

**Московский авиационный институт
(национальный исследовательский
университет)**

Институт №3.

«Системы управления, информатика и электроэнергетика»

Кафедра №304

**«Автоматизированные системы обработки информации и
управления»**

**Отчет по домашней работе
по учебной дисциплине
«Численные методы»**

Группа М30-207Б

Выполнил:

Гордеев Н.М.

Принял:

Яганов В.М.

Содержание

Задание №1 (20.03.20).....	4
Решение I.....	5
Решение II.....	5
Решение III. (Для решения написал программу)	6
Код программы.....	6
Результат работы программы	8
Задание №2 (03.04.20).....	10
Решение задачи №1	11
I.....	11
II. Для решения дописал ранее написанную программу.....	11
Дописанный код	11
Результат работы программы	12
III. Для решения дописал ранее дописанную программу	13
Дописанный код	13
Результат работы программы	14
IV	14
Решение задачи №2	15
I.....	15
II. Для решения дописал ранее написанную программу.....	15
Дописанный код	15
Результат работы программы	15
III. Для решения дописал ранее дописанную программу	16
Дописанный код	16
Результат работы программы	16
IV	16
Задание №3 (17.04.20)	17
1. Для решения дописал ранее написанную программу.....	18
Дописанный код	18
Результат работы программы	19
2. Для решения дописал ранее написанную программу.....	20
Дописанный код	20
Результат работы программы	20
3. Для решения дописал ранее написанную программу.....	21
Результат работы программы	21
Задание №4 (24.04.20.)	22

1. Для решения дописал ранее написанную программу	23
Дописанный код	23
Результат работы программы	24
2. Для решения дописал ранее написанную программу	25
Дописанный код	25
Результат работы программы	25
Задание №5.....	26
1. Для решения дописал ранее написанную программу	27
Дописанный код	27
Результат работы программы	28
Прямая и кривая в той же системе координат, где нанесены данные точки	28
Задача 2.	29

Задание №1 (20.03.20)

I. Выделить аналитическим способом корни следующих нелинейных уравнений:

а) $f(x) = 5^x - 6x - 3 = 0$

б) $f(x) = x^3 - 6x + 2 = 0$

в) $f(x) = x^3 + 2x^2 - 0.75x - 1 = 0$

Метод. указания:

1. Выделить, значит определить интервалы переменной X , в каждом из которых заведомо находится корень (один) данного уравнения.

2. Аналитически – это значит на границах искомых интервалов $f(x)$ должна иметь разные знаки.

3. С помощью 1-х производных этот процесс можно ускорить:

а) находятся значения X , в которых эти производные равны 0;

б) в этих точках и в $\pm\infty$ определяются знаки $f(x)$;

в) затем эти "грубые" (т.е. большие) интервалы уменьшаются путем банального пересчета $f(x)$.

II. Выделить графическим способом корни следующих нелинейных уравнений:

а) $f(x) = \sin x - x^2 = 0$

б) $f(x) = 3x + \cos x + 1 = 0$

Метод. указания:

1. Привести $f(x)$, например, к виду $x = \varphi(x)$.

2. В системе Oxy вычерчиваются две функции: $y_1 = x$ и $y_2 = \varphi(x)$

Точка их пересечения и будет искомым корнем (примерно!!!)

III. Найти положительные корни для уравнений I. (а,в) и II. (а,б) методом половинного деления. (Алгоритм, см. лекции)

Точность вычисления корней $\varepsilon = 0.01$

Решение I.

а) $f(x) = 5^x - 6x - 3 = 0$

$$f'(x) = 5^x \ln 5 - 6 = 0$$

$$5^x \ln 5 = 6$$

$$X = \log_5 \frac{6}{\ln 5} \approx 0.8176; \Rightarrow \gamma f(x) \text{ 2 корня}$$

$$f(1) = 5 - 6 - 3 = -4 < 0$$

$$f(-1) = \frac{1}{5} + 6 - 3 = 3.2 > 0 \quad X_0 \in (-1; 1)$$

$$f(2) = 25 + -12 - 3 = 10 > 0 \quad X_1 \in (1; 2)$$

б) $f(x) = x^3 - 6x + 2 = 0$

$$f'(x) = 3x^2 - 6 = 0$$

$$x^2 = 2$$

$$X = \pm \sqrt{2} \Rightarrow \gamma f(x) \text{ 3 корня}$$

$$f(-3) = -27 + 18 + 2 = -7 < 0$$

$$f(-2) = -8 + 12 + 2 = 10 > 0 \quad X_0 \in (-3; -2)$$

$$f(1) = 1 - 6 + 2 = -3 < 0 \quad X_1 \in (-2; 1)$$

$$f(3) = 27 - 18 + 2 = 11 > 0 \quad X_2 \in (1; 3)$$

в) $f(x) = x^3 + 2x^2 - 0.75x - 1 = 0$

$$f'(x) = 3x^2 + 4x - 0.75 = 0$$

$$X_1 = -1.5; X_2 \approx 0.1667 \Rightarrow \gamma f(x) \text{ 3 корня}$$

$$f(-3) = -27 + 18 + 2.25 - 1 = -7.75 < 0$$

$$f(-2) = -8 + 8 + 1.5 - 1 = 0.5 > 0 \quad X_0 \in (-3; -2)$$

$$f(0) = -1 = -1 < 0 \quad X_1 \in (-2; 0)$$

$$f(1) = 1 + 2 - 0.75 - 1 = 1.25 > 0 \quad X_2 \in (0; 1)$$

Решение II.

