Sprawozdanie z Laboratorium
Obliczenia Naukowe - Lista 2

Wojciech Typer

6 listopada 2025

Zadanie 1

Cel zadania: Celem zadania jest porównanie wyników z zadania 5 z Listy 1 oraz obecnych. W obydwu zadaniach obliczamy iloczyn skalarny tymi samymi metodami, jednak dane wejściowe w obydwu zadaniach nieco się różnią - sprawdzamy jak usunięcię dziesiątej cyfry po przecinku wpłynie na wyniki

Wyniki z zadania 5 z Listy 1:

Liczba bitów	**Metoda**	**Wynik**
Precyzja 32-bitowa (`Float32`)		
	Metoda 1	-0.4999443
	Metoda 2	-0.4543457
	Metoda 3	-0.5
	Metoda 4	-0.5
Precyzja 64-bitowa (`Float64`)		
	Metoda 1	1.0251881368296672e-10
	Metoda 2	-1.5643308870494366e-10
	Metoda 3	0.0
	Metoda 4	0.0

Wyniki z obecnego zadania:

Liczba bitów	**Metoda**	**Wynik**
Precyzja 32-bitowa (`Float32`)		
	Metoda 1	-0.4999443
	Metoda 2	-0.4543457
	Metoda 3	-0.5
	Metoda 4	-0.5
Precyzja 64-bitowa (`Float64`)		
	Metoda 1	-0.004296342739891585
	Metoda 2	-0.004296342998713953
	Metoda 3	-0.004296342842280865
	Metoda 4	-0.004296342842280865

Użyte metody:

Metoda 1: Obliczanie "w przód":

$$\sum_{i=1}^{n} x_i y_i$$

Metoda 2: Obliczanie "w tył":

$$\sum_{i=n}^{1} x_i y_i$$

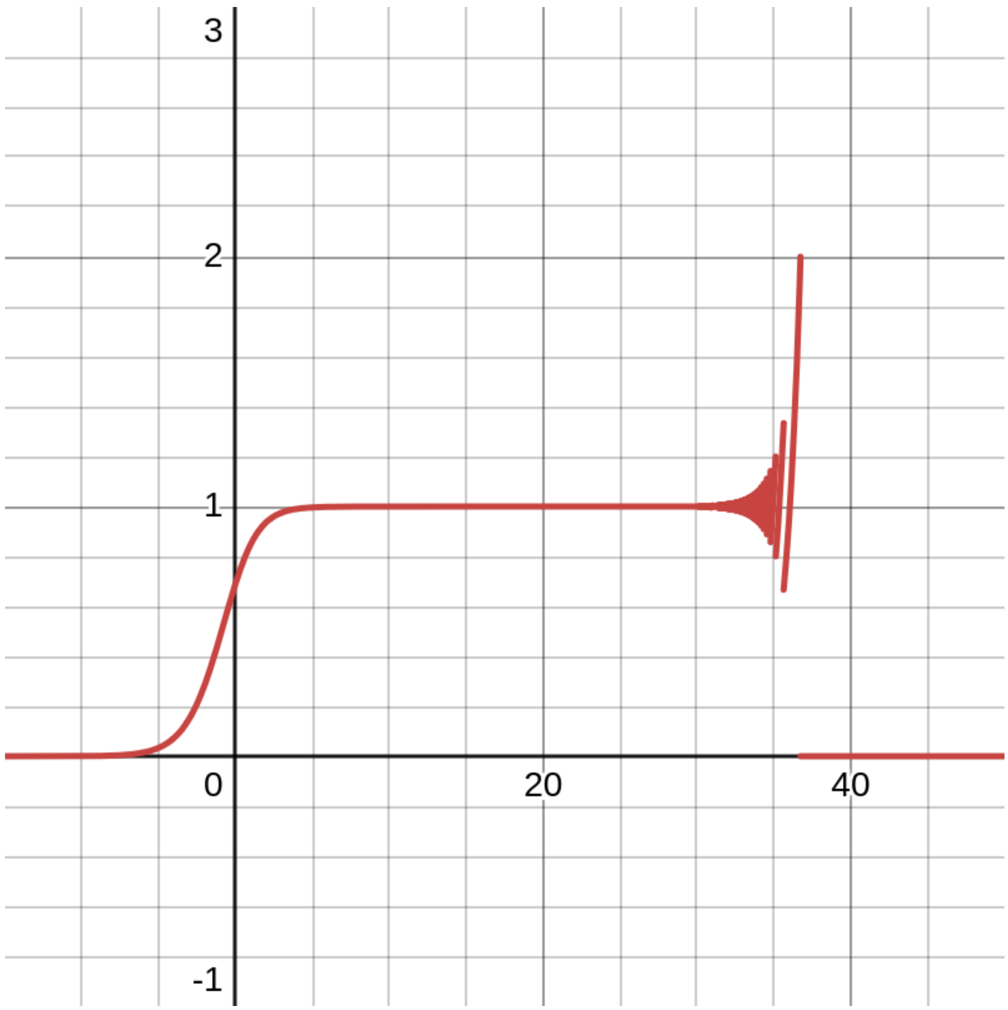
Metoda 3: Sumowanie osobno iloczynów dodatnich w porządku od największego do najmniejszego i osobno iloczynów ujemnych w porządku od najmniejszego do największego, a następnie dodanie obliczonych sum częściowych.

Metoda 4: Przeciwnie do metody 3.

