

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра КСУ**

**ОТЧЕТ**  
**по лабораторной работе №7**  
**по дисциплине «Программирование и основы алгоритмизации»**  
**Тема: “Разработка программы численного решения обыкновенных**  
**дифференциальных уравнений”**  
**Бригада №1**

Студент гр. 6493

Студентка гр.6493

Преподаватель

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Огурецкий Д.В.

Алексеева К.А.

Лукомская О.Ю

Санкт-Петербург

2017

Постановка формальной задачи: написать программу для решения обыкновенного дифференциального уравнения вида  $\frac{dy}{dx} = ax^3 + bx^2 + cx + d$  методом Эйлера и методом Рунге-Кутты 2-го порядка при нулевых начальных условиях  $y(x_0 = 0) = y_0 = 0$ . Шаг  $h \leq 0.1$ . Значения коэффициентов:

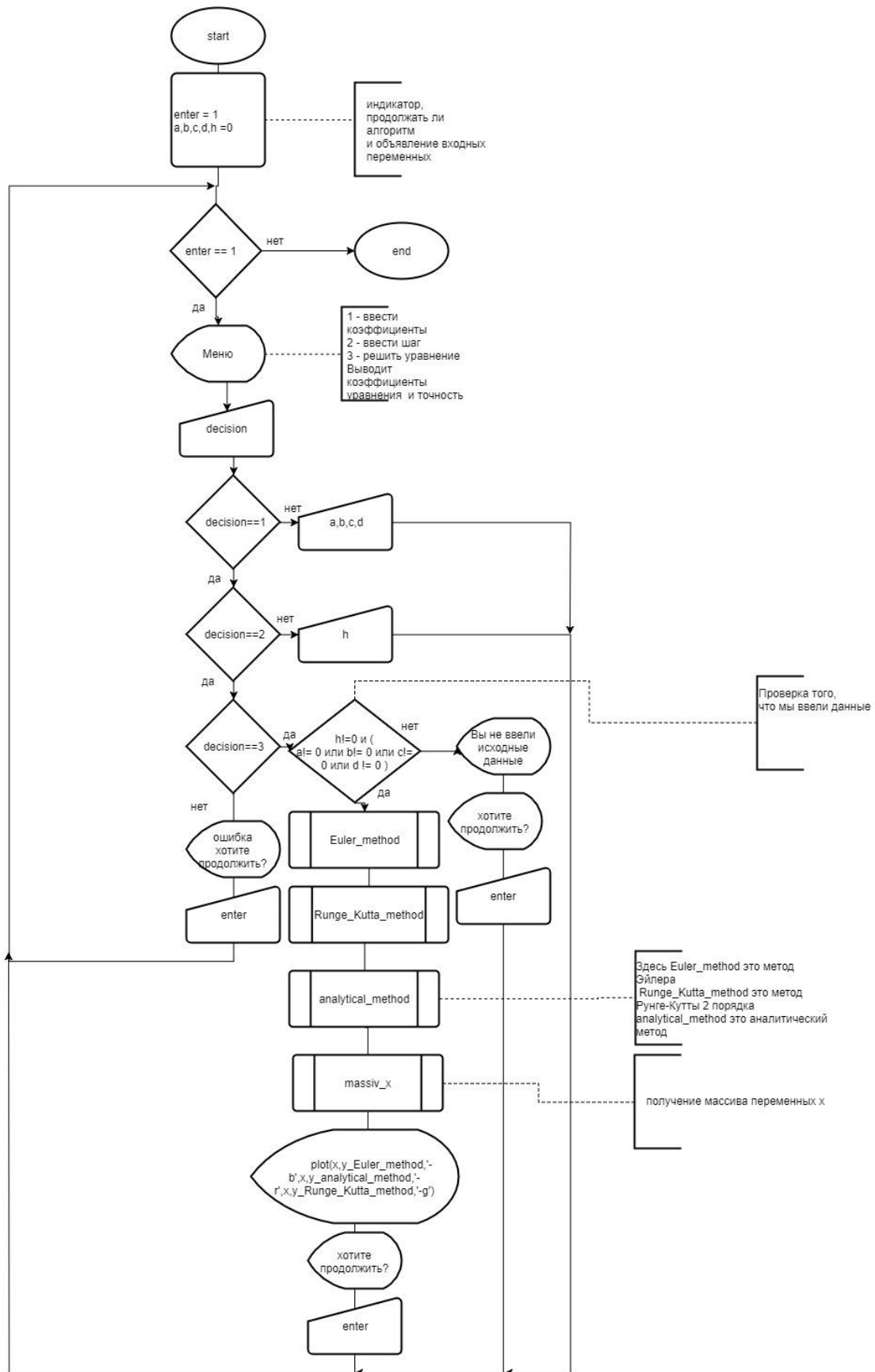
a	b	c	d
0.001	-0.125	4.90	-30

Замечание: коэффициенты вводятся с клавиатуры. Шаг интегрирования должен вводиться с клавиатуры. Результаты расчёта должны выводиться на экран или записываться в файл с заданным именем (по выбору пользователя). Методы решения должны быть оформлены в виде функций. Алгоритм должен предусматривать многократное изменение значений коэффициентов уравнения, шага интегрирования и последующего пересчёта результатов, а также возможность завершения работы по желанию пользователя. Полученная в результате интегрирования функция должна быть построена на интервале  $x=[0,30]$  в виде графика.

Цель : сравнить полученные функции при 1-ом и 2-ом методе с аналитически полученной функцией и её значениями.

Уловив смысл задания, мы решили сделать меню алгоритма, в котором можно менять коэффициенты и шаг интегрирования.