а) $f(x) = \sin x - x^2 = 0$

$$x^2 = \sin x;$$

$$X_1 = 0$$

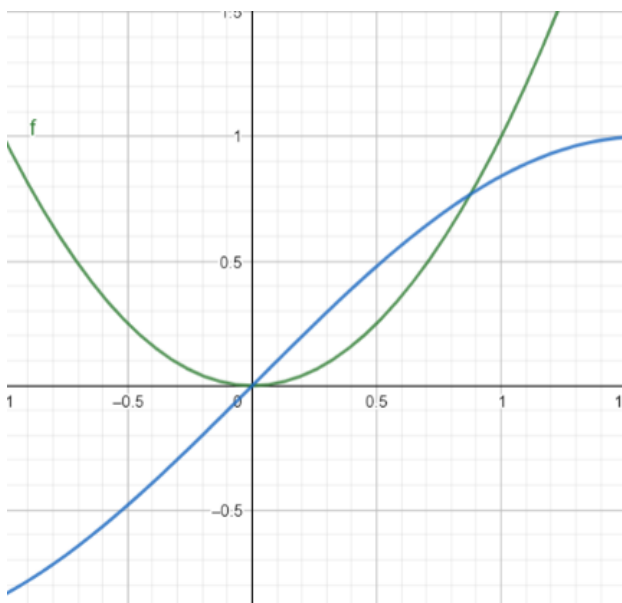
$$X_2 \approx 0.8$$

б) $f(x) = 3x + \cos x + 1 = 0$

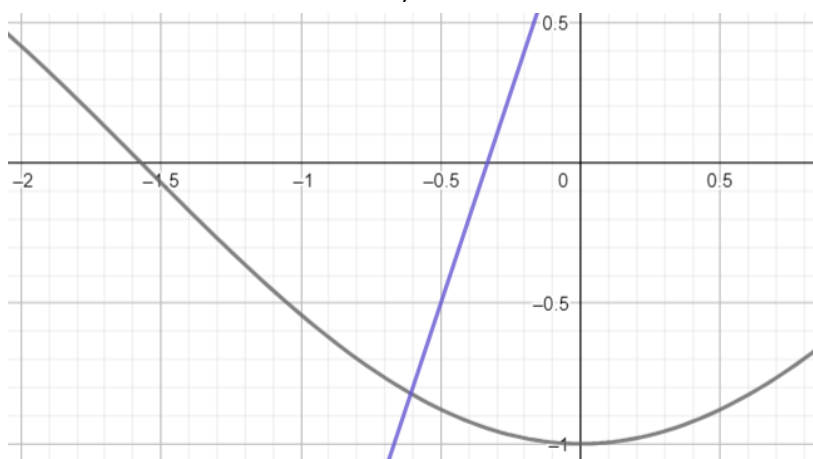
$$3x + 1 = -\cos x$$

$$X \approx -0.6$$

а)



б)



Решение III. (Для решения написал программу)

Код программы

```
#include <iostream>
#include <iomanip>

void a() { setlocale(LC_ALL, "C"); }
void r() { setlocale(LC_ALL, "Russian"); }
void N_tabl(int n, char** s);
void C_tabl(int n, double* N);
void K_tabl(int n, double* N);

double Fx1(double x) { return pow(5, x) - 6 * x - 3; }
double Fx2(double x) { return pow(x, 3) - 6 * x + 2; }
double Fx3(double x) { return pow(x, 3) + 2 * x * x - 0.75 * x - 1; }
double Fx4(double x) { return sin(x) - x * x; }
double Fx5(double x) { return 3 * x + cos(x) + 1; }

double ПоловинноеДеление(double a, double b, double (*F)(double), double Eps);

using namespace std;
int main()
{
    setlocale(LC_ALL, "Russian"); // подключаю русский язык
    double a = 5; // верхняя граница значений x
    double b = 0; // нижняя граница значений x
    double x = a; // переменная

    cout << "1\nКорни 1 ур-я\n";
    cout << "Ответ: x = " << ПоловинноеДеление(-1, 1, Fx1, 0.01) << endl;
    cout << "Ответ: x = " << ПоловинноеДеление(1, 2, Fx1, 0.01) << endl;
    cout << "\nКорни 2 ур-я\n";
    cout << "Ответ: x = " << ПоловинноеДеление(-3, -2, Fx2, 0.01) << endl;
    cout << "Ответ: x = " << ПоловинноеДеление(-2, 1, Fx2, 0.01) << endl;
    cout << "Ответ: x = " << ПоловинноеДеление(1, 3, Fx2, 0.01) << endl;
    cout << "\nКорни 3 ур-я\n";
    cout << "Ответ: x = " << ПоловинноеДеление(-3, -2, Fx3, 0.01) << endl;
    cout << "Ответ: x = " << ПоловинноеДеление(-2, 0, Fx3, 0.01) << endl;
    cout << "Ответ: x = " << ПоловинноеДеление(0, 1, Fx3, 0.01) << endl;
    cout << "\n2\nКорни 1 ур-я\n";
    cout << "Ответ: x = " << ПоловинноеДеление(0.5, 1, Fx4, 0.01) << endl;
    cout << "\nКорни 2 ур-я\n";
    cout << "Ответ: x = " << ПоловинноеДеление(-1, 0, Fx5, 0.01) << endl;
}

double ПоловинноеДеление(double a1, double b1, double (*F)(double), double Eps) {
    const int n = 5;
    int i = 0;
    double N[n];
    char** s;
    s = new char* [n];
    for (int i = 0; i < n; i++)
        s[i] = new char[15];

    char s0[15] = { " i " };
    char s1[15] = { " a " };
    char s2[15] = { " b " };
    char s3[15] = { " x " };
    char s4[15] = { " f(x) " };
    s[0] = s0;
```

```

s[1] = s1;
s[2] = s2;
s[3] = s3;
s[4] = s4;
N_tabl(n, s);
N[0] = i;
N[1] = a1;
N[2] = b1;
N[3] = a1;
N[4] = F(a1);

double a = a1;
double b = b1;
double x = a;
if (F(a) < F(b)) { a = b; b = x; x = a; }
while (abs(F(x)) > Eps)
{
    C_tabl(n, N);

    i++;
    if (F(x) < 0)
        b = x;
    else
        a = x;
    x = (a + b) / 2;

    N[0] = i;
    N[1] = a;
    N[2] = b;
    N[3] = x;
    N[4] = F(x);
}
K_tabl(n, N);
return x;
}

void N_tabl(int n, char** s)
{
    a()); cout << char(218) << setfill(char(196)) << setw(15);
    for (int i = 0; i < n - 1; i++)
    {
        cout << char(194) << setfill(char(196)) << setw(15);
    }
    cout << char(191) << endl;

    for (int i = 0; i < n; i++)
    {
        cout << char(179); r(); cout << s[i]; a();
    }
    cout << char(179) << endl;

    cout << char(195) << setfill(char(196)) << setw(15);
    for (int i = 0; i < n - 1; i++)
    {
        cout << char(197) << setfill(char(196)) << setw(15);
    }
    cout << char(180) << endl;
}

void C_tabl(int n, double* N) {
    a()); for (int i = 0; i < n; i++)

