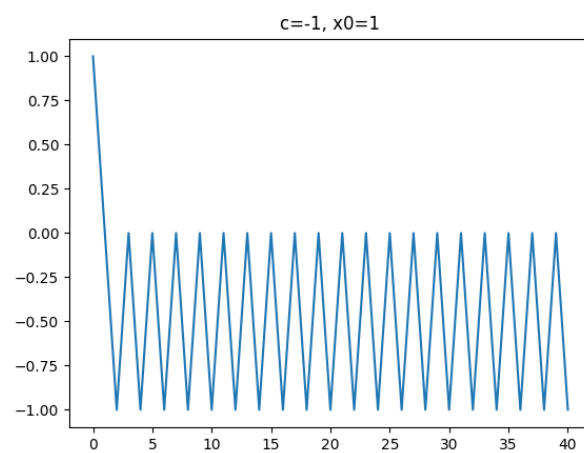
Rysunek 7: Eksperyment 1



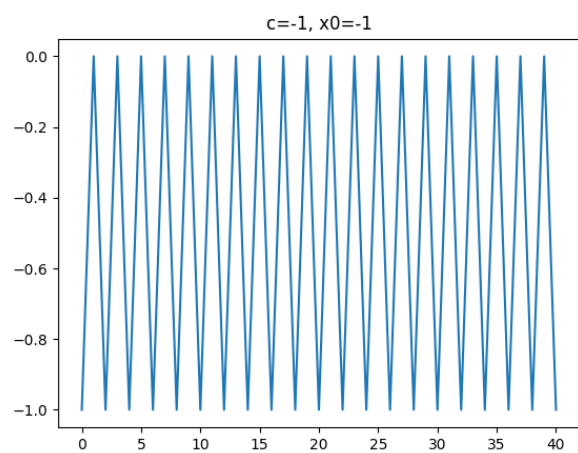
Rysunek 8: Eksperyment 2



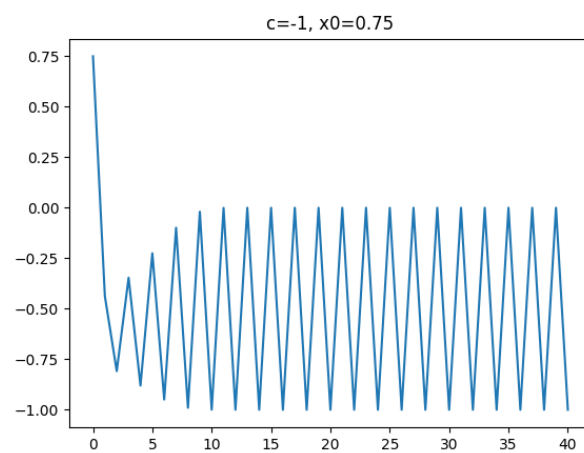
Rysunek 9: Eksperyment 3



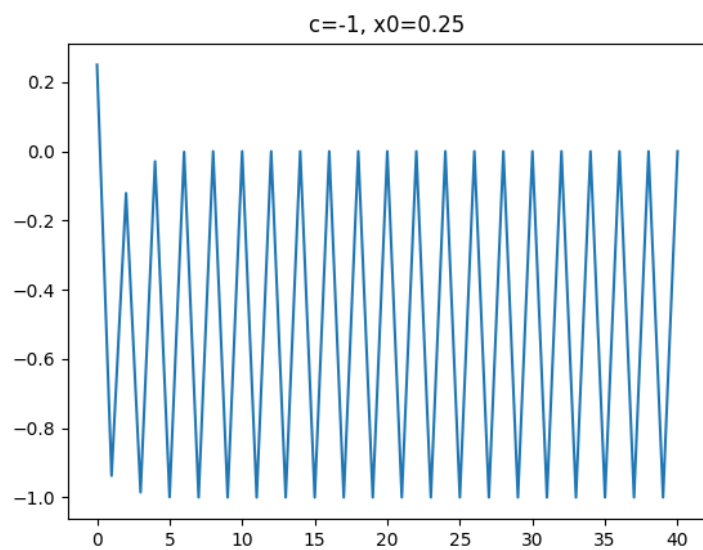
Rysunek 10: Eksperyment 4



Rysunek 11: Eksperyment 5



Rysunek 12: Eksperyment 6



Rysunek 13: Eksperyment 7

	$c = -2$			$c = -1$			
n	$x_0 = 1$	$x_0 = 2$	$x_0 = 1.99[...]$	$x_0 = 1$	$x_0 = -1$	$x_0 = 0.75$	$x_0 = 0.25$
0	1.0	2.0	2.000000e+00	1.0	-1.0	7.500000e-01	2.500000e-01
1	-1.0	2.0	2.000000e+00	0.0	0.0	-4.375000e-01	-9.375000e-01
2	-1.0	2.0	2.000000e+00	-1.0	-1.0	-8.085938e-01	-1.210938e-01
3	-1.0	2.0	2.000000e+00	0.0	0.0	-3.461761e-01	-9.853363e-01
4	-1.0	2.0	2.000000e+00	-1.0	-1.0	-8.801621e-01	-2.911237e-02
5	-1.0	2.0	2.000000e+00	0.0	0.0	-2.253147e-01	-9.991525e-01
6	-1.0	2.0	2.000000e+00	-1.0	-1.0	-9.492333e-01	-1.694342e-03
7	-1.0	2.0	2.000000e+00	0.0	0.0	-9.895619e-02	-9.999971e-01
8	-1.0	2.0	2.000000e+00	-1.0	-1.0	-9.902077e-01	-5.741579e-06
9	-1.0	2.0	2.000000e+00	0.0	0.0	-1.948876e-02	-1.000000e+00
10	-1.0	2.0	2.000000e+00	-1.0	-1.0	-9.996202e-01	-6.593148e-11
11	-1.0	2.0	2.000000e+00	0.0	0.0	-7.594796e-04	-1.000000e+00
12	-1.0	2.0	2.000000e+00	-1.0	-1.0	-9.999994e-01	0.000000e+00
13	-1.0	2.0	1.999999e+00	0.0	0.0	-1.153618e-06	-1.000000e+00
14	-1.0	2.0	1.999997e+00	-1.0	-1.0	-1.000000e+00	0.000000e+00
15	-1.0	2.0	1.999989e+00	0.0	0.0	-2.661649e-12	-1.000000e+00
16	-1.0	2.0	1.999957e+00	-1.0	-1.0	-1.000000e+00	0.000000e+00
17	-1.0	2.0	1.999828e+00	0.0	0.0	0.000000e+00	-1.000000e+00
18	-1.0	2.0	1.999313e+00	-1.0	-1.0	-1.000000e+00	0.000000e+00
19	-1.0	2.0	1.997254e+00	0.0	0.0	0.000000e+00	-1.000000e+00
20	-1.0	2.0	1.989024e+00	-1.0	-1.0	-1.000000e+00	0.000000e+00
21	-1.0	2.0	1.956215e+00	0.0	0.0	0.000000e+00	-1.000000e+00
22	-1.0	2.0	1.826779e+00	-1.0	-1.0	-1.000000e+00	0.000000e+00
23	-1.0	2.0	1.337120e+00	0.0	0.0	0.000000e+00	-1.000000e+00
24	-1.0	2.0	-2.121097e-01	-1.0	-1.0	-1.000000e+00	0.000000e+00
