



МИНОБРНАУКИ РОССИИ

**Федеральное государственное бюджетное образовательное
учреждение высшего образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ»**

Факультет информатики и прикладной математики

Кафедра прикладной математики и экономико-математических методов

ЛАБОРАТОРНАЯ РАБОТА

на тему:

“Решения нелинейного уравнения (2.1.10)”

Направление: 01.03.02 Прикладная математика и информатика

Обучающийся: Попова Софья Ивановна

Группа: ПМ-1901

Подпись: _____

Проверил: Хазанов Владимир Борисович

Должность: Профессор

Оценка: _____

Дата: _____

Подпись: _____

Санкт-Петербург
2021

Оглавление

Необходимые формулы	3
Исходные данные	4
Программа:	5
Выходные данные:	6
Оценка точности полученного результата:.....	7
Вывод	7

Необходимые формулы

Методы Паде-аппроксимации

В основе методов лежат два вида Паде-аппроксимации функции $x = g(y)$ обратной к функции $y = f(x)$.

2.1.10. $x = [0/1]_g(y)$

2.1.11. Метод Галлея (Хэлли) $x = [1/1]_g(y)$

x_0 – начальное приближение

$$f_k = f(x_k), f'_k = f'(x_k), f''_k = f''(x_k)$$

$$t_k = -\frac{f_k}{f'_k}$$

$$r_k = \frac{f''_k}{f'_k} t_k^2$$

$$x_{k+1} = \frac{x_k^2}{x_k - t_k}, k=1,2,3,\dots$$

$$x_{k+1} = x_k + \frac{t_k^2}{t_k + \frac{1}{2}r_k}, k=1,2,3,\dots$$

Рисунок 1 - Формулы

Исходные данные

10	$f(x) = x^3 + 6x^2 + 9x - 4$
-----------	------------------------------

Рисунок 2 – исходные данные

Программа:

```
Clear[Programm];
```

|очистить

```
Programm[f_, k_] := Module[{x = {k}, n = 100, i = 1}, For[i = 1, i ≤ n, i++, x = Append[x, {}]; x[[i + 1]] =  $\frac{x[[i]]^2}{x[[i]] - \frac{10-10x[[i]]+4x[[i]]^2-x[[i]]^3}{10-8x[[i]]+3x[[i]]^2}}$ ];
```

|программный модуль |цикл для |добавить в конец

```
If[x[[i]] == x[[i + 1]], Break[]];
```

|условный оператор |прервать цикл

```
Print["x=", x[[i]], ", кол-во итераций = ", i]
```

|печатать

Рисунок 3 – реализация

Выходные данные:

```
Programm[ $x^3 - 4x^2 + 10x - 10$ , 1208.]
```

$x=1.62936$, кол-во итераций = 29

Рисунок 4 – решение

```
NSolve[0 ==  $x^3 - 4x^2 + 10x - 10$ , {x}] [[3]]  
[численное решение уравнений]  
{x → 1.62936}
```

Рисунок 5 – решение встроенной функции

Полученный результат совпадает с встроенной функцией Wolfram Mathematica .

Оценка точности полученного результата:

```
r1 = Programm[ $x^3 - 4x^2 + 10x - 10$ , 1208.]
```

```
1.62936
```

```
r = NSolve[ $0 == x^3 - 4x^2 + 10x - 10$ ] [[3]]  
[численное решение уравнений]
```

```
{x → 1.62936}
```

```
r[[1, 2]] - r1
```

```
 $-2.88658 \times 10^{-15}$ 
```

Рисунок 6 – оценка точности