```

```

{
    cout << char(179) << " " << setfill(char(32)) << setw(12) << N[i] << " ";
}
cout << char(179) << endl;

cout << char(195);
cout << setfill(char(196)) << setw(15);
for (int i = 0; i < n - 1; i++)
{
    cout << char(197) << setfill(char(196)) << setw(15);
}
cout << char(180) << endl;
}

void K_tabl(int n, double* N) {
    a(); for (int i = 0; i < n; i++)
    {
        cout << char(179) << " " << setfill(char(32)) << setw(12) << N[i] << " ";
    }
    cout << char(179) << endl;

    cout << char(192);
    cout << setfill(char(196)) << setw(15);
    for (int i = 0; i < n - 1; i++)
    {
        cout << char(193) << setfill(char(196)) << setw(15);
    }
    cout << char(217) << endl; r();
}
}

```

Результат работы программы

Консоль отладки Microsoft Visual Studio

1
Корни 1 ур-я

i	a	b	x	f(x)
0	-1	1	-1	3.2
1	-1	1	0	-2
2	-1	0	-0.5	0.447214
3	-0.5	0	-0.25	-0.83126
4	-0.5	-0.25	-0.375	-0.203127
5	-0.5	-0.375	-0.4375	0.119539
6	-0.4375	-0.375	-0.40625	-0.042452
7	-0.4375	-0.40625	-0.421875	0.0383832
8	-0.421875	-0.40625	-0.414063	-0.002075

Ответ: x = -0.414063

i	a	b	x	f(x)
0	1	2	1	-4
1	2	1	1.5	-0.81966
2	2	1.5	1.75	3.21851
3	1.75	1.5	1.625	0.921818
4	1.625	1.5	1.5625	-0.0115228
5	1.625	1.5625	1.59375	0.4387
6	1.59375	1.5625	1.57813	0.20958
7	1.57813	1.5625	1.57031	0.0980387
8	1.57031	1.5625	1.56641	0.0430121
9	1.56641	1.5625	1.56445	0.0156834
10	1.56445	1.5625	1.56348	0.00206497

Ответ: x = 1.56348

Ответ: x = 1.56348

Корни 2 ур-я

i	a	b	x	f(x)
0	-3	-2	-3	-7
1	-2	-3	-2.5	1.375
2	-2.5	-3	-2.75	-2.29688
3	-2.5	-2.75	-2.625	-0.337891
4	-2.5	-2.625	-2.5625	0.548584
5	-2.5625	-2.625	-2.59375	0.112946
6	-2.59375	-2.625	-2.60938	-0.110561
7	-2.59375	-2.60938	-2.60156	0.00166845

Ответ: x = -2.60156

i	a	b	x	f(x)
0	-2	1	-2	6
1	-2	1	-0.5	4.875
2	-0.5	1	0.25	0.515625
3	0.25	1	0.625	-1.50586
4	0.25	0.625	0.4375	-0.54126
5	0.25	0.4375	0.34375	-0.0218811
6	0.25	0.34375	0.296875	0.244915
7	0.296875	0.34375	0.320313	0.110989
8	0.320313	0.34375	0.332031	0.0444172
9	0.332031	0.34375	0.337891	0.0112332
10	0.337891	0.34375	0.34082	-0.0053327

Ответ: x = 0.34082

Ответ: x = 0.34082

i	a	b	x	f(x)
0	1	3	1	-3
1	3	1	2	-2
2	3	2	2.5	2.625
3	2.5	2	2.25	-0.109375
4	2.5	2.25	2.375	1.14648
5	2.375	2.25	2.3125	0.491455
6	2.3125	2.25	2.28125	0.184357
7	2.28125	2.25	2.26563	0.0358315
8	2.26563	2.25	2.25781	-0.0371852
9	2.26563	2.25781	2.26172	-0.000780404

Ответ: x = 2.26172

Корни 3 ур-я

i	a	b	x	f(x)
0	-3	-2	-3	-7.75
1	-2	-3	-2.5	-2.25
2	-2	-2.5	-2.25	-0.578125
3	-2	-2.25	-2.125	0.0292969
4	-2.125	-2.25	-2.1875	-0.256592
5	-2.125	-2.1875	-2.15625	-0.109283
6	-2.125	-2.15625	-2.14063	-0.0389137
7	-2.125	-2.14063	-2.13281	-0.00453997

Ответ: x = -2.13281

Ответ: x = 0.875

Корни 2 ур-я

i	a	b	x	f(x)
0	-1	0	-1	-1.4597
1	0	-1	-0.5	0.377583
2	-0.5	-1	-0.75	-0.518311
3	-0.5	-0.75	-0.625	-0.0640369
4	-0.5	-0.625	-0.5625	0.158424
5	-0.5625	-0.625	-0.59375	0.0475985
6	-0.59375	-0.625	-0.609375	-0.0081191