Блок-схема меню программы:



Код меню программы:

```
clear
enter=1;
a=0;
b=0;
c=0;
d=0;
h=0;
while enter==1
    clc
    a
    b
    c
    d
    h
    decision = menu('Желаемая операция?', 'ввести коэффициенты уравнения', 'ввести шаг интегрирования', 'решить уравнение');
    switch decision
        case 1
            a = input('Введите коэфф. a');
            b = input('Введите коэфф. b');
            c = input('Введите коэфф. c');
            d = input('Введите коэфф. d');
        case 2
            h = input('Введите шаг интегрирования');
        case 3
            if ((a~=0||b~=0||c~=0||d~=0)&&h~=0)
                y_Euler_method = Euler_method(a,b,c,d,h) ;
                y_Runge_Kutta_method = Runge_Kutta_method(a,b,c,d,h);
                y_analytical_method = analytical_method(a,b,c,d,h);
                x=massiv_x(h);
                plot(x,y_Euler_method, '-b', x,y_analytical_method, '-r', x,y_Runge_Kutta_method, '-g')
                legend('график метода Эйлера', 'график аналитического метода', 'график метода Рунге-Кутты 2 порядка')
                grid
                title('Графики функций')
                xlabel('x')
                ylabel('y(x)')
                enter = menu('хотите продолжить?', 'да', 'нет');
            else
                enter = menu('Вы не ввели исходные данные, хотите продолжить?', 'да', 'нет');
            end
        end
    end
end
```

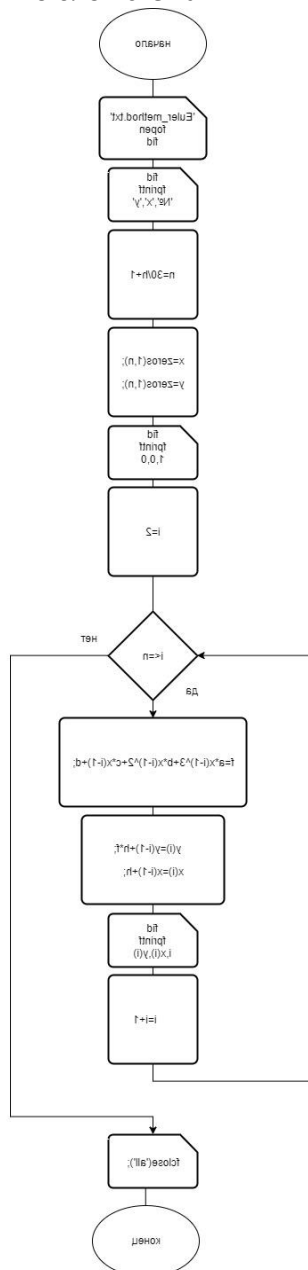
Код функции метода Эйлера , функции Euler\_method:

```
function y = Euler_method(a,b,c,d,h)
fid=fopen('Euler_method.txt','w');
fprintf(fid,'%6s %15s %20s\r\n','№','x','y');
n=30/h+1;%число итераций
x=zeros(1,n);%формирование массива
y=zeros(1,n);
fprintf(fid,'%6g %15g %20g\r\n',1,0,0);
i=2;%счётчик
while i<=n
    f=a*x(i-1)^3+b*x(i-1)^2+c*x(i-1)+d;
    y(i)=y(i-1)+h*f;
    x(i)=x(i-1)+h;
    fprintf(fid,'%6g %15g %20g\r\n',i,x(i),y(i));
    i=i+1;
end
```

```
fclose('all');
```

```
end
```

Его блок-схема:

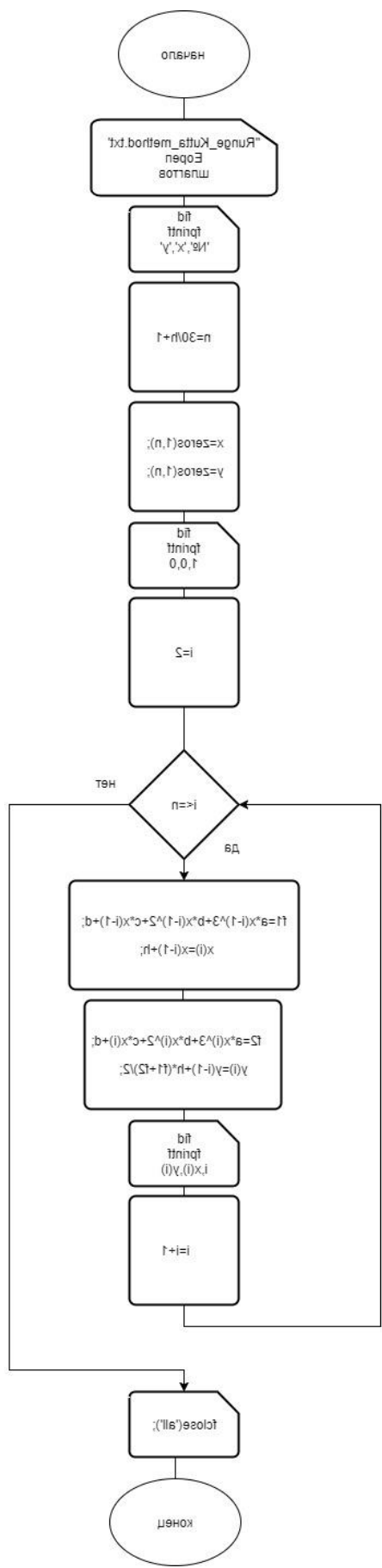


Метод Рунге-Кутты 2 порядка функция Runge\_Kutta\_method

```
function y = Runge_Kutta_method(a,b,c,d,h)
fid=fopen('Runge_Kutta_method.txt','w');
fprintf(fid,'%6s %15s %20s\r\n','№','x','y');
n=30/h+1;%число итераций
x=zeros(1,n);%формирование массива
y=zeros(1,n);
fprintf(fid,'%6g %15g %20g\r\n',1,0,0);
i=2;%счётчик
while i<=n
    f1=a*x(i-1)^3+b*x(i-1)^2+c*x(i-1)+d;
    x(i)=x(i-1)+h;
    f2=a*x(i)^3+b*x(i)^2+c*x(i)+d;
    y(i)=y(i-1)+h*(f1+f2)/2;
    fprintf(fid,'%6g %15g %20g\r\n',i,x(i),y(i));
    i=i+1;
end

fclose('all');
end
```

Блок-схема:



Метод аналитический Функция `analytical_method(a,b,c,d,h)`

Для нахождения точных значений функций и построения точного графика.