25	-1.0	2.0	-1.955009e+00	0.0	0.0	0.000000e+00	-1.000000e+00
26	-1.0	2.0	1.822062e+00	-1.0	-1.0	-1.000000e+00	0.000000e+00
27	-1.0	2.0	1.319910e+00	0.0	0.0	0.000000e+00	-1.000000e+00
28	-1.0	2.0	-2.578368e-01	-1.0	-1.0	-1.000000e+00	0.000000e+00
29	-1.0	2.0	-1.933520e+00	0.0	0.0	0.000000e+00	-1.000000e+00
30	-1.0	2.0	1.738500e+00	-1.0	-1.0	-1.000000e+00	0.000000e+00
31	-1.0	2.0	1.022383e+00	0.0	0.0	0.000000e+00	-1.000000e+00
32	-1.0	2.0	-9.547330e-01	-1.0	-1.0	-1.000000e+00	0.000000e+00
33	-1.0	2.0	-1.088485e+00	0.0	0.0	0.000000e+00	-1.000000e+00
34	-1.0	2.0	-8.152007e-01	-1.0	-1.0	-1.000000e+00	0.000000e+00
35	-1.0	2.0	-1.335448e+00	0.0	0.0	0.000000e+00	-1.000000e+00
36	-1.0	2.0	-2.165791e-01	-1.0	-1.0	-1.000000e+00	0.000000e+00
37	-1.0	2.0	-1.953094e+00	0.0	0.0	0.000000e+00	-1.000000e+00
38	-1.0	2.0	1.814574e+00	-1.0	-1.0	-1.000000e+00	0.000000e+00
39	-1.0	2.0	1.292680e+00	0.0	0.0	0.000000e+00	-1.000000e+00
40	-1.0	2.0	-3.289791e-01	-1.0	-1.0	-1.000000e+00	0.000000e+00

Tabela 7: Wartości z zadania 6

Wnioski:

W eksperymentach 1 i 2 możemy zauważyć że funkcja jest stała na całej dziedzinie (z wyjątkiem wartości początkowej). W eksperymencie 3 niewielkie zaburzenie wartości x_0 wprowadziło chaos w kolejnych iteracjach. Możemy zatem uznać proces numeryczny o powyższych parametrach za niestabilny oraz wiemy że obliczone wartości odbiegają od rzeczywistości. W eksperymentach 4 i 5 możemy zauważyć regularność, w obu przypadkach dwie wartości powtarzają się na przemian na całej dziedzinie (z wyjątkiem wartości początkowej). Dla eksperymentów 6 o 7 natomiast możemy zauważyć że funkcja po jakimś czasie zaczyna przyjmować tylko dwie wartości naprzemiennie. W przypadku tym proces numeryczny znajduje się w stanie "perfekcyjnie stabilnym". [1] Oznacza to, że arytmetyka o skończonej precyzji nadaje się do obliczenia kolejnych wartości.

Literatura

[1] Heinz-Otto Peitgen, Hartmut Jürgens, Dietmar Saupe, *Chaos and Fractals*, Springer, 2004, s.37-59