Lista2

Dominik Kuszczak

November 4, 2015

1 Zadanie 1

1.1 Opis problemu

Powtórzyć zadanie 5 z poprzedniej listy ze zmodyfikowanymi danymi. Czy niewielkie zmiany danych maja wpływ na wynik?

1.2 Wyniki oraz wnioski

Wyniki bez modyfikacji

Zadanie/Typ	Float64	Float32
A	1.0251881368296672e-10	-0.4999443
B	-1.5643308870494366e-10	-0.4543457
C	0.0	-0.5
D	0.0	-0.5

Wyniki z modyfikacja

Zadanie/Typ	Float64	Float32
A	-0.004296342739891585	-0.4999443
B	-0.004296342998713953	-0.4543457
C	-0.004296342842280865	-0.5
D	-0.004296342842280865	-0.5

Jak widać wyniki zmieniły sie tylko dla float64. Dochodze wiec do wniosku że taka sytuacja nastapiła z tego z powodu iż ostatnie cyfry które usuneliśmy przekraczały już zakres float32, dlatego zmiana zaszła tylko w float64.

2 Zadanie 2

3 Zadanie 3

3.1 Opis problemu

Policzyć pierwiastki wielomianu wilkonsona w naturalnej postaci przy użyciu funkcji roots z pakietu Polymonials. Nastepnie podstawić je do dwóch postaci wielomianu i porównać wyniki oraz sprawdzić jaki bład zachodzi miedzy realnymi pierwiastkami a wyliczonymi przez nas. Dodatkowo zapoznać sie z funkcjami poly,Poly,polyval.

3.2 Rozwiazanie

Definiuje dwie funkcje P(x) oraz p(x) która obliczaja podany wielomian. Nastepnie przy użyciu fukncji roots obliczam pierwiastki. Dla obu funkcji obliczam wynik dla podanych miejsc zerowych. Dodatkowo wypisuje bład miedzy pierwiastkami.

3.3 Wyniki

P(x)	p(x)
229376.0	229077.22309599616
1.209856e6	1.2112764345202812e6
5.04832e6	5.250167456133662e
2.34368e7	2.460679292715959e7
1.1296512e8	1.1605285790588897e
5.03290368e8	5.067220377662679e8
1.968515584e9	1.9839859022898273e9
6.86119424e9	6.96914843387858e
2.1149393408e10	2.149719479063666e10
6.454844928e10	6.5226368764453735e10
1.43521440768e11	1.4543192592369525e11
5.32673099264e11	5.2239735055675696e11
6.02947259392e11	6.153583718321979e1
2.4994706855470215e12	2.6341999173582505e12
2.4994706855470215e12	2.6341999173582505e1
7.95998408192e12	7.849598258605353e
1.183580855552e13	1.179098959042234e1
2.248445502208e13	2.2668335447335188e13
3.7714572212736e13	3.8449171301277555e13
6.5522804164608e13	6.6069971603451984e13

Roots	Pierwiastek	Blad
0.9999999999981168 + 0.0im	1	1.8831602943691905e-12
2.0000000001891918 + 0.0im	2	1.8919177335874338e-10
2.9999999926196894 + 0.0im	3	7.380310584892413e-9
4.000000196012741 + 0.0im	4	1.9601274114933176e-7
4.999996302203527 + 0.0im	5	3.6977964725792845e-6
6.000048439601834 + 0.0im	6	4.8439601833649704e-5
6.999557630040994 + 0.0im	7	0.0004423699590061503
8.002891069857936 + 0.0im	8	0.0028910698579363014
8.986693042189247 + 0.0im	9	0.013306957810753417
10.049974037139467 + 0.0im	10	0.04997403713946724
10.886016935269065 + 0.0im	11	0.11398306473093456
12.358657519230299 + 0.0im	12	0.35865751923029876
12.561193394139806 + 0.0im	13	0.4388066058601936
14.51895930872283 + 0.2...1im	14	0.5610860887429185
14.51895930872283 - 0.2...1im	15	0.5262119169452244
16.206794587063147 + 0.0im	16	0.2067945870631469
16.885716688231323 + 0.0im	17	0.11428331176867701
18.030097274474777 + 0.0im	18	0.030097274474776725
18.993902180590464 + 0.0im	19	0.006097819409536243
20.000542093702702 + 0.0im	20	0.0005420937027018624

3.4 Wnioski i odpowiedzi

4 Zadanie 4

4.1 Opis problemu

Dla podanego modelu logistycznego wykonać 40 iteracji, 40 iteracji w tym po 10 obciać liczbe oraz 40 iteracji dla Float32 i Float64. Wyniki