Решим аналитически дифф. уравнение и найдём его значения на интервале  $[0,30]$  с помощью программы в matlab:

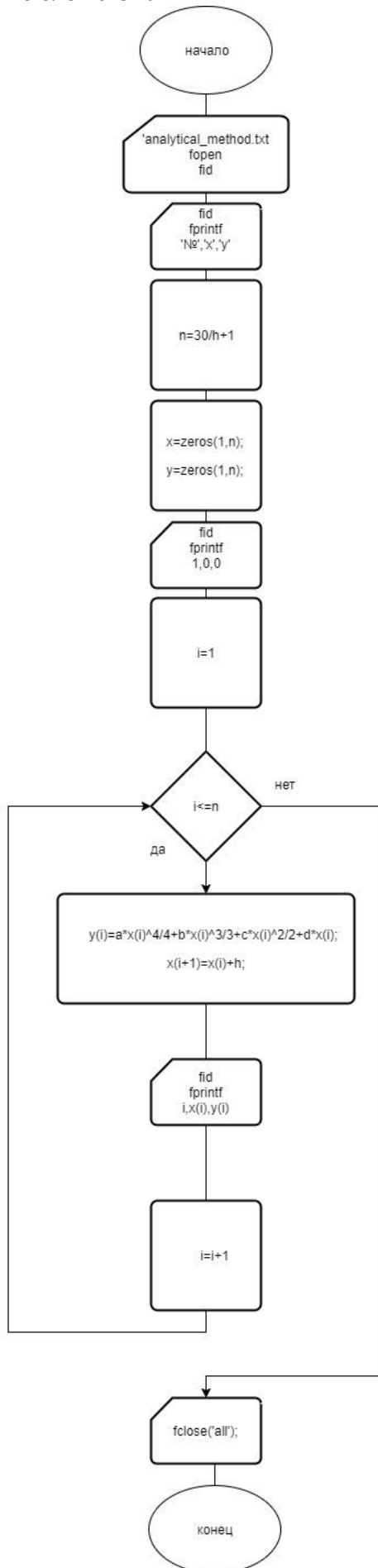
Используем полученное выражение в нашей функции `analytical_method`:

```
function y = analytical_method(a,b,c,d,h)
fid=fopen('analytical_method.txt','w');
fprintf(fid,'%6s %15s %20s\r\n','№','x','y');
n=30/h+1; %количество итераций
x=zeros(1,n);
y=zeros(1,n);
i=1; %счётчик
while i<=n
    y(i)=a*x(i)^4/4+b*x(i)^3/3+c*x(i)^2/2+d*x(i);
    x(i+1)=x(i)+h;
    fprintf(fid,'%6g %15g %20g\r\n',i,x(i),y(i));
    i=i+1;
end

fclose('all');
end
```



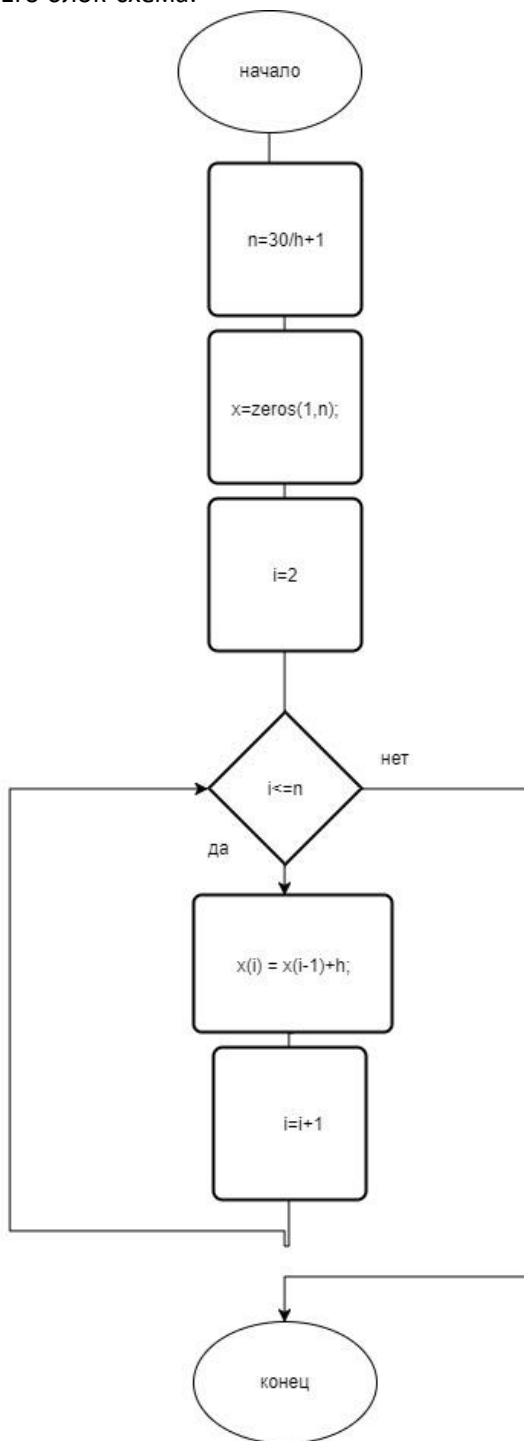
Его блок-схема:



Функция для создания массива значений  $x=[0,30]$  с шагом  $h$  функция `massiv_x`

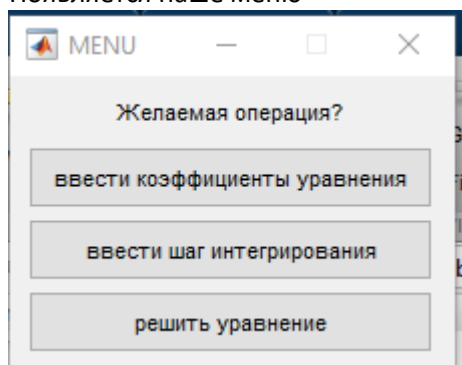
```
function x = massiv_x(h)
n=30/h+1;%количество элементов;
x=zeros(1,n);
i=2;
while i<=n
    x(i) = x(i-1)+h;
    i=i+1;
end
end
```

Его блок-схема:



Демонстрация работы:

Появляется наше меню



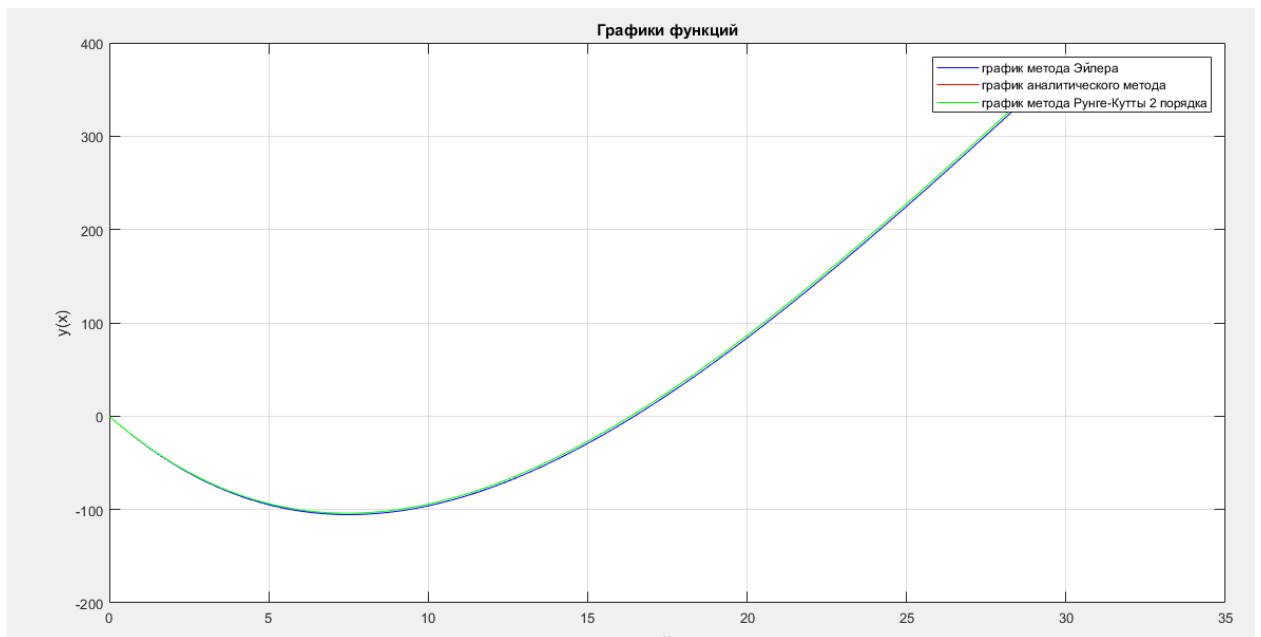
Мы выбираем нужный пункт и записываем исходные данные.

Видим, что в командной строке, отображаются введенные данные. (Это мы сделали для удобства )

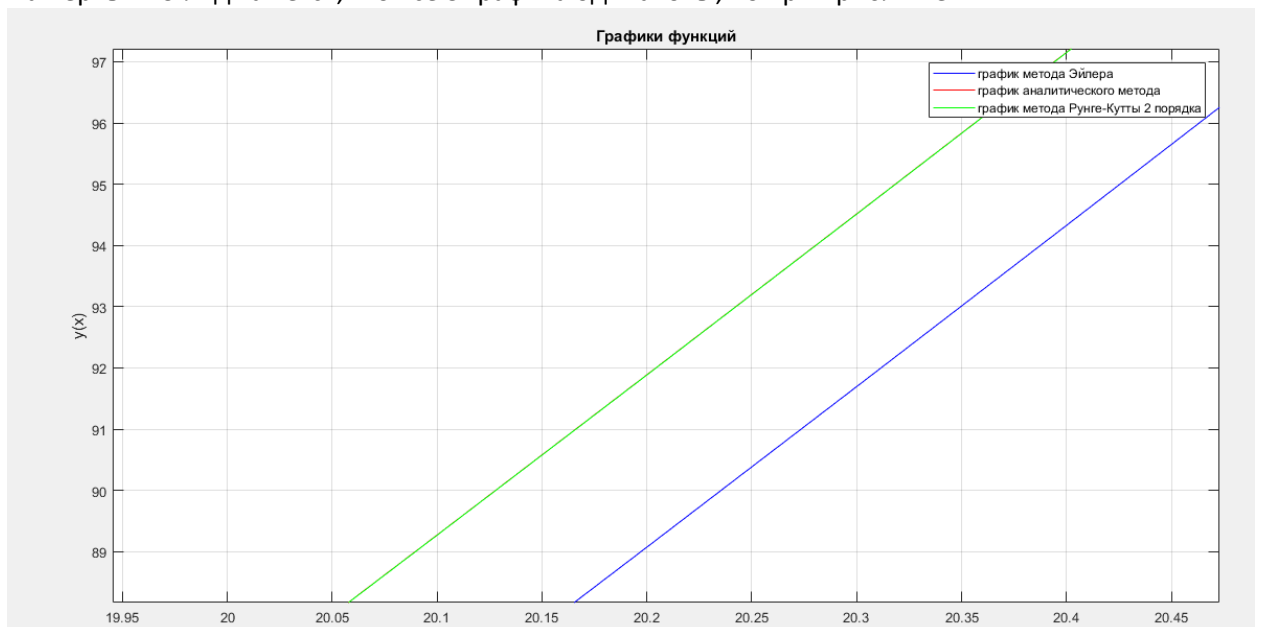
```
a =  
1.0000e-03  
  
b =  
-0.1250  
  
c =  
4.9000  
  
d =  
-30  
  
h =  
0.1000
```

Далее выбираем пункт решить уравнение:

Появились графики для двух методов + аналит. метод

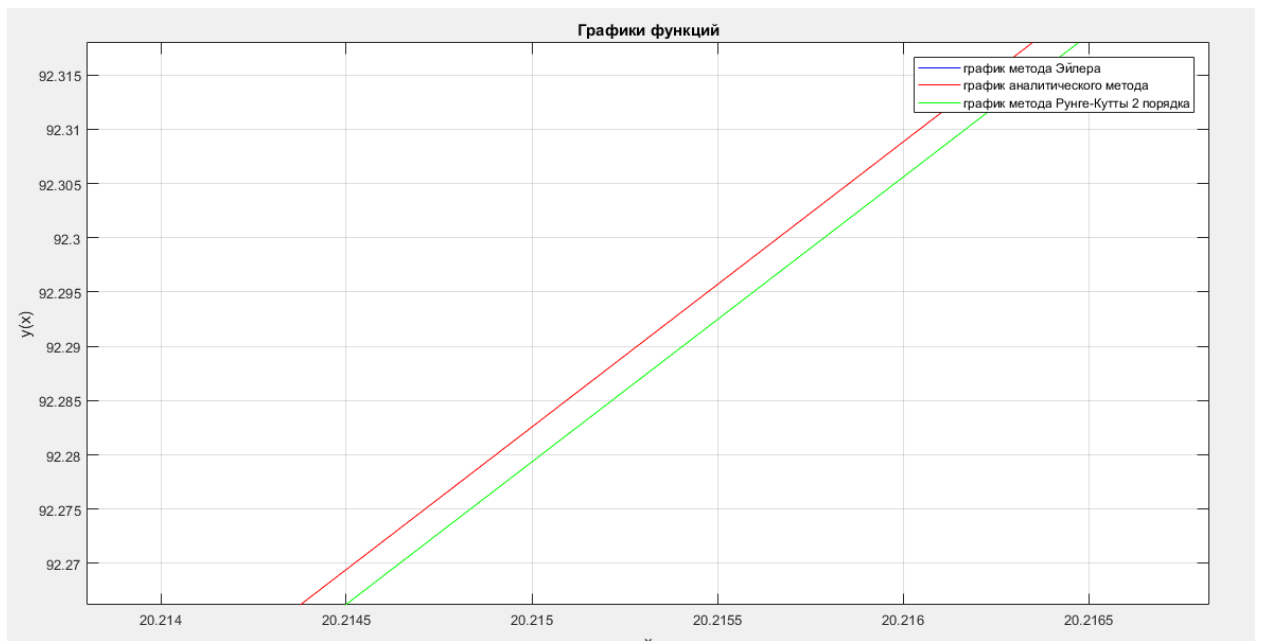


На первый взгляд кажется, что все 3 графика одинаковы, но при приближении



Мы видим, что синий график отличается от зеленого и красного, при этом красный и зеленый, кажутся совпадающими. Это и понятно, так как метод Эйлера, то есть синий график менее точный, чем метод Рунге-Кутты, то есть зеленый график.

При дальнейшем приближении мы видим, что и метод Рунге-Кутты немного отличается от аналитического графика, но это отличие значительно меньше по сравнению с методом Эйлера. У метода Эйлера отличие составляет порядка 3 единиц по оси  $y$ , а у метода Рунге-Кутты порядка 0.0025 по оси  $y$ .



Проконтролируем значения функции, записанные в текстовых файлах.

Euler\_method — Блокнот

Файл	Правка	Формат	Вид	Справка
N	x	y		
1	0	0		
2	0.1	-3		
3	0.2	-5.95112		
4	0.3	-8.85362		
5	0.4	-11.7077		
6	0.5	-14.5137		
7	0.6	-17.2719		
8	0.7	-19.9823		
9	0.8	-22.6454		
10	0.9	-25.2614		
11	1	-27.8304		
12	1.1	-30.3528		
13	1.2	-32.8288		
14	1.3	-35.2586		
15	1.4	-37.6425		
16	1.5	-39.9808		
17	1.6	-42.2736		
18	1.7	-44.5212		
19	1.8	-46.7238		
20	1.9	-48.8817		
21	2	-50.9951		
22	2.1	-53.0643		
23	2.2	-55.0895		
24	2.3	-57.071		
25	2.4	-59.0089		
26	2.5	-60.9035		
27	2.6	-62.7551		
28	2.7	-64.5638		
29	2.8	-66.33		
30	2.9	-68.0538		
31	3	-69.7355		
32	3.1	-71.3753		
33	3.2	-72.9734		
34	3.3	-74.5301		
35	3.4	-76.0457		
36	3.5	-77.5202		
37	3.6	-78.9541		
38	3.7	-80.3474		
39	3.8	-81.7005		
40	3.9	-83.0135		
41	4	-84.2867		
42	4.1	-85.5203		
43	4.2	-86.7145		
44	4.3	-87.8696		
45	4.4	-88.9858		
46	4.5	-90.0632		
47	4.6	-91.1023		
48	4.7	-92.103		
49	4.8	-93.0658		
50	4.9	-93.9907		
51	5	-94.8781		
52	5.1	-95.7281		
53	5.2	-96.5409		
54	5.3	-97.3169		
55	5.4	-98.0561		
56	5.5	-98.7589		
57	5.6	-99.4253		
58	5.7	-100.056		
59	5.8	-100.65		
60	5.9	-101.209		
61	6	-101.733		
62	6.1	-102.221		
63	6.2	-102.675		
64	6.3	-103.093		
65	6.4	-103.478		
66	6.5	-103.827		
67	6.6	-104.143		
68	6.7	-104.425		
69	6.8	-104.673		
70	6.9	-104.887		
71	7	-105.069		
72	7.1	-105.217		
73	7.2	-105.332		
74	7.3	-105.415		
75	7.4	-105.465		
76	7.5	-105.483		
77	7.6	-105.469		
78	7.7	-105.423		
79	7.8	-105.346		
80	7.9	-105.237		
81	8	-105.096		
82	8.1	-104.925		
83	8.2	-104.723		
84	8.3	-104.491		
85	8.4	-104.228		
86	8.5	-103.934		
87	8.6	-103.611		
88	8.7	-103.258		
89	8.8	-102.875		
90	8.9	-102.463		
91	9	-102.022		
92	9.1	-101.551		
93	9.2	-101.052		
94	9.3	-100.524		
95	9.4	-99.9678		
96	9.5	-99.3833		
97	9.6	-98.7706		
98	9.7	-98.1302		
99	9.8	-97.462		
100	9.9	-96.7664		
101	10	-96.0435		
102	10.1	-95.2935		
103	10.2	-94.5166		
104	10.3	-93.713		
105	10.4	-92.8828		
106	10.5	-92.0263		
107	10.6	-91.1437		

