Министерство науки и высшего образования Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧЕРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”

Факультет ПИиКТ



ОТЧЁТ

По лабораторной работе №2

По предмету: Вычислительная математика

Вариант: 1АГ

Студент:

Андрейченко Леонид Вадимович

Группа P3230

Преподаватель: Перл Ольга Вячеславовна

Санкт – Петербург

2022

**Описание методов**

В данной лабораторной работе мной были реализованы три метода решения нелинейных уравнений. Опишем алгоритм каждого из них:

**Метод половинного деления**

Идея метода: начальный интервал делим пополам, получаем начальное приближение к корню:

Вычисляем f(𝑥0). В качестве нового интервала выбираем ту половину отрезка, на концах которого функция имеет разные знаки: [a0 ,x0] либо [b0 ,x0]. Другую половину отрезка [a0 , b0], на которой функция f(х) знак не меняет, отбрасываем. Новый интервал вновь делим пополам, получаем очередное приближение к корню: х1=(a1+b1)/2.

Основная формула метода: . Окончания итерационного процесса: |bn – an |≤ ε

**Метод простой итерации**

Уравнение 𝑓 (𝑥) = 0 приведем к эквивалентному виду: 𝑥 = 𝜑(𝑥), выразив 𝑥 из исходного уравнения. Зная начальное приближение: 𝑥0 ∈ 𝑎, 𝑏 , найдем очередные приближения: 𝑥1 = 𝜑(𝑥0) → 𝑥2 = 𝜑 𝑥1 …

Рабочая формула метода: **xi+1= 𝜑(𝑥i)**

Условия сходимости метода простой итерации определяются следующей теоремой. Теорема. Если на отрезке локализации 𝑎, 𝑏 функция 𝜑(𝑥) определена, непрерывна и дифференцируема и удовлетворяет неравенству: 𝜑 ′ 𝑥 < 𝑞, где 0 ≤ 𝑞 < 1 , то независимо от выбора начального приближения 𝑥0 ∈ 𝑎, 𝑏 итерационная последовательность метода будет сходится к корню уравнения.

**Метод Ньютона (для решения систем)**

Метод решения систем нелинейных уравнений является обобщением метода Ньютона решения нелинейных уравнений, который основан на идее линеаризации. Пусть F(x):R1→R1 - дифференцируемая функция и необходимо решить уравнение F(x)=0. Взяв некоторое x0 в качестве начального приближения решения, мы можем построить линейную аппроксимацию F(x) в окрестности x0:F(x0+h)≈F(x0)+F′(x0)h и решить получающееся линейное уравнение F(x0)+F′(x0)h=0.

Таким образом получаем итеративный метод : **xk+1=xk−F′(xk)−1F(xk), k=0,1,…**

Итерационный процесс решения систем нелинейных уравнений методом Ньютона состоит в определении приращений ∆𝑥1 , ∆𝑥2 , … , ∆𝑥𝑛 к значениям неизвестных на каждой итерации. Критерий окончания итерационного процесса: max|∆xi<=ε|

**Блок-схемы методов**

|  |  |  |
| --- | --- | --- |
| Метод половинного деления | Метод простой итерации | Метод Ньютона |

**Листинг программы**

|  |  |  |
| --- | --- | --- |
| Метод половинного деления  def bisection\_method(equation, a, b, sigma):  print(**"Метод деления пополам:"**)  while b - a > sigma:  middle = equation(a) \* equation((a + b) / 2)  if middle > 0:  a = (a + b) / 2  elif middle < 0:  b = (a + b) / 2  else:  break  print(**"Корень: "** + str((a + b) / 2) + **"**\n**"**)  return (a + b) / 2 | Метод простой итерации  def fixed\_point\_iteration\_method(equation, a, sigma):  print(**"Метод простой итерации:"**)  dx = float(1)  while dx \* dx > sigma:  b = a  a = equation(b)  dx = a - b  print(**"Корень: "** + str(a) + **"**\n**"**)  return a | Метод Ньютона  def newton\_method(kind, x, y, sigma=0.001):  X = np.array([x, y])  dx = [1, 1]  while abs(dx[0]) > sigma or abs(dx[1]) > sigma:  X\_last = X  dx = np.dot(np.linalg.inv(df(X[0], X[1], kind)), f(X\_last[0], X\_last[1], kind))  X += dx  print(**"Вычисленные корни системы:** \n**x1 = "** + str(X[0]) + **"**\n**x2 = "** + str(X[1])) |

**Пример работы**

Изображение выглядит как текст

Автоматически созданное описаниеИзображение выглядит как текст

Автоматически созданное описание

**Вывод**

В ходе выполнения данной лабораторной работы я узнал какими способами можно решать системы нелинейных уравнений и обычные нелинейные уравнения. Если обобщить реализованные мной методы, то можно сказать следующее: *Метод половинного деления* прост и надежен, обладает абсолютной сходимостью надо применять, когда требуется высокая надежность счета, а скорость несущественна. Также если интервал содержит несколько корней, то неизвестно к какому относится вычислительный процесс, имеет линейную сходимость. *Метод простой итерации –* довольно прост, однако недостатком этого метода является его сходимость в малой окрестности корня и вытекающая отсюда необходимость выбора начального приближения к корню из этой малой окрестности. В противном случае итерационный процесс расходится или сходится к другому корню этого уравнения. Если 𝜑 ′ 𝑥 ≈ 1, то сходимость может быть очень медленной. *В методе Ньютона* важен удачный выбор начального приближения для обеспечения хорошей сходимости. Сходимость ухудшается с увеличением числа уравнений системы.