Ответ: x = -0.609375

Ответ: x = -2.13281

i	a	b	x	f(x)
0	-2	0	-2	0.5
1	-2	0	-1	0.75
2	-1	0	-0.5	-0.25
3	-1	-0.5	-0.75	0.265625
4	-0.75	-0.5	-0.625	0.00585938

Ответ: x = -0.625

i	a	b	x	f(x)
0	0	1	0	-1
1	1	0	0.5	-0.75
2	1	0.5	0.75	-0.015625
3	1	0.75	0.875	0.544922
4	0.875	0.75	0.8125	0.247314
5	0.8125	0.75	0.78125	0.111603
6	0.78125	0.75	0.765625	0.0469398
7	0.765625	0.75	0.757813	0.0153966
8	0.757813	0.75	0.753906	-0.000179231

Ответ: x = 0.753906

2

Корни 1 ур-я

i	a	b	x	f(x)
0	0.5	1	0.5	0.229426
1	0.5	1	0.75	0.119139
2	0.75	1	0.875	0.0019185

Ответ: x = 0.875

Задание №2 (03.04.20)

Тема: Метод простой итерации и метод Ньютона.

Задача №1

$$f(x) = x^4 - 2x - 4 = 0$$

- I. Выделить аналитическим способом **один положительный** корень.
- II. Уточнить этот корень методом простой итерации с точностью $\varepsilon = 0.001$

Метод. указания:

Привести уравнение $f(x)=0$ любым способом к итерационной формуле вида $x^{k+1} = \varphi(x^k)$.

Чтобы при этом выполнялось условие сходимости (см. лекцию), эту итерационную формулу удобно записать в виде $x = x - \theta f(x)$, где $\theta = 1/\max(f'(x))$. Максимальное значение $f'(x)$ ищется на выделенном интервале. Тогда тут $\varphi(x) = x - \theta f(x)$.

- III. Решить задачу №1 методом Ньютона с точностью $\varepsilon = 0.001$.

(Проверить задачу на сходимость!! См. лекцию)

- IV. Сравнить результаты полученные двумя разными методами.

Задача № 2.

$$f(x) = \sin 2x - x^2 = 0$$

- I. Выделить графическим способом корень нелинейного уравнения.
- II. Решить задачу методом простой итерации с точностью $\varepsilon = 0.001$

(С предварительной проверкой на сходимость!)

- III. Решить задачу № 2 методом Ньютона с точностью $\varepsilon = 0.001$.

(С предварительной проверкой на сходимость!)

- IV. Сравнить результаты.

Решение задачи №1

$$f(x) = x^4 - 2x - 4 = 0$$

I.

$$f(1) = -5$$

$$f(2) = 8$$

Ответ: $X_0 \in (1; 2)$

II. Для решения дописал ранее написанную программу

$$f'(x) = 4x^3 - 2$$

$$f'(1) = 1 - 2 = -1$$

$$f'(2) = 32 - 2 = 30$$

$$\lambda = -2/30 = -0.0667$$

Дописанный код

Глобальная область:

```
double Fx6(double x) { return pow(x, 4) - 2 * x - 4; }
```

```
double ПростойИтерации(double x, double a, double (*F)(double), double Eps);
```

main:

```
cout << "Ответ: x = " << ПростойИтерации(2, -0.0667, Fx6, 0.001) << endl;
```

Реализация ф-ий:

```
double ПростойИтерации(double x1, double a, double (*F)(double), double Eps) {
    const int n = 5;
    int i = 0;
    double N[n];
    char** s;
    s = new char* [n];
    for (int i = 0; i < n; i++)
        s[i] = new char[15];
    char s0[15] = { "   i   " };
    char s1[15] = { "   x   " };
    char s2[15] = { "   a   " };
    char s3[15] = { "  f(x)  " };
    char s4[15] = { " fl(x)  " };
    s[0] = s0;
    s[1] = s1;
    s[2] = s2;
    s[3] = s3;
    s[4] = s4;
    N_tabl(n, s);
    N[0] = i;
    N[1] = x1;
    N[2] = a;
    N[3] = F(x1);
    N[4] = x1 + a * F(x1);

    double x = x1;
    while (abs(F(x)) > Eps)
    {
        C_tabl(n, N);

        i++;
        x = x + a * F(x);

        N[0] = i;
```


```

        N[1] = x;
        N[2] = a;
        N[3] = F(x);
        N[4] = x + a * F(x);

    }
    K_tabl(n, N);
    return x;
}

```

Результат работы программы

 Консоль отладки Microsoft Visual Studio

i	x	a	f(x)	f1(x)
0	2	-0.0667	8	1.4664
1	1.4664	-0.0667	-2.30889	1.6204
2	1.6204	-0.0667	-0.34648	1.64351
3	1.64351	-0.0667	0.00910194	1.64291
4	1.64291	-0.0667	-0.000458395	1.64294

Ответ: x = 1.64291

III. Для решения дописал ранее дописанную программу

Задача №1

$$f(x) = x^4 - 2x - 4 = 0$$

$$f'(x) = 4x^3 - 2$$

Дописанный код

Глобальная область:

```
double Fx16(double x) { return 4*pow(x, 3) - 2; }
```

```
double Ньютон(double x1, double (*F)(double), double (*F1)(double), double Eps);
```

main:

```
cout << "Ответ: x = " << Ньютон(2, Fx6, Fx16, 0.001) << endl;
```

Реализация ф-ий:

```
double Ньютон(double x1, double (*F)(double), double (*F1)(double), double Eps) {  
    const int n = 5;  
    int i = 0;  
    double N[n];  
    char** s;  
    s = new char* [n];  
    for (int i = 0; i < n; i++)  
        s[i] = new char[15];  
    char s0[15] = { " i " };  
    char s1[15] = { " xn " };  
    char s2[15] = { " f(x) " };  
    char s3[15] = { " f1(x) " };  
    char s4[15] = { " xn+1 " };  
    s[0] = s0;  
    s[1] = s1;  
    s[2] = s2;  
    s[3] = s3;  
    s[4] = s4;  
    N_tabl(n, s);  
    N[0] = i;  
    N[1] = x1;  
    N[2] = F(x1);  
    N[3] = F1(x1);  
    N[4] = x1 - F(x1) / F1(x1);  
  
    double x = x1;  
    while (abs(F(x)) > Eps)  
    {  
        C_tabl(n, N);  
  
        i++;  
        x = x - F(x) / F1(x);  
  
        N[0] = i;  
        N[1] = x;  
        N[2] = F(x);  
        N[3] = F1(x);  
        N[4] = x - F(x) / F1(x);  
    }  
    K_tabl(n, N);  
    return x;  
}
```