analytical\_method — Блокнот

Файл	Правка	Формат	Вид	Справка
N	x	y		
1	0	0		
2	0.1	-2.97554		
3	0.2	-5.90233		
4	0.3	-8.78062		
5	0.4	-11.6107		
6	0.5	-14.3927		
7	0.6	-17.127		
8	0.7	-19.8137		
9	0.8	-22.4532		
10	0.9	-25.0457		
11	1	-27.5914		
12	1.1	-30.0906		
13	1.2	-32.5435		
14	1.3	-34.9503		
15	1.4	-37.3114		
16	1.5	-39.6269		
17	1.6	-41.897		
18	1.7	-44.1221		
19	1.8	-46.3024		
20	1.9	-48.438		
21	2	-50.5293		
22	2.1	-52.5765		
23	2.2	-54.5798		
24	2.3	-56.5395		
25	2.4	-58.4557		
26	2.5	-60.3288		
27	2.6	-62.1589		
28	2.7	-63.9463		
29	2.8	-65.6913		
30	2.9	-67.394		
31	3	-69.0548		
32	3.1	-70.6737		
33	3.2	-72.2511		
34	3.3	-73.7872		
35	3.4	-75.2823		
36	3.5	-76.7364		
37	3.6	-78.15		
38	3.7	-79.5232		
39	3.8	-80.8562		
40	3.9	-82.1493		
41	4	-83.4027		
42	4.1	-84.6166		
43	4.2	-85.7912		
44	4.3	-86.9268		
45	4.4	-88.0236		
46	4.5	-89.0819		
47	4.6	-90.1017		
48	4.7	-91.0835		
49	4.8	-92.0273		
50	4.9	-92.9334		
51	5	-93.8021		
52	5.1	-94.6335		
53	5.2	-95.4279		
54	5.3	-96.1854		
55	5.4	-96.9064		
56	5.5	-97.591		
57	5.6	-98.2395		
58	5.7	-98.852		
59	5.8	-99.4288		
60	5.9	-99.97		
61	6	-100.476		
62	6.1	-100.947		
63	6.2	-101.383		
64	6.3	-101.784		
65	6.4	-102.151		
66	6.5	-102.484		
67	6.6	-102.783		
68	6.7	-103.048		
69	6.8	-103.279		
70	6.9	-103.477		
71	7	-103.641		
72	7.1	-103.773		
73	7.2	-103.872		
74	7.3	-103.939		
75	7.4	-103.973		
76	7.5	-103.975		
77	7.6	-103.945		
78	7.7	-103.883		
79	7.8	-103.79		
80	7.9	-103.665		
81	8	-103.509		
82	8.1	-103.323		
83	8.2	-103.105		
84	8.3	-102.858		
85	8.4	-102.579		
86	8.5	-102.271		
87	8.6	-101.933		
88	8.7	-101.565		
89	8.8	-101.167		
90	8.9	-100.741		
91	9	-100.285		
92	9.1	-99.7999		
93	9.2	-99.2864		
94	9.3	-98.7442		
95	9.4	-98.1738		
96	9.5	-97.5752		
97	9.6	-96.9486		
98	9.7	-96.2943		
99	9.8	-95.6124		
100	9.9	-94.9031		
101	10	-94.1667		
102	10.1	-93.4032		
103	10.2	-92.6129		
104	10.3	-91.796		
105	10.4	-90.9527		
106	10.5	-90.0831		
107	10.6	-89.1875		

Runge\_Kutta\_method — Блокнот


Файл	Правка	Формат	Вид	Справка
N	x	y		
1	0	0		
2	0.1	-2.97556		
3	0.2	-5.90237		
4	0.3	-8.78069		
5	0.4	-11.6107		
6	0.5	-14.3928		
7	0.6	-17.1271		
8	0.7	-19.8139		
9	0.8	-22.4534		
10	0.9	-25.0459		
11	1	-27.5916		
12	1.1	-30.0908		
13	1.2	-32.5437		
14	1.3	-34.9506		
15	1.4	-37.3117		
16	1.5	-39.6272		
17	1.6	-41.8974		
18	1.7	-44.1225		
19	1.8	-46.3027		
20	1.9	-48.4384		
21	2	-50.5297		
22	2.1	-52.5769		
23	2.2	-54.5803		
24	2.3	-56.5399		
25	2.4	-58.4562		
26	2.5	-60.3293		
27	2.6	-62.1594		
28	2.7	-63.9469		
29	2.8	-65.6919		
30	2.9	-67.3946		
31	3	-69.0554		
32	3.1	-70.6743		
33	3.2	-72.2518		
34	3.3	-73.7879		
35	3.4	-75.2829		
36	3.5	-76.7371		
37	3.6	-78.1507		
38	3.7	-79.5239		
39	3.8	-80.857		
40	3.9	-82.1501		
41	4	-83.4035		
42	4.1	-84.6174		
43	4.2	-85.792		
44	4.3	-86.9277		
45	4.4	-88.0245		
46	4.5	-89.0827		
47	4.6	-90.1026		
48	4.7	-91.0844		
49	4.8	-		