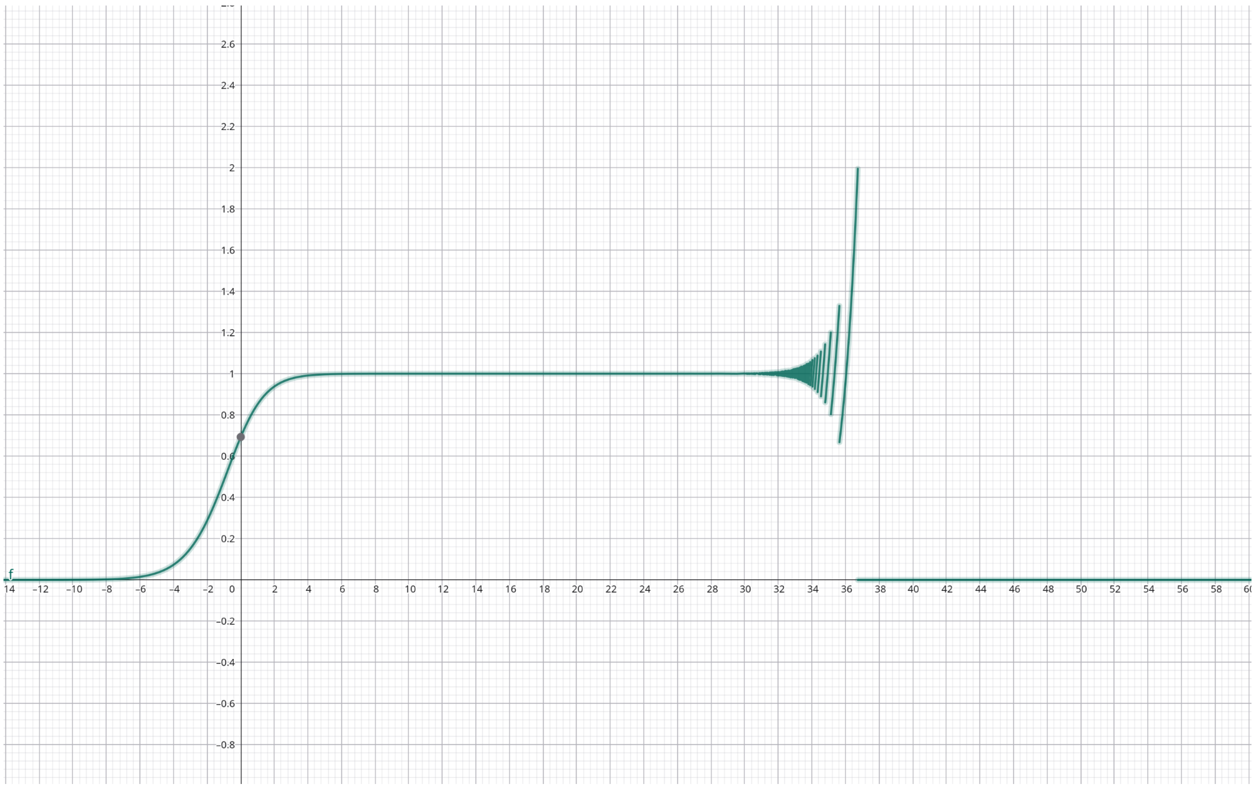
Wnioski: Zauważmy, że dla arytemtyki 32-bitowej wyniki nie uległy zmianie - wynika to ze zbyt małej precyzji tej arytmetyki. Dla arytmetyki 64-bitowej możemy zauważyć duże rozbieżności w uzyskanych wynikach, pomimo tego, że zmiana wektora x jest bardzo niewielka - możemy zatem wysnuć wnioski, że algorytmy, z których skorzystaliśmy są bardzo wrażliwe na zmiany danych, co z kolei świadczy o tym, że obliczenie iloczynu skalarnego $x \cdot y$ jest źle uwarunkowane.

Zadanie 2

Cel zadania: Narysowanie wykresu funkcji: $f(x) = e^x ln(1 + e^{-x})$ w co najmniej dwóch programach do wizualizacji oraz policzenie granicy: $\lim_{x\to\infty} f(x)$ oraz porównanie uzyskanego wyniku z wykresem funkcji.



Rysunek 1: Wykres 1, stworzony w programie Desmos.



Rysunek 2: Wykres 2, stworzony w programie Geogebra.

Obliczmy teraz granicę funkcji $f(x) = e^x \ln(1 + e^{-x})$ przy x dążącym do nieskończoności:

$$\begin{aligned}
\lim_{x\to\infty} e^x \ln(1+e^{-x}) &= \lim_{x\to\infty} \frac{\ln(1+e^{-x})}{e^{-x}} \\
&\overset{H}{=} \lim_{x\to\infty} \frac{\frac{d}{dx}\left(\ln(1+e^{-x})\right)}{\frac{d}{dx}\left(e^{-x}\right)} \\
&= \lim_{x\to\infty} \frac{\frac{1}{1+e^{-x}} \cdot (-e^{-x})}{-e^{-x}} \\
&= \lim_{x\to\infty} \frac{1}{1+e^{-x}} \\
&= \frac{1}{1+0} \\
&= 1
\end{aligned}$$

Wnioski: Zauważmy, że obliczona granica nie pokrywa się z uzyskanymi wykresami funkcji. Na wykresach wartość funkcji zdaje się dążyć do zera wraz ze wzrostem wartości x. Dzieje się tak dlatego, że dla dużych wartości x wyrażenie $ln(1 + e^{-x})$ jest bardzo małe i podczas obliczeń numerycznych jest zaokrąglane do zera co powoduje, że wartość funkcji $f(x)$ jest również zaokrąglana do zera. Czynnik e^x dla dużych wartości x jest bardzo duży, a mnożenie liczb różniących się wielkością rzędów jest obarczone bardzo dużym błędem, przez co użyte programy graficzne pokazują błędne wyniki.

Zadanie 3

Cel zadania: Rozwiązać układ równań liniowych postaci $Ax = b$ dla danej macierzy współczynników $A \in R^{n \times n}$ oraz wektora prawych stron $b \in R^n$, dwoma metodami: metodą eliminacji Gaussa, oraz metodą korzystającą wprost z równania $x = A^{-1}b$. Macierz A ma być wygenerowana na dwa sposoby: jako macierz Hilberta oraz jako macierz losowa o zadanym uwarunkowaniu.