Результат работы программы

i	x_n	$f(x)$	$f'(x)$	x_{n+1}
0	2	8	30	1.73333
1	1.73333	1.56002	18.8308	1.65049
2	1.65049	0.119824	15.9845	1.64299
3	1.64299	0.0009157	15.7406	1.64293

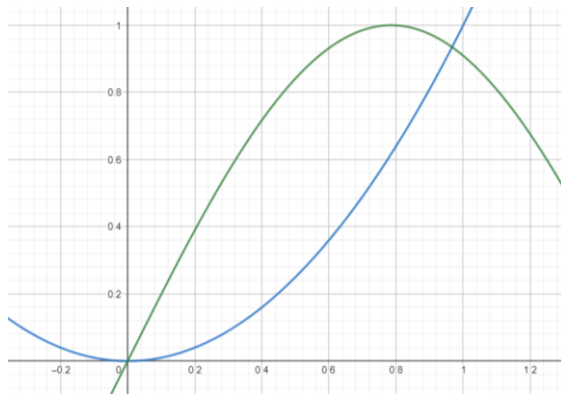
Ответ: $x = 1.64299$

IV

Метод ньютона показал более быструю сходимость, чем метод простой итерации.

Решение задачи №2

$$f(x) = \sin 2x - x^2 = 0$$



Ответ: $x_0 \in (\pi/3; 1)$

II. Для решения дописал ранее написанную программу

$$f'(x) = 2\cos 2x - 2x$$

$$f'(1) = -2.8$$

$$f'(\pi/3) = -1 - 2\pi/3 = -3$$

$$\lambda = 2/3 = 0.667$$

Дописанный код

Глобальная область:

```
double Fx7(double x) { return sin(2*x) - x * x; }
```

main:

```
cout << "Ответ: x = " << ПростойИтерации(1.05, 0.667, Fx7, 0.001) << endl;
```

Результат работы программы

i	x	a	f(x)	φ(x)
0	1.05	0.667	-0.239291	0.890393
1	0.890393	0.667	0.185233	1.01394
2	1.01394	0.667	-0.130741	0.926739
3	0.926739	0.667	0.101466	0.994417
4	0.994417	0.667	-0.0749767	0.944407
5	0.944407	0.667	0.0579522	0.983061
6	0.983061	0.667	-0.0435384	0.954021
7	0.954021	0.667	0.0335132	0.976374
8	0.976374	0.667	-0.0253684	0.959454
9	0.959454	0.667	0.0194674	0.972438
10	0.972438	0.667	-0.0147925	0.962572
11	0.962572	0.667	0.0113286	0.970128
12	0.970128	0.667	-0.00862577	0.964375
13	0.964375	0.667	0.00659758	0.968775
14	0.968775	0.667	-0.0050292	0.965421
15	0.965421	0.667	0.00384374	0.967985
16	0.967985	0.667	-0.00293189	0.966029
17	0.966029	0.667	0.00223977	0.967523
18	0.967523	0.667	-0.00170906	0.966383
19	0.966383	0.667	0.00130526	0.967254
20	0.967254	0.667	-0.00099619	0.966589

Ответ: $x = 0.967254$

III. Для решения дописал ранее дописанную программу

Задача №1

$$f(x) = \sin 2x - x^2 = 0$$

$$f'(x) = 2\cos 2x - 2x$$

Дописанный код

Глобальная область:

```
double Fx17(double x) { return 2 * cos(2*x) - 2*x; }
```

main:

```
cout << "Ответ: x = " << Ньютон(1.05, Fx7, Fx17, 0.001) << endl;
```

Результат работы программы

i	xn	f(x)	f1(x)	xn+1
0	1.05	-0.239291	-3.10969	0.97305
1	0.97305	-0.0164301	-2.67921	0.966918
2	0.966918	-0.000107697	-2.64407	0.966877

Ответ: x = 0.966918

IV

Метод ньютона показал более быструю сходимость, чем метод простой итерации.

Задание №3 (17.04.20)

1. Решить нелинейную задачу из предыдущего задания, т.е. $f(x) = x^4 - 2x - 4 = 0$ методом хорд.
Точность $\varepsilon = 0.001$

2. Найти корни методом простых итераций для след. системы:

$$\sin(x - 0.6) - y = 1.6$$

$$3x - \cos y = 0.9$$

Точность $\varepsilon = 0.001$

3. Методом Ньютона найти корни в системе уравнений:

$$x + y - 3 = 0$$

$$x^2 + y^2 - 9 = 0$$

Точность $\varepsilon = 0.001$

1. Для решения дописал ранее написанную программу

Дописанный код

Глобальная область:

```
double МетодХорд(double a, double b, double (*F)(double), double Eps);
```

