Grupa lab.3	Data wykonania 7.06.2024	Inżynieria Obliczeniowa 2023/2024
Temat ćwiczenia **Rozwiązywanie równań nieliniowych -metoda stycznych i siecznych**		
Imię i nazwisko Karolina Kurowska		Ocena i uwagi

Cel ćwiczenia:

Celem ćwiczenia było zapoznanie się z metodami numerycznymi do rozwiązywania równań nieliniowych: metodą stycznych oraz metodą siecznych. Zadaniem było zaimplementowanie obu metod w języku programowania, rozwiązanie przykładowego równania nieliniowego oraz porównanie wyników uzyskanych za pomocą obu metod.

Wstęp:

Rozwiązywanie równań nieliniowych jest jednym z podstawowych problemów w matematyce i inżynierii. Istnieje wiele metod numerycznych do znajdowania rozwiązań takich równań, z których dwie najbardziej popularne to metoda stycznych oraz metoda siecznych.

Metoda stycznych opiera się na iteracyjnym przybliżaniu pierwiastka równania przy użyciu pochodnej funkcji. Przybliżenie pierwiastka w k+1 iteracji jest dane wzorem:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

Metoda siecznych jest podobna do metody stycznych, ale nie wymaga obliczania pochodnej funkcji. Pochodna jest aproksymowana ilorazem różnicowym:

$$f'(x_k) \approx \frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}$$

Stąd przybliżenie pierwiastka w k+1 iteracji jest dane wzorem:

$$x_{k+1} = x_k - f(x_k)\frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})}$$

Implementacja ćwiczenia:

Wzory i pochodne funkcji:

```cpp
// Definicja funkcji f(x)
double f (double x) {
    return (-1*(pow (x, 3)) + (10 * x) + 5); //FUNCKJA Z INSTRUKCJI
    //return ((pow (x, 4)) + (3 * pow(x, 2)) + (2 * x) + 1);//FUNCKJA WŁASNA
}

// Definicja pochodnej f'(x)
double f_pochodna (double x) {
    return (( -1 * 3 * pow (x, 2)) + 10);//POCHODNA Z INSTRUKCJI
    //return ((4 * pow (x, 3)) + (6 * x) + 2);//POCHODNA WŁASNA
}
```

Metoda siecznych i stycznych

```cpp
void metoda_stycznych (double (*f)(double), double (*f_poch)(double), double x0, int iterations) {
    double x = x0;
    cout << "Iteracja 0" << ": x = " << x0 << ", f(x) = " << f (x0) << endl;
    for (int i = 0; i < iterations; ++i) {
        double x_next = x - (f(x) / f_poch(x));
        cout << "Iteracja " << i + 1 << ": x = " << x_next << ", f(x) = " << f (x_next) << endl;
        x = x_next;
    }
}

void metoda_siecznych (double (*f)(double), double x_prev, double x0, int iterations) {
    cout << "Iteracja 0" << ": x = " << x0 << ", f(x) = " << f (x0) << endl;
    for (int i = 0; i < iterations; ++i) {
        double x_next = x0 - (f(x0) * ((x0 - x_prev) / (f(x0) - f(x_prev))));
        cout << "Iteracja " << i + 1 << ": x = " << x_next << ", f(x) = " << f (x_next) << endl;
        x_prev = x0;
        x0 = x_next;
    }
}
```

Wywołanie:

```cpp
int main () {
    double x0 = 6;
    int iterations = 5; //ZMIANA ITERACJI

    cout << "Metoda stycznych:" << endl;
    metoda_stycznych (f, f_pochodna, x0, iterations);

    cout << "\nMetoda siecznych:" << endl;
    double x_prev = x0 - 0.1;
    metoda_siecznych (f, x_prev, x0, iterations);

    return 0;
}
```

Wyniki:

Dla funkcji $-x^3 + 10x + 5$:

```
Konsola debugowania progra

Metoda stycznych:
Iteracja 0: x = 6, f(x) = -151
Iteracja 1: x = 4.45918, f(x) = -39.076
Iteracja 2: x = 3.6722, f(x) = -7.79786
Iteracja 3: x = 3.41616, f(x) = -0.705444
Iteracja 4: x = 3.38795, f(x) = -0.00813104
Iteracja 5: x = 3.38762, f(x) = -1.12544e-06

Metoda siecznych:
Iteracja 0: x = 6, f(x) = -151
Iteracja 1: x = 4.43052, f(x) = -37.6636
Iteracja 2: x = 3.90895, f(x) = -15.6389
Iteracja 3: x = 3.53861, f(x) = -3.92348
Iteracja 4: x = 3.41458, f(x) = -0.666019
Iteracja 5: x = 3.38922, f(x) = -0.0391795
```

Dla funkcji $x^4 + 3x^2 + 2x + 1$: (dla x=3 i 5 iteracji)

```
Metoda stycznych:
Iteracja 0: x = 3, f(x) = 115
Iteracja 1: x = 2.10156, f(x) = 37.9589
Iteracja 2: x = 1.36786, f(x) = 12.8497
Iteracja 3: x = 0.739346, f(x) = 4.4174
Iteracja 4: x = 0.190784, f(x) = 1.49209
Iteracja 5: x = -0.279538, f(x) = 0.681454

Metoda siecznych:
Iteracja 0: x = 3, f(x) = 115
Iteracja 1: x = 2.0606, f(x) = 35.8887
Iteracja 2: x = 1.63445, f(x) = 19.4196
Iteracja 3: x = 1.13194, f(x) = 8.74949
Iteracja 4: x = 0.719891, f(x) = 4.26308
Iteracja 5: x = 0.328349, f(x) = 1.99176
```

Wnioski:

Obie metody, zarówno stycznych, jak i siecznych, skutecznie znajdują przybliżone rozwiązania równań nieliniowych. Metoda stycznych, wymagająca znajomości pochodnej, może być bardziej precyzyjna i szybka, ale metoda siecznych, nie wymagająca pochodnej, również daje dokładne wyniki przy odpowiedniej liczbie iteracji. Wyniki uzyskane za pomocą obu metod są zbliżone do siebie i wskazują na prawidłowość implementacji.