БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет прикладной математики и информатики

Отчет по лабораторной работе №1
Тема: «Численное решение нелинейных уравнений»

Выполнил:
Гаргома А. О.

Преподаватель:
Горбачева Ю. Н.

Минск 2020

Содержание

1 Постановка задачи

Решить уравнение $f(x) = 0$ согласно своему варианту с точностью $\epsilon = 10^{-7}$ методом простой итерации и методом Ньютона. Корень отделяем сначала графически, затем с помощью метода половинного деления до $\epsilon = 10^{-2}$. Если корней несколько, то необходимо найти ближайший к началу координат. Провести сравнительный анализ полученных результатов.

$$f(x) = ln(x + 2) - x^2 \tag{1}$$

2 Теория

2.1 Метод простой итерации решения нелинейных уравнений

Рассмотрим уравнение вида

$$f(x) = 0 \tag{2}$$

где f – некоторая заданная функция, x - неизвестная величина.

Пусть уравнение (2) каким-либо способом приведено к виду, пригодному для итерации:

$$x = \phi(x)$$

Будем считать, что корень x_∞ отделен и указано некоторое начальное приближение x_0. Тогда уточнение этого значения производят по правилу

$$x_{k+1} = \phi(x_k), k = 0, 1, \ldots \tag{3}$$

Формула (3) задаёт вычислительный процесс метода простой итерации решения нелинейных уравнений.

Теорема 1. *Пусть функция* $\phi(x)$ *на отрезке* $\Delta = x : |x - x_0| \leq \beta$ *имеет непрерывную производную и удовлетворяет условиям*

$$|\phi'(x)| \leq q < 1, x \in \Delta \tag{4}$$

$$|x_0 - \phi(x_0)| \leq (1 - q)\delta$$

Тогда уравнение $x = \phi(x)$ *имеет на отрезке* Δ *единственное решение, и последовательность* (3) сходится к этому решению.

2.2 Вычислительный процесс метода Ньютона

Рассмотрим уравнение вида

$$f(x) = 0$$

где f - заданная функция, x - неизвестная числовая величина.

Предположим, что каким-либо способом получено приближение x_k к корню x_∞ . Погрешность этого приближения обозначим через ϵ_k. При известном x_k поиск корня равносилен поиску погрешности ϵ_k. Имеем:

$$f(x_k + \epsilon_k) = 0 \tag{5}$$

Рассмотрим разложение левой части уравнения (5) в ряд Тейлора, взяв в разложении два первых слагаемых:

$$f(x_k) + \epsilon_k f'(x_k) + O(\epsilon_k^2) = 0$$

Если считать величину ϵ_k небольшой по модулю и отбросить остаточный член $O(\epsilon_k^2)$, то получим приближенное равенство

$$f(x_k) + \epsilon_k * f'(x_k) \approx 0$$

из которого найдём некоторое приближение Δx_k значения ϵ_k:

$$\Delta x_k = -\frac{f(x_k)}{f'(x_k)}$$

Прибавив эту поправку Δx_k к x_k получим новое приближение:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}, k = 0, 1, \ldots \tag{6}$$

Теорема 2. *Если:*

1) функция $f(x)$ определена и дважды непрерывно дифференцируема на отрезке S_0 вещественной оси с концами в точках x_0 и $x_0 + 2h_0$, где $h_0 = -\frac{f(x_0)}{f'(x_0)}$, при этом на концах отрезка $S_0 f(x) f'(x) \neq 0$
2) выполняется неравенство $2|h_0|M \leq |f'(x_0)|$, где $M = \max x \in S_0 |f''(x)|$, то
1)внутри отрезка S_0 единственный корень
2)последовательность приближений может быть построена по методу Ньютона, которая будет сходится к этому корню