main:

```
cout << "Ответ: x = " << МетодХорд(1, 2, Fx6, 0.01) << endl;
```

Реализация ф-ий:

```
double МетодХорд(double a1, double b1, double (*F)(double), double Eps) {
    const int n = 5;
    int i = 0;
    double N[n];
    char** s;
    s = new char* [n];
    for (int i = 0; i < n; i++)
        s[i] = new char[15];

    char s0[15] = { "   i   " };
    char s1[15] = { "   a   " };
    char s2[15] = { "   b   " };
    char s3[15] = { "   x   " };
    char s4[15] = { " f(x)  " };
    s[0] = s0;
    s[1] = s1;
    s[2] = s2;
    s[3] = s3;
    s[4] = s4;
    N_tabl(n, s);
    N[0] = i;
    N[1] = a1;
    N[2] = b1;
    N[3] = a1;
    N[4] = F(a1);
    double a = a1;
    double b = b1;
    double x = b;
    bool t = 0;
    if (a < b) { b = a; a = x; x = b; }
    if (F(a) > 0) t = 1;
    while (abs(F(x)) > Eps)
    {
        C_tabl(n, N);

        i++;
        if(t)
            x = x - F(x) / (F(x) - F(a)) * (x - a);
        else
            x = x - F(x) / (F(b) - F(x)) * (b - x);
        N[0] = i;
        N[1] = a;
        N[2] = b;
        N[3] = x;
        N[4] = F(x);
    }
    K_tabl(n, N);
    return x;
}
```

Результат работы программы

Консоль отладки Microsoft Visual Studio

i	a	b	x	f(x)
0	1	2	1	-5
1	2	1	1.38462	-3.09373
2	2	1	1.55623	-1.24711
3	2	1	1.61608	-0.411138
4	2	1	1.63484	-0.126281
5	2	1	1.64052	-0.0379349
6	2	1	1.64222	-0.0113193
7	2	1	1.64272	-0.00337074

Ответ: x = 1.64272

2. Для решения дописал ранее написанную программу

$$\sin(x - 0.6) - y = 1.6$$
$$3x - \cos y = 0.9$$

$$\sin(x - 0.6) - 1.6 = y$$
$$x = 0.3 + \cos y / 3$$

Подставляю ϕ -ю в программу $y = \sin(0.3 + \cos x / 3 - 0.6) - 1.6 - x$
 $Y(0) < 0$
 $Y(1) > 0$

Дописанный код

Глобальная область:

```
double Fx8(double x) { return sin(0.3 + cos(x)/3 - 0.6) - 1.6 - x; }
```

main:

```
cout << "Ответ: y = " << ПростойИтерации(1, 1, Fx8, 0.001) << endl;
```

Результат работы программы

Консоль отладки Microsoft Visual Studio				
i	x	a	f(x)	$\phi(x)$
0	1	1	-2.71961	-1.71961
1	-1.71961	1	-0.222743	-1.94236
2	-1.94236	1	-0.066339	-2.00869
3	-2.00869	1	-0.0184623	-2.02716
4	-2.02716	1	-0.00501071	-2.03217
5	-2.03217	1	-0.00134991	-2.03352
6	-2.03352	1	-0.000362935	-2.03388

Ответ: y = -2.03352

$$x = 0.3 + \cos(-2.03352) / 3 = 0.1512$$

3. Для решения дописал ранее написанную программу

$$x + y - 3 = 0$$
$$x^2 + y^2 - 9 = 0$$

Вводим уравнение $X^2 + (3-x)^2 - 9 = y$
 $y = 2x - 2(3-x)$

```
double Fx9(double x) { return x*x + (3 - x)*(3 - x) - 9; }  
double Fx19(double x) { return 2x - 2*(3 - x); }  
  
cout << "Ответ: x = " << Ньютон(1, Fx9, Fx19, 0.001) << endl;  
cout << "Ответ: x = " << Ньютон(4, Fx9, Fx19, 0.001) << endl;
```

Результат работы программы

Консоль отладки Microsoft Visual Studio

i	xn	f(x)	f1(x)	xn+1
0	1	-4	-2	-1
1	-1	8	-10	-0.2
2	-0.2	1.28	-6.8	-0.0117647
3	-0.0117647	0.0708651	-6.04706	-4.57771e-05
4	-4.57771e-05	0.000274667	-6.00018	-6.98492e-10

Ответ: x = -4.57771e-05

i	xn	f(x)	f1(x)	xn+1
0	4	8	10	3.2
1	3.2	1.28	6.8	3.01176
2	3.01176	0.0708651	6.04706	3.00005
3	3.00005	0.000274667	6.00018	3

Ответ: x = 3.00005

Корни очевидны.

Ответ (0;3), (3;0).

Задание №4 (24.04.20.)

№1

Заданы значения функции $f(x)$ в узлах x_i . Найти значения функции $f(x)$ при $x_1 = 1,1$ с помощью интерполяционных формул Ньютона.

x_i	1	2
1,0	0,8	1,1
1,2	1,8	2,2
1,4	2,9	3,0
1,6	4,0	4,1
1,8	4,9	4,9
2,0	6,1	5,9

№2

Заданы значения y_i функции $f(x)$ в точках x_i . Найти значение функции $f(x)$ при $x = x^*$. Задачу решить с помощью интерполяционного многочлена Лагранжа.