Tabela 1: Porównanie błędów względnych dla macierzy Hilberta o rosnącym rozmiarze n

n	cond(A)	rank(A)	$\frac{\|\mathbf{x}-\tilde{\mathbf{x}}_{\text{gauss}}\|}{\|\mathbf{x}\|}$	$\frac{\|\mathbf{x}-\tilde{\mathbf{x}}_{\text{inv}}\|}{\|\mathbf{x}\|}$
1	1.00	1	0.0	0.0
2	$1.93e1$	2	$5.66e{-16}$	$1.40e{-15}$
3	$5.24e2$	3	$8.02e{-15}$	0.0
4	$1.55e4$	4	$4.14e{-14}$	0.0
5	$4.77e5$	5	$1.68e{-12}$	$3.35e{-12}$
6	$1.50e7$	6	$2.62e{-10}$	$2.02e{-10}$
7	$4.75e8$	7	$1.26e{-8}$	$4.71e{-9}$
8	$1.53e10$	8	$6.12e{-8}$	$3.08e{-7}$
9	$4.93e11$	9	$3.88e{-6}$	$4.54e{-6}$
10	$1.60e13$	10	$8.67e{-5}$	$2.50e{-4}$
11	$5.22e14$	10	$1.58e{-4}$	$7.62e{-3}$
12	$1.75e16$	11	$1.34e{-1}$	$2.59e{-1}$
13	$3.19e18$	11	$1.10e{-1}$	5.33
14	$6.20e17$	11	1.46	8.71
15	$3.68e17$	12	4.70	7.34
16	$7.05e17$	12	$5.42e1$	$2.98e1$
17	$1.25e18$	12	$1.37e1$	$1.05e1$
18	$2.25e18$	12	$1.03e1$	$2.48e1$

Tabela 2: Porównanie błędów względnych dla losowych macierzy R_n o zadanym wskaźniku uwarunkowania

c	n	cond(A)	rank(A)	$\frac{\|\mathbf{x}-\tilde{\mathbf{x}}_{\text{gauss}}\|}{\|\mathbf{x}\|}$	$\frac{\|\mathbf{x}-\tilde{\mathbf{x}}_{\text{inv}}\|}{\|\mathbf{x}\|}$
1.00	5	1.00	5	$1.99e{-16}$	$1.40e{-16}$
$1.00e1$	5	$1.00e1$	5	$1.40e{-16}$	$1.49e{-16}$
$1.00e3$	5	$1.00e3$	5	$5.93e{-14}$	$5.90e{-14}$
$1.00e7$	5	$1.00e7$	5	$3.37e{-10}$	$3.51e{-10}$
$1.00e12$	5	$1.00e12$	5	$1.32e{-5}$	$1.35e{-5}$
$1.00e16$	5	$8.26e15$	4	$1.66e{-1}$	$1.71e{-1}$
1.00	10	1.00	10	$2.67e{-16}$	$3.46e{-16}$
$1.00e1$	10	$1.00e1$	10	$5.04e{-16}$	$3.29e{-16}$
$1.00e3$	10	$1.00e3$	10	$1.52e{-14}$	$1.60e{-14}$
$1.00e7$	10	$1.00e7$	10	$1.53e{-10}$	$1.24e{-10}$
$1.00e12$	10	$1.00e12$	10	$1.74e{-6}$	$4.22e{-6}$
$1.00e16$	10	$8.25e15$	9	$2.13e{-1}$	$2.86e{-1}$
1.00	20	1.00	20	$4.44e{-16}$	$4.82e{-16}$
$1.00e1$	20	$1.00e1$	20	$3.24e{-16}$	$4.04e{-16}$
$1.00e3$	20	$1.00e3$	20	$4.18e{-15}$	$4.62e{-15}$
$1.00e7$	20	$1.00e7$	20	$3.35e{-10}$	$4.61e{-10}$
$1.00e12$	20	$1.00e12$	20	$1.32e{-5}$	$1.04e{-5}$
$1.00e16$	20	$8.31e15$	19	$5.17e{-2}$	$3.46e{-2}$

Wnioski: W przypadku macierzy Hilberta, wraz ze wzrostem n rośnie jej wskaźnik uwarunkowania, co przekłada się na wzrost błędów względnych obu metod rozwiązywania układu równań. Możemy z tego wywnioskować, że zadanie obliczenia układu równań z macierzą Hilberta jest źle uwarunkowane. W przypadku macierzy losowych, możemy zauważyć, że błędy względne są znacznie bardziej zależne od zadanych wskaźników uwarunkowania c niż od rozmiaru macierzy n. Gdy dowolna macierz ma wysoki wskaźnik uwarunkowania, to zadanie obliczenia układu równań z tą macierzą jest źle uwarunkowane.

Zadanie 4

Cel zadania: Dany jest wielomian P, będący naturalną postacią wielomianu Wilkinsona: $p(x) = \prod_{i=1}^{20}(x-i)$. Należy obliczyć: $|P(z_k)|, |p(z_k)| oraz |z_k - k|$ Następnie należy powtórzyć eksperyment Wilkinsona, tj. zmienić współczynik przy $x^{19}z - 210$ na $-210 - 2^{-23}$ i opisać otrzymane wyniki

Tabela 3: Analiza błędów dla pierwiastków wielomianu Wilkinsona w jego nie zaburzonej reprezentacji numerycznej

k	z_k	$\|z_k - k\|$	$\|P(z_k)\|$	$\|p(z_k)\|$
1	1.0	0.0000	$3.57e4$	$3.66e4$
2	2.0	0.0000	$1.76e5$	$1.81e5$
3	3.0	0.0000	$2.79e5$	$2.90e5$
4	4.0	0.0000	$3.03e6$	$2.04e6$
5	5.0	0.0000	$2.29e7$	$2.09e7$
6	6.0	0.0000	$1.29e8$	$1.13e8$
7	7.0001	0.0001	$4.81e8$	$4.57e8$
8	7.9994	0.0006	$1.64e9$	$1.56e9$
9	9.0029	0.0029	$4.88e9$	$4.69e9$
10	9.9904	0.0096	$1.36e10$	$1.26e10$
11	11.025	0.0250	$3.59e10$	$3.30e10$
12	11.9533	0.0467	$7.53e10$	$7.39e10$
13	13.0743	0.0743	$1.96e11$	$1.85e11$
14	13.9148	0.0852	$3.58e11$	$3.55e11$
15	15.0755	0.0755	$8.22e11$	$8.42e11$
16	15.9463	0.0537	$1.55e12$	$1.57e12$
17	17.0254	0.0254	$3.69e12$	$3.32e12$
18	17.9909	0.0091	$7.65e12$	$6.34e12$
19	19.0019	0.0019	$1.14e13$	$1.23e13$
20	19.9998	0.0002	$2.79e13$	$2.32e13$