3 Листинг программы

```
# coding: utf-8
import itertools as it
import sympy as sp
from math import *
import matplotlib.pylab as pl
import numpy as np
import scipy.optimize as opt
from tabulate import tabulate


def f(x):
    return log(x + 2) - x ** 2

from IPython.display import display, Markdown, Latex

# Локализация корня методом дихотомии
def localize_root(func, a0, b0, epsilon):
    k = [0]
    a = [a0]
    b = [b0]
    f_a = [func(a0)]
    f_b = [func(b0)]
    med = [(a[-1] + b[-1]) / 2]
    f_med = [func(med[-1])]
    error = [abs(a0 - b0) / 2]
    while (error[-1] >= epsilon):
        if (f_med[-1] > 0):
            a.append(a[-1])
            b.append(med[-1])
        elif (f_med[-1] < 0):
```

```
            b.append(b[-1])
            a.append(med[-1])
        else:
            break
        k.append(k[-1] + 1)
        f_a.append(func(a[-1]))
        f_b.append(func(b[-1]))
        med.append((a[-1] + b[-1]) / 2)
        f_med.append(func(med[-1]))
        error.append(abs(a[-1] - b[-1]) / 2)
    display(Markdown(tabulate([['$' + column + '$' for column in
                                ['k', 'a_k', 'b_k', 'f(a_k)', 'f(b_k)', '\\frac{a_k+b_k}{2}', 'f(\\frac{a_k+b_k}{2})', '\\epsilon_k']],
                               *list(zip(k, a, b, f_a, f_b, med, f_med, error))], headers='firstrow', tablefmt='pipe')))
    return a[-1], b[-1]


delta = [-1, -0.5]
error = 10 ** -7
localization_error = 10 ** -2
a, b = localize_root(f, *delta, localization_error)

t = sp.symbols('t')


def phi(t):
    return t + c * f(t)


def fsym(x):
    return sp.log(t + 2) - t ** 2


c = -0.3
deriv = sp.lambdify(t, sp.diff(t + c * fsym(t), t)) # производная \phi
x0 = (a + b) / 2
delta1 = abs(b - x0)
q = 0.45
abs(x0 - phi(x0)) <= (1 - q) * delta1

# Метод простых итераций
def im(a, b, x0, error):
    k = [0]
    result = [x0]
```

```
    eps = [float('inf')]
    while (eps[-1] >= error):
        result.append(phi(result[-1]))
        eps.append(abs(result[-1] - result[-2]))
        k.append(k[-1] + 1)
    return k, result, eps


df = sp.lambdify(t, sp.diff(fsym(t), t))

# Метод Ньютона
def newton(a, b, x0, error):
    k = [0]
    result = [x0]
    eps = [float('inf')]
    while (eps[-1] >= error):
        result.append(result[-1] - f(result[-1]) / df(result[-1]))
        eps.append(abs(result[-1] - result[-2]))
        k.append(k[-1] + 1)
    return k, result, eps



k1, res1, eps1 = im(a, b, x0, error)
k2, res2, eps2 = newton(a, b, x0, error)
print(tabulate([*list(it.zip_longest(k1, res1, eps1, res2, eps2, fillvalue='-'))], tablefmt='pipe', floatfmt='.9f'))
```

4 Результаты

Для отделения корня построим график функции. Будем находить корень $x \in [-1; -0.5]$ (графическое отделение)