1		2	
x	y	x	y
0	11	0	11
2	13	1	12
3	13	3	13
5	14	5	14
$x^* = 1$		$x^* = 2$	

1. Для решения дописал ранее написанную программу

Дописанный код

Глобальная область:

```
const int m4 = 6;
double x3[m4] = { 1, 1.2, 1.4, 1.6, 1.8, 2 };
double y3[m4] = { 0.8, 1.8, 2.9, 4.0, 4.9, 6.1 };
double y4[m4] = { 1.1, 2.2, 3.0, 4.1, 4.9, 5.9 };
double ИнтерполяцияНьютона(double* x, double* y, double* a, int k, double t);
```

main:

```
double ab4[m3];
cout << "Ответ: y(1.1) = " << ИнтерполяцияНьютона(x3, y3, ab4, m4, 1.1) << endl;
cout << "Ответ: y(1.1) = " << ИнтерполяцияНьютона(x3, y4, ab4, m4, 1.1) << endl;
```

Реализация ф-ий:

```
double Px(double* x, double* a, int k, double t)
{
    double S = 0;
    double P = 1;
    for (size_t i = 0; i < k; i++)
    {
        for (size_t j = 0; j < i; j++)
            P = P * (t - x[j]);
        S = S + a[i]*P;
        P = 1;
    }
    return S;
}

double ИнтерполяцияНьютона(double* x, double* y, double* a, int k, double t) {
    const int n = 5;
    double N[n];
    char** s;
    s = new char* [n];
    for (int i = 0; i < n; i++)
        s[i] = new char[15];

    char s0[15] = { "    i    " };
    char s1[15] = { "    x    " };
    char s2[15] = { "    y    " };
    char s3[15] = { "    P(xi)  " };
    char s4[15] = { "    a    " };
    s[0] = s0;
    s[1] = s1;
    s[2] = s2;
    s[3] = s3;
    s[4] = s4;
    N_tabl(n, s);
    N[0] = 0;
    N[1] = x[0];
    N[2] = y[0];
    N[3] = Px(x, a, 0, x[0]);
    N[4] = y[0];

    double S = 0;
    double P = 1;
    for (size_t i = 0; i < k; i++)
        a[i] = 0;

    for (size_t i = 0; i < k; i++)
    {
        C_tabl(n, N);

        for (size_t j = 0; j < i; j++)
```

```

        P = P * (x[i] - x[j]);
a[i] = (y[i]- Px(x, a, i, x[i])) / P;
P = 1;

N[0] = i;
N[1] = x[i];
N[2] = y[i];
N[3] = Px(x, a, i, x[i]);
N[4] = a[i];
}
K_tabl(n, N);
return Px(x, a, k, t);
}

```

Результат работы программы

i	x	y	P(xi)	a
0	1	0.8	0	0.8
0	1	0.8	0	0.8
1	1.2	1.8	0.8	5
2	1.4	2.9	2.8	1.25
3	1.6	4	4.1	-2.08333
4	1.8	4.9	5	-2.60417
5	2	6.1	5.3	20.8333

Ответ: $y(1.1) = 1.30703$

i	x	y	P(xi)	a
0	1	1.1	0	1.1
0	1	1.1	0	1.1
1	1.2	2.2	1.1	5.5
2	1.4	3	3.3	-3.75
3	1.6	4.1	3.5	12.5
4	1.8	4.9	6.1	-31.25
5	2	5.9	3.6	59.8958

Ответ: $y(1.1) = 1.83477$

2. Для решения дописал ранее написанную программу

Дописанный код

Глобальная область:

```
const int m5 = 4;
double x5[m5] = { 0, 2, 3, 5 };
double y5[m5] = { 11,13,13,14 };
double x6[m5] = { 0, 1, 3, 5 };
double y6[m5] = { 11,12,13,14 };
double ИнтерполяцияЛогранж(double* x, double* y, double* a, int k, double t);
```

main:

```
double ab5[m5];
cout << "Ответ: y(2.1) = " << ИнтерполяцияЛогранж(x5, y5, ab5, m5, 1) << endl;
cout << "Ответ: y(2.1) = " << ИнтерполяцияЛогранж(x6, y6, ab5, m5, 2) << endl;
```

Реализация ф-ий:

```
double ИнтерполяцияЛогранж(double* x, double* y, double* a, int k, double t) {
    double S = 0;
    double P = 1;
    for (size_t i = 0; i < k; i++)
    {
        for (size_t j = 0; j < k; j++)
            if (i != j)
                P *= (t - x[j]) / (x[i] - x[j]);
        cout << "L" << i << "(" << t << ") = " << P << " ";
        S += y[i] * P;
        P = 1;
    }
    cout << "\n";
    return S;
}
```

Результат работы программы

```
L0(1) = 0.266667  L1(1) = 1.33333  L2(1) = -0.666667  L3(1) = 0.0666667
Ответ: y(2.1) = 12.5333
L0(2) = -0.2  L1(2) = 0.75  L2(2) = 0.5  L3(2) = -0.05
Ответ: y(2.1) = 12.6
```

№ 1

Задание №5

Задача 1.

По заданным значениям x и y найти прямую $y = a_0 + a_1 x$ и параболу

$y = a_0 + a_1 x + a_2 x^2$ методом наименьших квадратов. Найти погрешность. Построить прямую и кривую в той же системе координат, где нанесены данные точки.

N	1	2	3	4	5	6	7
X	0.4	0.95	1.12	1.24	2.34	2.78	3.7
Y	0.96	2.23	2.38	2.98	4.77	6.07	7.77

Задача 2.

Найти 1-ю и 2-ю производную для функции заданной в виде таблицы в точке $X_* = 0,5$

X	-1	0	1	2
Y	0	1	2	3

1. Для решения дописал ранее написанную программу

Дописанный код

Глобальная область:

```
double x7[7] = { 0.4, 0.95, 1.12, 1.24, 2.34, 2.78, 3.7 };
double y7[7] = { 0.96, 2.23, 2.38, 2.98, 4.77, 6.07, 7.77 };
void НаимКвЛин(double* x, double* y, double* a, int k, bool t = 1);
```

main:

```
double ab6[3];
НаимКвЛин(x7, y7, ab6, 7);
cout << "Ответ: (a;b) = (" << ab6[1] << ";" << ab6[0] << ")" << endl;
НаимКвЛин(x7, y7, ab6, 7, 0);
cout << "Ответ: (a;b;c) = (" << ab6[2] << ";" << ab6[1] << ";" << ab6[0] << ")" <<
endl;
```