Tabela 4: Analiza błędów dla pierwiastków zaburzonego wielomianu Wilkinsona

k	Przybliżony z_k	$\lvert z_k - k\rvert$	$\lvert P(z_k)\rvert$	$\lvert p(z_k)\rvert$
1	1.0	0.0000	2.03e4	2.00e4
2	2.0	0.0000	3.47e5	3.52e5
3	3.0	0.0000	2.26e6	2.42e6
4	4.0	0.0000	1.05e7	1.13e7
5	5.0	0.0000	3.76e7	4.48e7
6	6.0	0.0000	1.31e8	2.14e8
7	6.9996	0.0004	3.94e8	1.78e9
8	8.0078	0.0078	1.18e9	1.87e10
9	8.9158	0.0842	2.23e9	1.37e11
10	10.0955 - 0.6449i	0.6520	1.07e10	1.49e12
11	10.0955 + 0.6449i	1.1109	1.07e10	1.49e12
12	11.7939 - 1.6525i	1.6653	3.14e10	3.30e13
13	11.7939 + 1.6525i	2.0458	3.14e10	3.30e13
14	13.9924 - 2.5188i	2.5188	2.16e11	9.55e14
15	13.9924 + 2.5188i	2.7129	2.16e11	9.55e14
16	16.7307 - 2.8126i	2.9060	4.85e11	2.74e16
17	16.7307 + 2.8126i	2.8255	4.85e11	2.74e16
18	19.5024 - 1.9403i	2.4540	4.56e12	4.25e17
19	19.5024 + 1.9403i	2.0043	4.56e12	4.25e17
20	20.8469	0.8469	8.76e12	1.37e18

Wnioski: W przypadku nie zaburzonego wielomianu Wilkinsona, możemy zauważyć, że wyliczone pierwiastki nie pokrywają się z rzeczywistymi wartościami, jednak błędy bezwzględne są stosunkowo niewielkie. Niemniej, wartości wielomianu (zarówno w jego naturalnej jak i oryginalnej formie) w tych punktach są bardzo duże i rosną wraz ze wzrostem wartości k. Dzieje się tak, ponieważ wyliczenie pierwiastków w wielomianie Wilkinsona jest ekstrymalnie źle uwarunkowane.
W przypadku zaburzonego wielomianu Wilkinsona, sytuacja jest jeszcze gorsza - błędy bezwzględne pierwiastków są znacznie większe, a dodatkowo pojawiają się pierwiastki zespolone. Wszystkie te błędy wynikają z faktu, że jesteśmy zmuszni pracowac w arytmetyce z ograniczoną precyzją, a wielomian Wilkinsona jest przykładem wielomianu, którego pierwiastki są bardzo wrażliwe na nawet niewielkie zmiany współczynników.

Zadanie 5

Cel zadania: Rozpatrujemy następujące równanie rekurencyjne: $p_{n+1} = p_n + rp_n(1 - p_n)$, gdzie r jest pewną stałą, $r(1 - p_n)$ jest czynnikiem wzrostu populacji a p_0 jest wielkością populacji, stanowiącą procent maksymalnej wielkości populacji dla danego stanu środowiska. W zadaniu należy przeprowadzić 40 iteracji dla $p_0 = 0.01$ i $r = 3$ w arytmetyce float32, następnie ponownie wykonać 40 iteracji, z niewielką modyfikacją, tj. wykonać 10 iteracji, zatrzymać, zastosować obcięcie wyniku, odrzucając cyfry po 3 miejscu po przecinku i kontynuować kolejne 30 iteracji. Należy porównać oba wyniki. Następnie należy wykonać kolejnych 40 iteracji dla danych $p_0 = 0.01$ i $r = 3$ w arytmetyce float32 i float64.

Tabela 5: Wyniki 40 iteracji dla modelu logistycznego z $p_0 = 0.01$ i $r = 3$

Numer iteracji (n)	Wartość p_n
1	0.0397
2	0.154 071 73
3	0.545 072 6
4	1.288 978 1
5	0.171 518 8
6	0.597 819 1
7	1.319 113 4
8	0.056 273 222
9	0.215 592 86
10	0.722 930 6
11	1.323 836 4
12	0.037 716 985
13	0.146 600 22
14	0.521 926
15	1.270 483 7
16	0.239 548 2
17	0.786 042 8
18	1.290 581 3
19	0.165 524 72
20	0.579 903 6
21	1.310 749 8
22	0.088 804 245
23	0.331 558 4
24	0.996 440 7
25	1.007 080 6
26	0.985 688 5
27	1.028 008 6
28	0.941 629 4
29	1.106 519 8
30	0.752 920 9
31	1.311 013 9
32	0.087 783 1
33	0.328 014 8
34	0.989 278 1
35	1.021 099
36	0.956 466 56
37	1.081 381 4
38	0.817 368 27
39	1.265 200 4
40	0.258 605 48

Tabela 6: Wyniki 40 iteracji dla modelu logistycznego z $p_0 = 0.01$ i $r = 3$, z obcięciem po 10 iteracjach

Numer iteracji (n)	Wartość p_n
1	0.0397
2	0.154 071 73
3	0.545 072 6
4	1.288 978 1
5	0.171 518 8
6	0.597 819 1
7	1.319 113 4
8	0.056 273 222
9	0.215 592 86
10	0.722 930 6
10	**0.722**
11	1.324 147 9
12	0.036 488 414
13	0.141 959 44
14	0.507 380 37
15	1.257 216 9
16	0.287 084 52
17	0.901 085 5
18	1.168 476 8
19	0.577 893
20	1.309 691 1
21	0.092 892 17
22	0.345 681 82
23	1.024 239 5
24	0.949 758 23
25	1.092 910 8
26	0.788 281 2
27	1.288 963 1
28	0.171 574 83
29	0.597 985 57
30	1.319 182 2
31	0.056 003 93
32	0.214 606 39
33	0.720 257 8
34	1.324 717 3
35	0.034 241 438
36	0.133 448 33
37	0.480 367 96
38	1.229 211 8
39	0.383 962 2
40	1.093 568