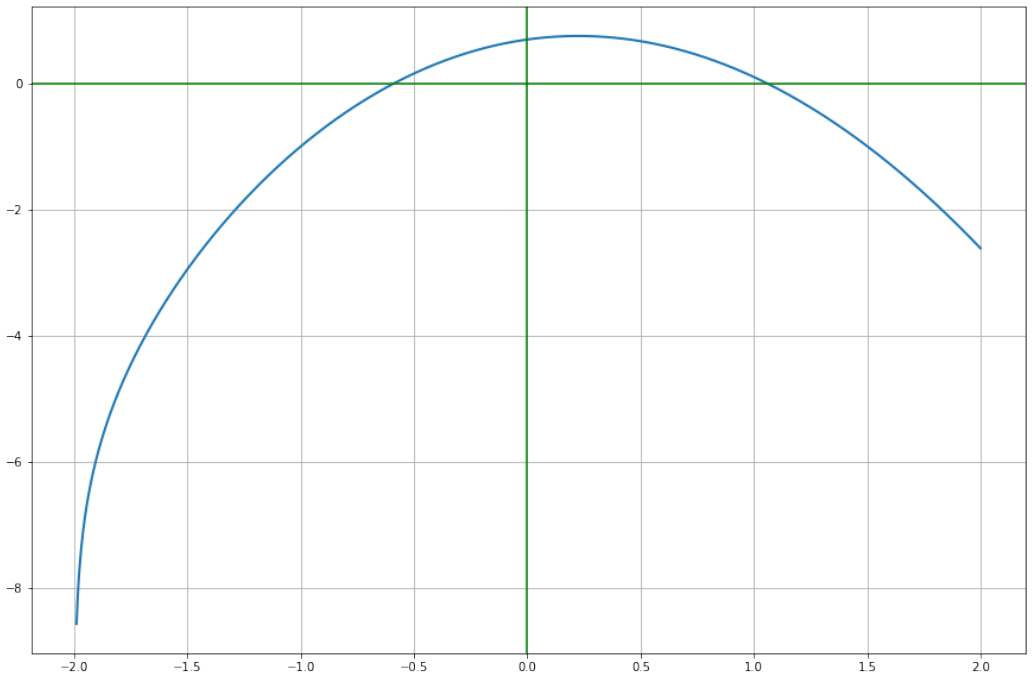

Рис. 1: График функции $f(x)$

При помощи метода дихотомии, уточним значение корня до $\epsilon = 10^{-2}$.

k	a_k	b_k	$f(a_k)$	$f(b_k)$	$\frac{a_k+b_k}{2}$	$f(\frac{a_k+b_k}{2})$	ϵ_k
0	-1	-0.5	-1	0.155465	-0.75	-0.339356	0.25
1	-0.75	-0.5	-0.339356	0.155465	-0.625	-0.0721713	0.125
2	-0.625	-0.5	-0.0721713	0.155465	-0.5625	0.0464992	0.0625
3	-0.625	-0.5625	-0.0721713	0.0464992	-0.59375	-0.0116125	0.03125
4	-0.59375	-0.5625	-0.0116125	0.0464992	-0.578125	0.0177479	0.015625
5	-0.59375	-0.578125	-0.0116125	0.0177479	-0.585938	0.00314401	0.0078125

Таблица 1: Метод половинного деления

Для уточнения значения корня до $\epsilon = 10^{-7}$ воспользуемся методам простых итераций и методом Ньютона.

Метод простых итераций. Поскольку корень x_∞ локализован на отрезке $[-1; -0.5]$, то для функции $\phi(x)$ достаточно проверить выполнение условия (4).

Приведём функцию (1) к виду, пригодному для итераций:

$$\phi(x) = x + 0.3(ln(x+2) - x^2)$$

Проверим выполнение условия (4):

$$\phi'(x) = 1 + \frac{0.3}{x+2} - 0.6 * x \tag{7}$$

Найдём максимум (7) на отрезке $x \in [-1; -0.5]$

$$\phi''(x) \neq 0,\ x \in [-1; -0.5]$$

$$\phi'(-1) = 0.43 \qquad\qquad \phi'(-0.5) = 0.44$$

Условие (4) выполняется $\Rightarrow$ итерационный процесс метода простых итераций сходится.

Метод Ньютона. Возьмем начальное приближение $x_0 = -0.5859375$ Для применения (6) найдём $f'(x), f''(x)$ и проверим условия теоремы 2:

$$f'(x) = \frac{1}{x+2} - 2x$$
$$f''(x) = -\frac{1}{(x+2)^2} - 2$$

Функция дважды непрерывно дифференцирумая на отрезке $[x_0; x_0 + 2h]$, $h_0 = -0.001673, M = -2.5024$

$$2 * 0.001673 * 2.5024 \leq 1.879 \equiv 0.00837 \leq 1.879$$

Следовательно можно построить итерационный процесс по методу Ньютона

k	Метод простой итерации		Метод Ньютона	
	x_k	ϵ_k	x_k	ϵ_k
0	-0.5859375	∞	-0.5859375	∞
1	-0.5868807	0.0009432	-0.5876106865525098	0.0016731865525098
2	-0.5872919	0.0004112	-0.5876088279784095	1.8585741002885e-08
3	-0.5874709	0.0001790	-0.5876088279761155	2.2940538357829e-10
4	-0.5875488	0.0000779	-	-
5	-0.5875827	0.0000339	-	-
6	-0.5875975	0.0000147	-	-
7	-0.5876039	0.0000064	-	-
8	-0.5876067	0.0000028	-	-
9	-0.5876079	0.0000012	-	-
10	-0.5876084	0.0000005	-	-
11	-0.5876087	0.0000002	-	-
12	-0.5876088	0.0000001	-	-

Таблица 2: Метод простой итерации и метод Ньютона

5 Выводы

Для решения нелинейных уравнений применяются метод Ньютона и метод простых итераций(МПИ), МПИ сходится со скоростью геометрической прогрессии со знаменателем q, а метон Ньютона обладает квадратичной скоростью сходимости, что сразу заметно в Таблице 2