Реализация ф-ий:

```
void НаимКвЛин(double* x, double* y, double* a, int k, bool t) {
    double t1 = 0;
    double x = 0;
    double x2 = 0;
    double x3 = 0;
    double x4 = 0;
    double y = 0;
    double xy = 0;
    double x2y = 0;
    for (size_t i = 0; i < k; i++)
    {
        x += x[i];
        x2 += x[i] * x[i];
        x3 += x[i] * x[i] * x[i];
        x4 += x[i] * x[i] * x[i] * x[i];
        y += y[i];
        xy += y[i] * x[i];
        x2y += y[i] * x[i] * x[i];
    }
    if (t) {
        a[1] = (k * xy - x * y) / (k * x2 - x * x);
        a[0] = (y - a[1] * x) / k;
        for (size_t i = 0; i < k; i++)
        {
            t1 += pow(x[i] * a[1] + a[0] - y[i], 2);
            cout << x[i] << " " << y[i] << " " << x[i] * a[1] + a[0] << " " <<
            pow(x[i] * a[1] + a[0] - y[i], 2) << "\n";
        }
    }
    else {
        a[0] = (y * x2 * x4 + x * x3 * x2y + xy * x3 * x2 - x2 * x2 * x2y - x * xy
        * x4 - x3 * x3 * y) / (k * x2 * x4 + x * x3 * x2 + x * x3 * x2 - x2 * x2 * x2 - x * x *
        x4 - x3 * x3 * k);
        a[1] = (k * xy * x4 + y * x3 * x2 + x * x2y * x2 - x2 * xy * x2 - y * x *
        x4 - x3 * x2y * k) / (k * x2 * x4 + x * x3 * x2 + x * x3 * x2 - x2 * x2 * x2 - x * x * x4
        - x3 * x3 * k);
        a[2] = (k * x2 * x2y + x * xy * x2 + x * x3 * y - y * x2 * x2 - x * x * x2y
        - xy * x3 * k) / (k * x2 * x4 + x * x3 * x2 + x * x3 * x2 - x2 * x2 * x2 - x * x * x4 -
        x3 * x3 * k);
        for (size_t i = 0; i < k; i++)
        {
            t1 += pow(x[i] * x[i] * a[2] + x[i] * a[1] + a[0] - y[i], 2);
            cout << x[i] << " " << y[i] << " " << x[i] * x[i] * a[2] + x[i] *
            a[1] + a[0] << " " << pow(x[i] * x[i] * a[2] + x[i] * a[1] + a[0] - y[i], 2) << "\n";
        }
    }
    cout << "Погрешность = " << pow((t1/k), 0.5) << "\n";
}
```

}

Результат работы программы

```
0.4 0.96 1.03439 0.00553403
0.95 2.23 2.16035 0.00485092
1.12 2.38 2.50838 0.0164803
1.24 2.98 2.75404 0.0510581
2.34 4.77 5.00596 0.0556773
2.78 6.07 5.90673 0.0266575
3.7 7.77 7.79015 0.000406153
```

Погрешность = 0.151499

Ответ: (a;b) = (2.0472;0.215511)

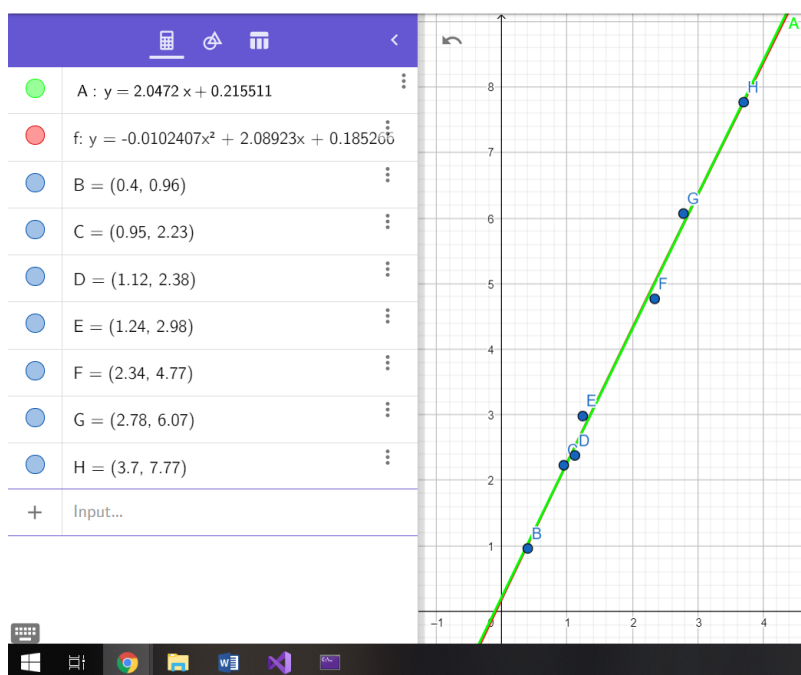
```
0.4 0.96 1.01932 0.00351868
0.95 2.23 2.16079 0.00479001
1.12 2.38 2.51236 0.0175179
1.24 2.98 2.76016 0.0483286
2.34 4.77 5.01799 0.0614966
2.78 6.07 5.91418 0.0242814
3.7 7.77 7.77521 2.71837e-05
```

Погрешность = 0.151167

Ответ: (a;b;c) = (-0.0102407;2.08923;0.185266)

Прямая и кривая в той же системе координат, где нанесены данные точки

GeoGebra Калькулятор Люкс



Задача 2.

Найти 1-ю и 2-ю производную для функции заданной в виде таблицы в точке $X_* = 0,5$

X	-1	0	1	2
Y	0	1	2	3

$$y'_k = \frac{y_{k+1} - y_{k-1}}{2h}$$

$$Y'(0.5) = \frac{1-0}{2*0.5} = 1$$

X	-0.5	0.5	1.5
Y'	1	1	1

$$Y''(0.5) = \frac{1-1}{2*1} = 0$$

Ответ $Y'(0.5) = 1$, $Y''(0.5) = 0$.