Tabela 7: Wyniki 40 iteracji dla modelu logistycznego w arytmetyce Float64

Numer iteracji (n)	Wartość p_n
1	0.0397
2	0.15407173000000002
3	0.5450726260444213
4	1.2889780011888006
5	0.17151914210917552
6	0.5978201201070994
7	1.3191137924137974
8	0.056271577646256565
9	0.21558683923263022
10	0.722914301179573
11	1.3238419441684408
12	0.03769529725473175
13	0.14651838271355924
14	0.521670621435246
15	1.2702617739350768
16	0.24035217277824272
17	0.7881011902353041
18	1.2890943027903075
19	0.17108484670194324
20	0.5965293124946907
21	1.3185755879825978
22	0.058377608259430724
23	0.22328659759944824
24	0.7435756763951792
25	1.315588346001072
26	0.07003529560277899
27	0.26542635452061003
28	0.8503519690601384
29	1.2321124623871897
30	0.37414648963928676
31	1.0766291714289444
32	0.8291255674004515
33	1.2541546500504441
34	0.29790694147232066
35	0.9253821285571046
36	1.1325322626697856
37	0.6822410727153098
38	1.3326056469620293
39	0.0029091569028512065
40	0.011611238029748606

Wnioski: W przypadku arytmetyki Float32 widzimy, że obcięcie wartości po 10 iteracjach prowadzi do znacznie różnych wyników końcowych. Oznacza to, że model logistyczny jest bardzo wrażliwy na dokładność obliczeń, a nawet niewielkie zmiany mogą prowadzić do znacznych różnic w wynikach. Porównując wyniki uzyskane w arytmetyce Float32 i Float64, widzimy, że są one znacznie różne, co podkreśla wpływ precyzji obliczeń na dynamikę modelu.

Zadanie 6

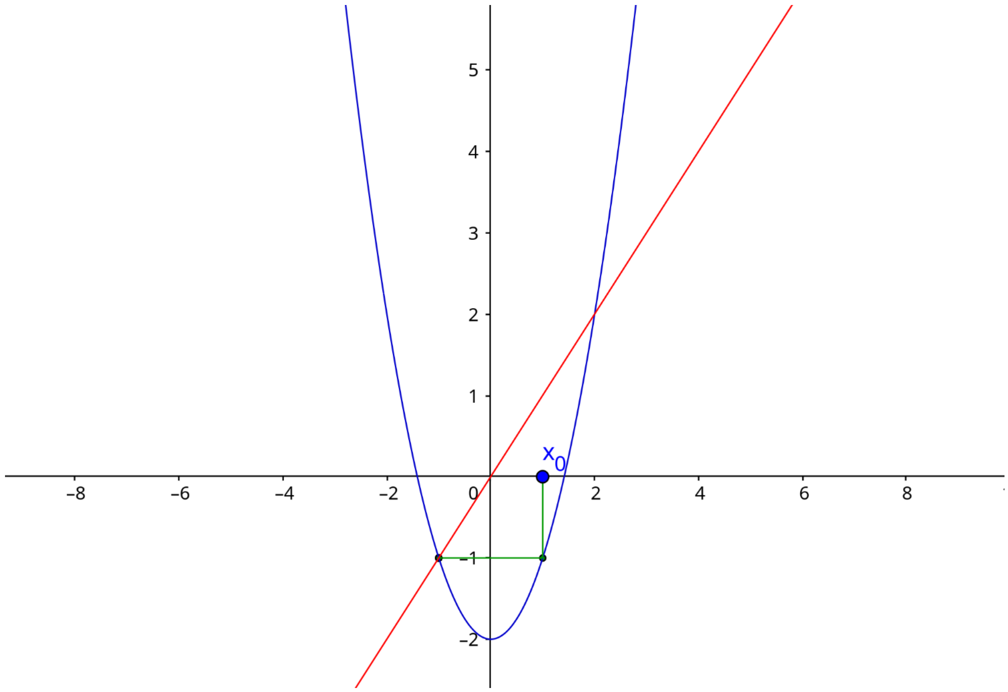
Cel zadania: Rozważamy równanie rekurencyjne: $x_{n+1} = x_n^2 + c$, gdzie c jest pewną stałą, a x_0 jest punktem startowym. Należy przeprowadzić po 40 iteracji dla następujących par: $(c, x_0) = (-2, 1), (-2, 2), (-2, 1.99999999999999), (-1, 1), (-1, -1), (-1, 0.75), (-1, 0.25)$ w arytmetyce Float64

Tabela 8: Porównanie przebiegu iteracji dla $c = -2$ i różnych wartości początkowych x_0

