|  |  |
| --- | --- |
| Stanislaw Kozera – 216810  Jan Kozera - 216809  Michal Kidawa - 216796 | Rok akademicki *2017/18*  *środa, 08:30* |

**METODY NUMERYCZNE – LABORATORIUM**

Zadanie *1* – *metoda bisekcji + regula falsi*

**Opis rozwiązania**

Aby odnaleźć pierwiastek równania nieliniowego zaimplementowaliśmy dwie metody obliczania miejsca zerowego: metodę bisekcji oraz regułę falsi.

**Metoda bisekcji** dzieli dany przedział na dwie części, po czym odrzuca połowę, w której wartość krańca przedziału ma identyczny znak, co wartość funkcji we wcześniej wyznaczonym środku przedziału. Czynność ta wykonywana jest do momentu osiągnięcia wyznaczonej wcześniej dokładności, lub po osiągnięciu maksymalnej liczby iteracji (w zależności od kryterium stopu wybranego przez użytkownika).

Przebieg algorytmu:

* Sprawdzenie, czy wartości funkcji na krańcach przedziału posiadają różne znaki
* Sprawdzenie, czy wartość funkcji w otrzymanym środku przedziału jest rozwiązaniem. Jeżeli tak, algorytm kończy działanie
* Jeżeli tak nie jest, dopóki dokładność ma wartość mniejszą od epsilonu / ilość iteracji jest mniejsza od maksymalnej ilości iteracji wybranej przez użytkownika:
  + Znowu podziel przedział na dwa równe przedziały
  + Środek przedziału staje się krańcem przedziału, którego wartość została odrzucona, gdyż miał przeciwny znak do wartości funkcji w środku przedziału
* Gdy została osiągnięta podana dokładność / ilość iteracji algorytm kończy działanie
* Pierwiastkiem równania jest wartość funkcji środka przedziału, który został wyznaczony jako ostatni.

**Regula falsi** na początku przeprowadza cięciwę między krańcami przedziałów, a następnie traktuje punkt przecięcia z osią OX jako przybliżenie pierwiastka i punkt ten staje się krańcem przedziału, którego wartość ma taki sam znak. Tak jak w przypadku poprzedniego algorytmu czynność ta jest powtarzana do momentu osiągnięcia wyznaczonej wcześniej dokładności, lub po osiągnięciu maksymalnej liczby iteracji (w zależności od kryterium stopu wybranego przez użytkownika).

Przebieg algorytmu:

* Sprawdzenie, czy wartości funkcji na krańcach przedziału posiadają różne znaki
* Przeprowadzenie przez punkty A = (a,f(a)) i B = (b,f(b)) cięciwy (a i b to krańce przedziałów)
* Uznanie punktu przecięcia z osią odciętych jako pierwszy przybliżenie pierwiastka.

**Wzór na miejsce pierwszego przecięcia z osią OX:**

* Jeżeli wartość bezwzględna funkcji w punkcie jest większa od wyznaczonego epsilonu to program kończy swoje działanie
* W przeciwnym przypadku, obliczony wcześniej punkt staje się nowym krańcem przedziału – tym, którego wartość funkcji miała identyczny znak co odnaleziony punkt.
* Gdy została osiągnięta podana dokładność / ilość iteracji algorytm kończy działanie

**Wyniki**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Metoda** | **Kraniec lewy** | **Kraniec prawy** | **Epsilon** | **Ilość Iteracji** | **Miejsce Zerowe** |
| Metoda Bisekcji | -5 | 10 | 0.001 | 14 | **0.695**953 |
| Regula Falsi | -5 | 10 | 0.001 | 330 | **0.695**443 |
| Metoda Bisekcji | -5 | 10 | 0.000001 | 24 | **0.695620**328 |
| Regula Falsi | -5 | 10 | 0.000001 | 498 | **0.695620**592 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Metoda** | **Kraniec lewy** | **Kraniec prawy** | **Epsilon** | **Ilość Iteracji** | **Miejsce Zerowe** |
| Metoda Bisekcji | 0 | 4 | 0.001 | 12 | **1.570**800 |
| Regula Falsi | 0 | 4 | 0.001 | 4 | **1.570**797 |
| Metoda Bisekcji | 0 | 4 | 0.000001 | 22 | **1.570796**489 |
| Regula Falsi | 0 | 4 | 0.000001 | 6 | **1.570796**326 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Metoda** | **Kraniec lewy** | **Kraniec prawy** | **Epsilon** | **Ilość Iteracji** | **Miejsce Zerowe** |
| Metoda Bisekcji | -2 | 1 | 0.001 | 12 | **-0.000**122 |
| Regula Falsi | -2 | 1 | 0.001 | 11 | **-0.000**210 |
| Metoda Bisekcji | -2 | 1 | 0.00001 | 22 | **-0.000000**119 |
| Regula Falsi | -2 | 1 | 0.000001 | 19 | **-0.000000**358 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Metoda** | **Kraniec lewy** | **Kraniec prawy** | **Epsilon** | **Ilość Iteracji** | **Miejsce Zerowe** |
| Metoda Bisekcji | -2 | 1 | 0.001 | 12 | **0.489**868 |
| Regula Falsi | -2 | 1 | 0.001 | 51 | **0.493**643 |
| Metoda Bisekcji | -2 | 1 | 0.000001 | 22 | **0.490072**369 |
| Regula Falsi | -2 | 1 | 0.000001 | 152 | **0.490076**165 |

**Wnioski**

* Obydwie metody można stosować do każdego rodzaju funkcji – są uniwersalne
* Ilość iteracji jest odwrotnie proporcjonalna do wartości epsilon
* Niemożliwe jest określenie, która z metod jest szybsza