

# Pierwszy projekt z labu

Zamiast w octave można zaprogramować odpowiednią funkcję i testy w Pascalu, C/C++, fortranie.

Projekt składa się z dwóch części:

1. Funkcja octave'a z metodą Steffensena: tzn.

Zaprogramować funkcję octave'a w m-pliku `steff.m` z **metodą Steffensena** zdefiniowaną wzorem

$$x_{n+1} = x_n - \frac{f(x_n)^2}{f(x_n + f(x_n)) - f(x_n)}$$

- która ma znaleźć przybliżenie  $f(x^*) = 0$ . Jako warunek stopu wziąć  $|f(x_n)| < 10^{-7}$  lub  $|x_{n+1} - x_n| < 10^{-6}$  i dodatkowo warunek negatywny że ilość iteracji przekroczyła 100. W tym ostatnim przypadku uznajemy że nie ma zbieżności metody.

Parametrami funkcji:

```
function [x, it, kod, er] = steff(FCN, x0, sol)
# kod funkcji
endfunction
```

mają być:

- FCN- 'wskaźnik' do funkcji  $f$  (function handle),
- x0 - przybliżenie startowe
- sol - rozwiązanie równania (o ile je znamy) - parametr opcjonalny

Funkcja ma zwracać:

- (a) obliczony przybliżony pierwiastek  $x$
- (b) `it` - ilość iteracji
- (c) `kod` - kod wyniku - funkcja zwraca zero jak metoda zbiegła i.e. któryś z warunków stopu spełniony i jeden jak przekroczone max ilość iteracji (ustaloną w kodzie funkcji na 100)
- (d) `er` - wektor błędów:  $er(k+1) = |x_k - sol|$  dla  $k = 0, 1, 2, \dots, it$  (opcjonalnie tylko o ile `sol` został podany jako parametr wejściowy)

2. Testy tej funkcji i porównanie z metodą Newtona: Przetestować na przykładach np. na tych które zostały podane w zad 1 na drugim labie, w szczególności:

- (a) zbadać rząd lokalnej zbieżności metody - czy jest kwadratowy lub liniowy czyli wyświetlić  $|e_{n+1}|/|e_n|^2$  i  $|e_{n+1}|/|e_n|$  z  $e_n = |x_n - sol|$  dla kolejnych  $n$  tak aby dla przykładów dla których znamy rozwiązanie `sol` i dla których metoda zbiega sprawdzić czy te ciągi zbiegają.
- (b) zbadać na przykładzie  $\cos(x) = 0$  i  $x * x - 4 = 0$  zbieżność dla różnych  $x_0$  - sprawdzić eksperymentalnie czy jeśli metoda Newtona zbiega dla danego  $x_0$  to m. Steffensena też zbiega.

W przypadku napisania funkcji/testów w innym języku trzeba się trzymać zasad danego języka i np w C - parametrami opcjonalnymi powinny być wskaźniki, a wyjściowe też mogą być wskaźnikami.