108	10.7	-90.2351	108	10.7	-88.266	108	10.7	-88.2679
109	10.8	-89.3007	109	10.8	-87.3188	109	10.8	-87.3207
110	10.9	-88.3408	110	10.9	-86.3461	110	10.9	-86.3481
111	11	-87.3554	111	11	-85.3481	111	11	-85.3501
112	11.1	-86.3448	112	11.1	-84.3249	112	11.1	-84.327
113	11.2	-85.3091	113	11.2	-83.2769	113	11.2	-83.2789
114	11.3	-84.2486	114	11.3	-82.204	114	11.3	-82.2061
115	11.4	-83.1635	115	11.4	-81.1066	115	11.4	-81.1086
116	11.5	-82.0538	116	11.5	-79.9848	116	11.5	-79.9868
117	11.6	-80.9199	117	11.6	-78.8387	117	11.6	-78.8408
118	11.7	-79.7618	118	11.7	-77.6687	118	11.7	-77.6708
119	11.8	-78.5797	119	11.8	-76.4747	119	11.8	-76.4768
120	11.9	-77.3739	120	11.9	-75.2571	120	11.9	-75.2592
121	12	-76.1445	121	12	-74.016	121	12	-74.0181
122	12.1	-74.8917	122	12.1	-72.7516	122	12.1	-72.7537
123	12.2	-73.6157	123	12.2	-71.464	123	12.2	-71.4662
124	12.3	-72.3166	124	12.3	-70.1535	124	12.3	-70.1556
125	12.4	-70.9947	125	12.4	-68.8201	125	12.4	-68.8223
126	12.5	-69.65	126	12.5	-67.4642	126	12.5	-67.4664
127	12.6	-68.2828	127	12.6	-66.0858	127	12.6	-66.088
128	12.7	-66.8933	128	12.7	-64.6852	128	12.7	-64.6874
129	12.8	-65.4816	129	12.8	-63.2624	129	12.8	-63.2647
130	12.9	-64.0478	130	12.9	-61.8178	130	12.9	-61.8201
131	13	-62.5923	131	13	-60.3514	131	13	-60.3537
132	13.1	-61.1151	132	13.1	-58.8635	132	13.1	-58.8658
133	13.2	-59.6164	133	13.2	-57.3541	133	13.2	-57.3564
134	13.3	-58.0964	134	13.3	-55.8235	134	13.3	-55.8259
135	13.4	-56.5553	135	13.4	-54.2719	135	13.4	-54.2742
136	13.5	-54.9932	136	13.5	-52.6994	136	13.5	-52.7017
137	13.6	-53.4103	137	13.6	-51.1061	137	13.6	-51.1085
138	13.7	-51.8067	138	13.7	-49.4923	138	13.7	-49.4947
139	13.8	-50.1827	139	13.8	-47.8582	139	13.8	-47.8606
140	13.9	-48.5384	140	13.9	-46.2038	140	13.9	-46.2062
141	14	-46.874	141	14	-44.5293	141	14	-44.5318
142	14.1	-45.1896	142	14.1	-42.835	142	14.1	-42.8375
143	14.2	-43.4854	143	14.2	-41.121	143	14.2	-41.1234
144	14.3	-41.7615	144	14.3	-39.3874	144	14.3	-39.3899
145	14.4	-40.0182	145	14.4	-37.6345	145	14.4	-37.6369
146	14.5	-38.2556	146	14.5	-35.8623	146	14.5	-35.8648
147	14.6	-36.4739	147	14.6	-34.071	147	14.6	-34.0735
148	14.7	-34.6732	148	14.7	-32.2609	148	14.7	-32.2634
149	14.8	-32.8537	149	14.8	-30.432	149	14.8	-30.4346
150	14.9	-31.0155	150	14.9	-28.5846	150	14.9	-28.5871
151	15	-29.1588	151	15	-26.7188	151	15	-26.7213
152	15.1	-27.2838	152	15.1	-24.8347	152	15.1	-24.8372
153	15.2	-25.3906	153	15.2	-22.9325	153	15.2	-22.9351
154	15.3	-23.4795	154	15.3	-21.0123	154	15.3	-21.0149
155	15.4	-21.5504	155	15.4	-19.0745	155	15.4	-19.0771
156	15.5	-19.6037	156	15.5	-17.1189	156	15.5	-17.1216
157	15.6	-17.6394	157	15.6	-15.146	157	15.6	-15.1486
158	15.7	-15.6578	158	15.7	-13.1557	158	15.7	-13.1584
159	15.8	-13.6589	159	15.8	-11.1483	159	15.8	-11.151
160	15.9	-11.643	160	15.9	-9.1239	160	15.9	-9.12658
161	16	-9.61016	161	16	-7.08267	161	16	-7.08536
162	16.1	-7.56056	162	16.1	-5.02475	162	16.1	-5.02746
163	16.2	-5.49436	163	16.2	-2.95031	163	16.2	-2.95303
164	16.3	-3.4117	164	16.3	-0.859498	164	16.3	-0.862229
165	16.4	-1.31275	165	16.4	1.24754	165	16.4	1.24479
166	16.5	0.80234	166	16.5	3.37064	166	16.5	3.36788
167	16.6	2.93343	167	16.6	5.50966	167	16.6	5.50689
168	16.7	5.08036	168	16.7	7.66445	168	16.7	7.66167
169	16.8	7.24298	169	16.8	9.83485	169	16.8	9.83206
170	16.9	9.42114	170	16.9	12.0207	170	16.9	12.0179
171	17	11.6147	171	17	14.2219	171	17	14.2191
172	17.1	13.8235	172	17.1	16.4383	172	17.1	16.4354
173	17.2	16.0474	173	17.2	18.6697	173	17.2	18.6668
174	17.3	18.2862	174	17.3	20.9159	174	17.3	20.9131
175	17.4	20.5399	175	17.4	23.1769	175	17.4	23.174
176	17.5	22.8082	176	17.5	25.4525	176	17.5	25.4496
177	17.6	25.091	177	17.6	27.7425	177	17.6	27.7396
178	17.7	27.3882	178	17.7	30.0468	178	17.7	30.0439
179	17.8	29.6996	179	17.8	32.3652	179	17.8	32.3623
180	17.9	32.0251	180	17.9	34.6977	180	17.9	34.6948
181	18	34.3645	181	18	37.044	181	18	37.0411
182	18.1	36.7177	182	18.1	39.404	182	18.1	39.4011
183	18.2	39.0845	183	18.2	41.7777	183	18.2	41.7747
184	18.3	41.4649	184	18.3	44.1647	184	18.3	44.1617
185	18.4	43.8586	185	18.4	46.5651	185	18.4	46.5621
186	18.5	46.2655	186	18.5	48.9786	186	18.5	48.9756
187	18.6	48.6856	187	18.6	51.4051	187	18.6	51.4021
188	18.7	51.1186	188	18.7	53.8445	188	18.7	53.8415
189	18.8	53.5644	189	18.8	56.2966	189	18.8	56.2936
190	18.9	56.0228	190	18.9	58.7614	190	18.9	58.7583
191	19	58.4938	191	19	61.2386	191	19	61.2355
192	19.1	60.9772	192	19.1	63.7281	192	19.1	63.7251
193	19.2	63.4729	193	19.2	66.2299	193	19.2	66.2268
194	19.3	65.9807	194	19.3	68.7437	194	19.3	68.7406
195	19.4	68.5005	195	19.4	71.2694	195	19.4	71.2663
196	19.5	71.0321	196	19.5	73.8069	196	19.5	73.8038
197	19.6	73.5755	197	19.6	76.3561	197	19.6	76.3529
198	19.7	76.1304	198	19.7	78.9168	198	19.7	78.9136
199	19.8	78.6968	199	19.8	81.4888	199	19.8	81.4857
200	19.9	81.2746	200	19.9	84.0722	200	19.9	84.069
201	20	83.8635	201	20	86.6667	201	20	86.6635
202	20.1	86.4635	202	20.1	89.2721	202	20.1	89.269
203	20.2	89.0744	203	20.2	91.8885	203	20.2	91.8853
204	20.3	91.6962	204	20.3	94.5156	204	20.3	94.5124
205	20.4	94.3286	205	20.4	97.1533	205	20.4	97.1501
206	20.5	96.9716	206	20.5	99.8015	206	20.5	99.7983
207	20.6	99.6249	207	20.6	102.46	207	20.6	102.457
208	20.7	102.289	208	20.7	105.129	208	20.7	105.126
209	20.8	104.962	209	20.8	107.808	209	20.8	107.804
210	20.9	107.646	210	20.9	110.497	210	20.9	110.493
211	21	110.34	211	21	113.195	211	21	113.192
212	21.1	113.044	212	21.1	115.904	212	21.1	115.9
213	21.2	115.757	213	21.2	118.622	213	21.2	118.618
214	21.3	118.48	214	21.3	121.349	214	21.3	121.346
215	21.4	121.212	215	21.4	124.086	215	21.4	124.083