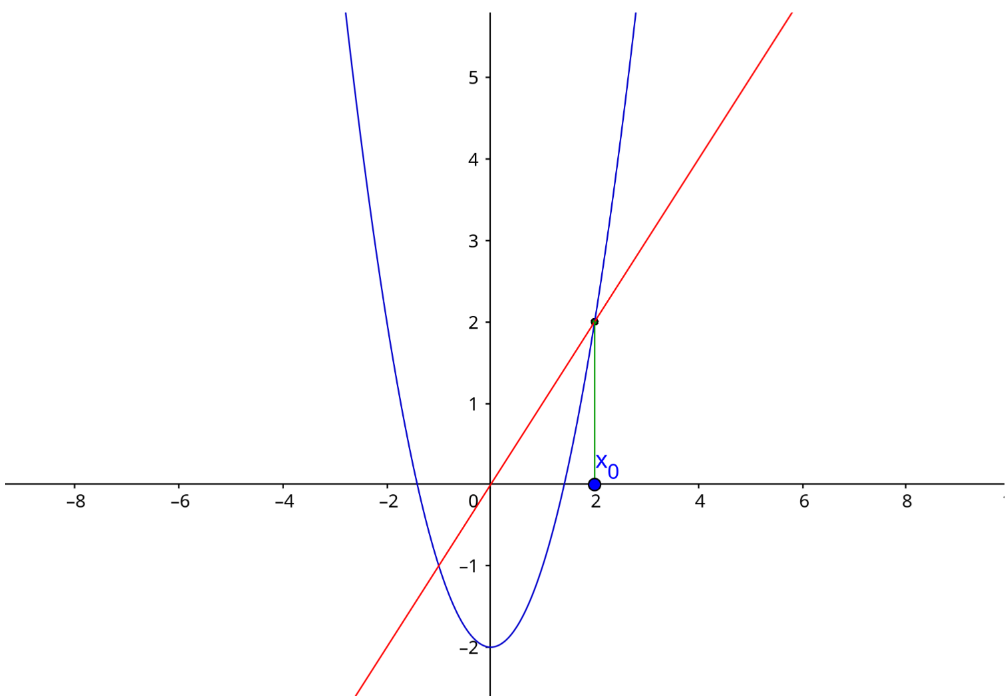
Krok (n)	x_n **(dla** $x_0 = 1$**)**	x_n **(dla** $x_0 = 2$**)**	x_n **(dla** $x_0 \approx 2$**)**
1	-1.0	2.0	1.99999999999996
2	-1.0	2.0	1.9999999999998401
3	-1.0	2.0	1.9999999999993605
4	-1.0	2.0	1.999999999997442
5	-1.0	2.0	1.9999999999897682
6	-1.0	2.0	1.9999999999590727
7	-1.0	2.0	1.999999999836291
8	-1.0	2.0	1.9999999993451638
9	-1.0	2.0	1.9999999973806553
10	-1.0	2.0	1.999999989522621
11	-1.0	2.0	1.9999999580904841
12	-1.0	2.0	1.9999998323619383
13	-1.0	2.0	1.9999993294477814
14	-1.0	2.0	1.9999973177915749
15	-1.0	2.0	1.9999892711734937
16	-1.0	2.0	1.9999570848090826
17	-1.0	2.0	1.999828341078044
18	-1.0	2.0	1.9993133937789613
19	-1.0	2.0	1.9972540465439481
20	-1.0	2.0	1.9890237264361752
21	-1.0	2.0	1.9562153843260486
22	-1.0	2.0	1.82677862987391
23	-1.0	2.0	1.3371201625639997
24	-1.0	2.0	-0.21210967086482313
25	-1.0	2.0	-1.9550094875256163
26	-1.0	2.0	1.822062096315173
27	-1.0	2.0	1.319910282828443
28	-1.0	2.0	-0.2578368452837396
29	-1.0	2.0	-1.9335201612141288
30	-1.0	2.0	1.7385002138215109
31	-1.0	2.0	1.0223829934574389
32	-1.0	2.0	-0.9547330146890065
33	-1.0	2.0	-1.0884848706628412
34	-1.0	2.0	-0.8152006863380978
35	-1.0	2.0	-1.3354478409938944
36	-1.0	2.0	-0.21657906398474625
37	-1.0	2.0	-1.953093509043491
38	-1.0	2.0	1.8145742550678174
39	-1.0	2.0	1.2926797271549244
40	-1.0	2.0	-0.3289791230026702

Tabela 9: Porównanie przebiegu iteracji dla $c = -1$ i różnych wartości początkowych x_0

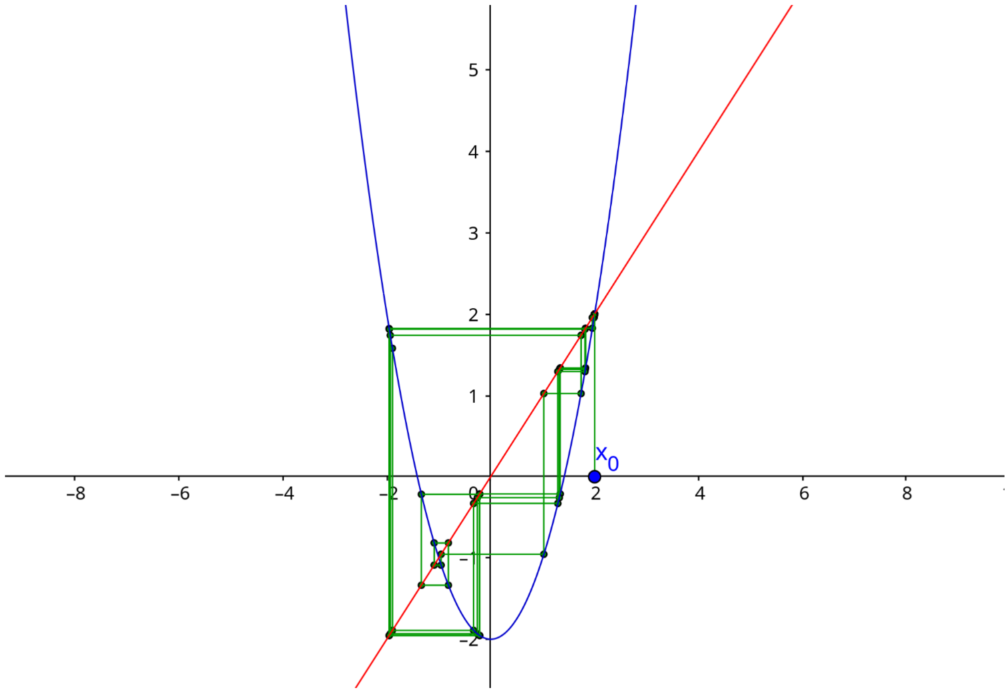
Krok (n)	x_n **(dla** $x_0 = 1$**)**	x_n **(dla** $x_0 = -1$**)**	x_n **(dla** $x_0 = 0.75$**)**	x_n **(dla** $x_0 = 0.25$**)**
1	0.0	0.0	-0.4375	-0.9375
2	-1.0	-1.0	-0.80859375	-0.12109375
3	0.0	0.0	-0.34617614	-0.98533630
4	-1.0	-1.0	-0.88016207	-0.02911237
5	0.0	0.0	-0.22531472	-0.99915247
6	-1.0	-1.0	-0.94923328	-0.00169434
7	0.0	0.0	-0.09895619	-0.99999713
8	-1.0	-1.0	-0.99020767	-5.741e-06
9	0.0	0.0	-0.01948876	-1.0
10	-1.0	-1.0	-0.99962019	0.0
11	0.0	0.0	-0.00075948	-1.0
12	-1.0	-1.0	-0.99999942	0.0
13	0.0	0.0	-1.154e-06	-1.0
14	-1.0	-1.0	-1.0	0.0
15	0.0	0.0	0.0	-1.0
16	-1.0	-1.0	-1.0	0.0
17	0.0	0.0	0.0	-1.0
18	-1.0	-1.0	-1.0	0.0
19	0.0	0.0	0.0	-1.0
20	-1.0	-1.0	-1.0	0.0
21	0.0	0.0	0.0	-1.0
22	-1.0	-1.0	-1.0	0.0
23	0.0	0.0	0.0	-1.0
24	-1.0	-1.0	-1.0	0.0
25	0.0	0.0	0.0	-1.0
26	-1.0	-1.0	-1.0	0.0
27	0.0	0.0	0.0	-1.0
28	-1.0	-1.0	-1.0	0.0
29	0.0	0.0	0.0	-1.0
30	-1.0	-1.0	-1.0	0.0
31	0.0	0.0	0.0	-1.0
32	-1.0	-1.0	-1.0	0.0
33	0.0	0.0	0.0	-1.0
34	-1.0	-1.0	-1.0	0.0
35	0.0	0.0	0.0	-1.0
36	-1.0	-1.0	-1.0	0.0
37	0.0	0.0	0.0	-1.0
38	-1.0	-1.0	-1.0	0.0
39	0.0	0.0	0.0	-1.0
40	-1.0	-1.0	-1.0	0.0



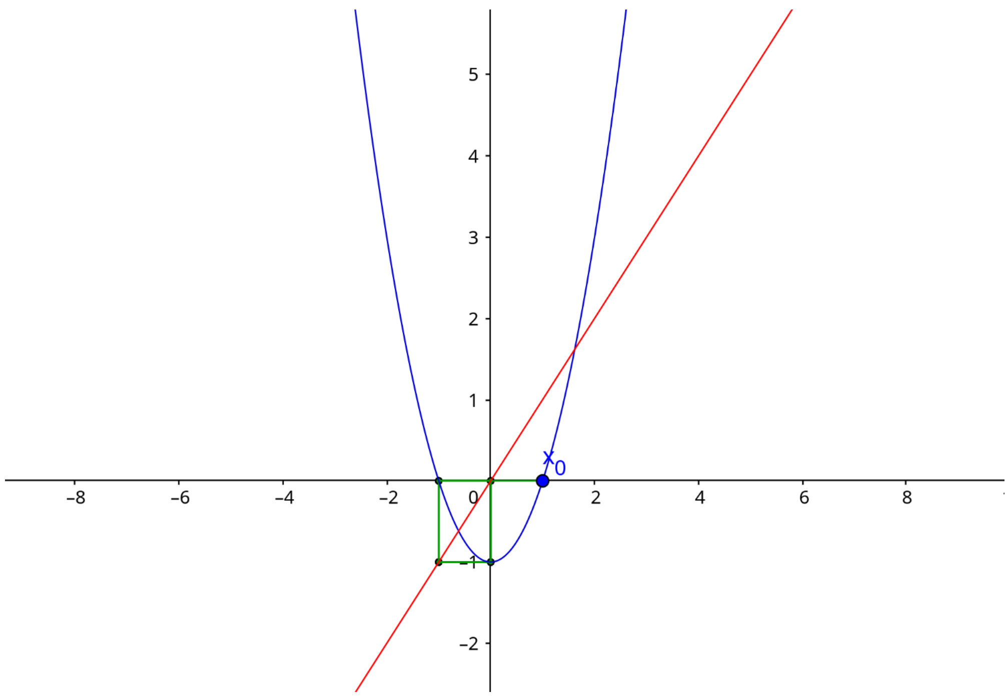
Rysunek 3: Wykres dla podpunktu 1



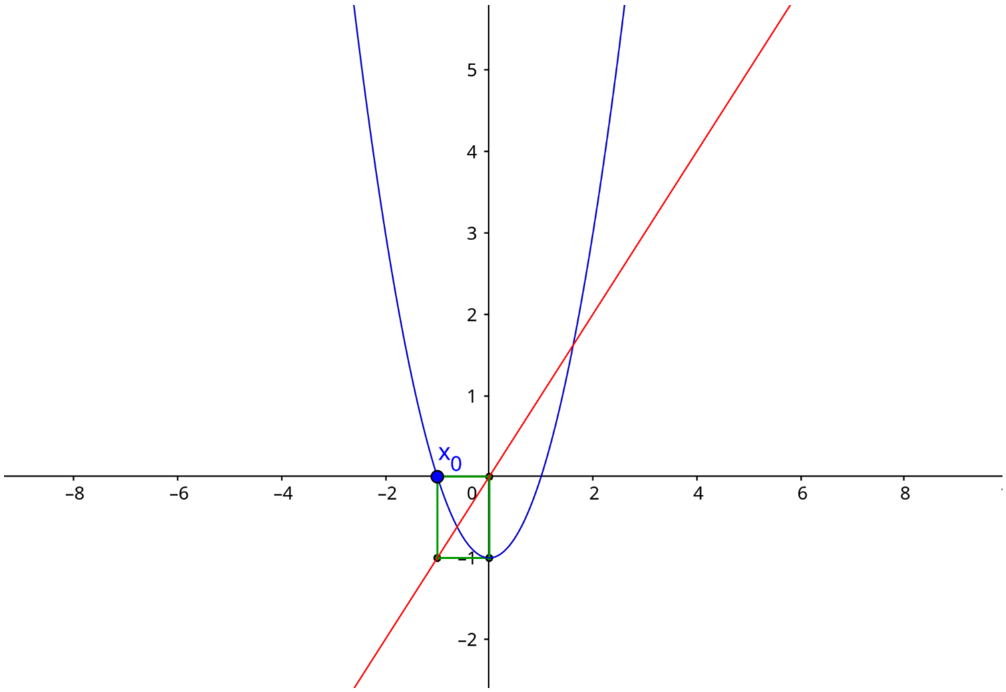
Rysunek 4: Wykres dla podpunktu 2



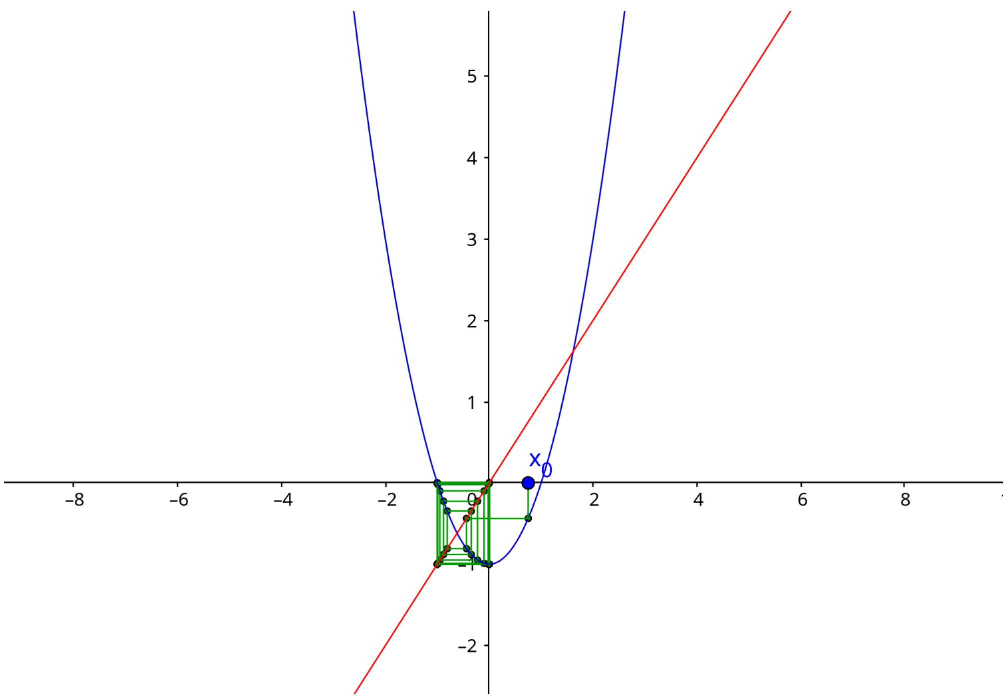
Rysunek 5: Wykres dla podpunktu 3



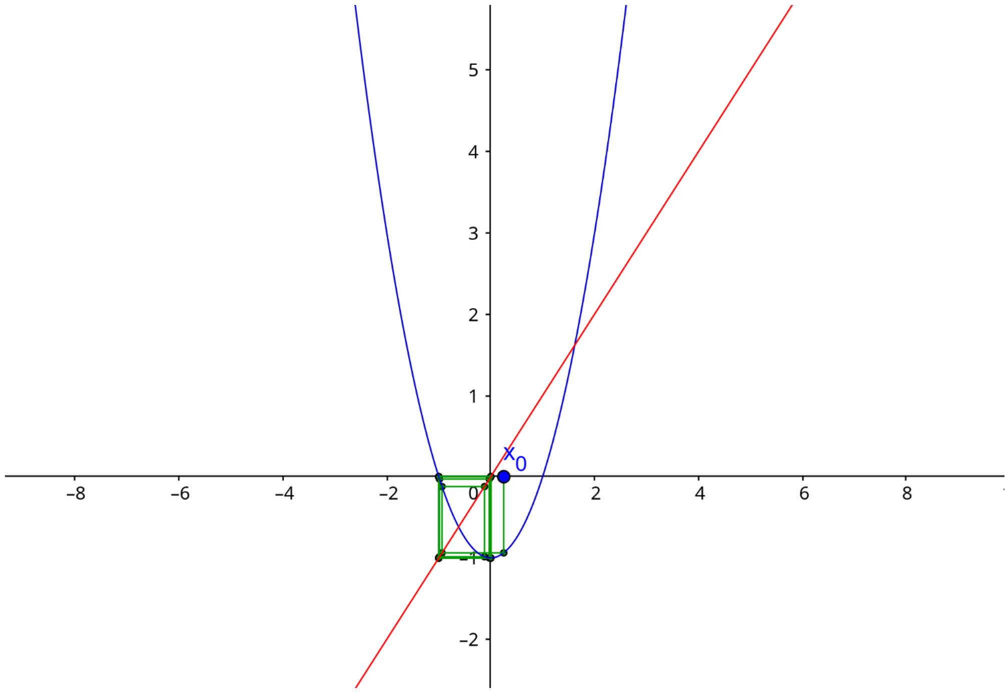
Rysunek 6: Wykres dla podpunktu 4



Rysunek 7: Wykres dla podpunktu 5



Rysunek 8: Wykres dla podpunktu 6



Rysunek 9: Wykres dla podpunktu 7

Wnioski: Dla podpunktu 1 i 2 widzimy, że kolejne iteracje nie wpływają na zmianę wartości - pozostają one stałe. Graficzne iteracje dla tych przypadków prowadzą do zachowania stabilnego. Dla podpunktu 3, pomimo bardzo dużego podobieństwa do podpunktu 2, widzimy, że uzyskane wartości są znacznie różne - wraz z kolejnymi iteracjami wartości zdają się coraz bardziej odbiegać od początkowej wartości. Dla kolejnych podpunktów, gdzie stała c jest równa -1 widzimy, że wraz z kolejnymi iteracjami, uzyskiwane wyniki zaczynają się stabilizować, wchodząc w cykl (0 , -1). Wartość x_0 w tym wypadku decyduje tylko o tym, jak szybko osiągnięty zostanie ten cykl.