216	21.5	123.954	216	21.5	126.832	216	21.5	126.829
217	21.6	126.704	217	21.6	129.588	217	21.6	129.584
218	21.7	129.464	218	21.7	132.352	218	21.7	132.348
219	21.8	132.233	219	21.8	135.125	219	21.8	135.122
220	21.9	135.01	220	21.9	137.907	220	21.9	137.903
221	22	137.797	221	22	140.697	221	22	140.694
222	22.1	140.591	222	22.1	143.496	222	22.1	143.493
223	22.2	143.395	223	22.2	146.304	223	22.2	146.3
224	22.3	146.206	224	22.3	149.12	224	22.3	149.116
225	22.4	149.026	225	22.4	151.943	225	22.4	151.94
226	22.5	151.854	226	22.5	154.775	226	22.5	154.772
227	22.6	154.69	227	22.6	157.615	227	22.6	157.612
228	22.7	157.534	228	22.7	160.463	228	22.7	160.46
229	22.8	160.385	229	22.8	163.318	229	22.8	163.315
230	22.9	163.245	230	22.9	166.181	230	22.9	166.178
231	23	166.111	231	23	169.052	231	23	169.048
232	23.1	168.986	232	23.1	171.93	232	23.1	171.926
233	23.2	171.867	233	23.2	174.815	233	23.2	174.811
234	23.3	174.756	234	23.3	177.707	234	23.3	177.704
235	23.4	177.652	235	23.4	180.606	235	23.4	180.603
236	23.5	180.554	236	23.5	183.513	236	23.5	183.509
237	23.6	183.464	237	23.6	186.426	237	23.6	186.422
238	23.7	186.38	238	23.7	189.346	238	23.7	189.342
239	23.8	189.304	239	23.8	192.272	239	23.8	192.268
240	23.9	192.233	240	23.9	195.205	240	23.9	195.201
241	24	195.169	241	24	198.144	241	24	198.14
242	24.1	198.112	242	24.1	201.09	242	24.1	201.086
243	24.2	201.06	243	24.2	204.041	243	24.2	204.038
244	24.3	204.015	244	24.3	206.999	244	24.3	206.995
245	24.4	206.976	245	24.4	209.963	245	24.4	209.959
246	24.5	209.942	246	24.5	212.932	246	24.5	212.929
247	24.6	212.915	247	24.6	215.908	247	24.6	215.904
248	24.7	215.893	248	24.7	218.889	248	24.7	218.885
249	24.8	218.877	249	24.8	221.875	249	24.8	221.872
250	24.9	221.866	250	24.9	224.867	250	24.9	224.864
251	25	224.861	251	25	227.865	251	25	227.861
252	25.1	227.861	252	25.1	230.867	252	25.1	230.864
253	25.2	230.866	253	25.2	233.875	253	25.2	233.871
254	25.3	233.876	254	25.3	236.888	254	25.3	236.884
255	25.4	236.892	255	25.4	239.906	255	25.4	239.902
256	25.5	239.912	256	25.5	242.928	256	25.5	242.924
257	25.6	242.937	257	25.6	245.956	257	25.6	245.952
258	25.7	245.967	258	25.7	248.988	258	25.7	248.984
259	25.8	249.001	259	25.8	252.024	259	25.8	252.02
260	25.9	252.04	260	25.9	255.065	260	25.9	255.062
261	26	255.083	261	26	258.111	261	26	258.107
262	26.1	258.131	262	26.1	261.16	262	26.1	261.157
263	26.2	261.183	263	26.2	264.214	263	26.2	264.211
264	26.3	264.239	264	26.3	267.272	264	26.3	267.269
265	26.4	267.299	265	26.4	270.334	265	26.4	270.331
266	26.5	270.363	266	26.5	273.4	266	26.5	273.396
267	26.6	273.43	267	26.6	276.47	267	26.6	276.466
268	26.7	276.502	268	26.7	279.543	268	26.7	279.54
269	26.8	279.577	269	26.8	282.621	269	26.8	282.617
270	26.9	282.656	270	26.9	285.701	270	26.9	285.697
271	27	285.739	271	27	288.785	271	27	288.781
272	27.1	288.824	272	27.1	291.873	272	27.1	291.869
273	27.2	291.913	273	27.2	294.963	273	27.2	294.96
274	27.3	295.006	274	27.3	298.057	274	27.3	298.054
275	27.4	298.101	275	27.4	301.154	275	27.4	301.151
276	27.5	301.2	276	27.5	304.255	276	27.5	304.251
277	27.6	304.301	277	27.6	307.358	277	27.6	307.354
278	27.7	307.406	278	27.7	310.463	278	27.7	310.46
279	27.8	310.513	279	27.8	313.572	279	27.8	313.568
280	27.9	313.623	280	27.9	316.683	280	27.9	316.68
281	28	316.736	281	28	319.797	281	28	319.793
282	28.1	319.851	282	28.1	322.914	282	28.1	322.91
283	28.2	322.969	283	28.2	326.033	283	28.2	326.029
284	28.3	326.089	284	28.3	329.154	284	28.3	329.15
285	28.4	329.211	285	28.4	332.277	285	28.4	332.274
286	28.5	332.336	286	28.5	335.403	286	28.5	335.399
287	28.6	335.463	287	28.6	338.531	287	28.6	338.527
288	28.7	338.531	288	28.7	341.661	288	28.7	341.657
289	28.8	341.722	289	28.8	344.793	289	28.8	344.789
290	28.9	344.855	290	28.9	347.926	290	28.9	347.922
291	29	347.99	291	29	351.062	291	29	351.058
292	29.1	351.126	292	29.1	354.199	292	29.1	354.195
293	29.2	354.264	293	29.2	357.338	293	29.2	357.334
294	29.3	357.404	294	29.3	360.479	294	29.3	360.475
295	29.4	360.545	295	29.4	363.621	295	29.4	363.617
296	29.5	363.688	296	29.5	366.764	296	29.5	366.76
297	29.6	366.832	297	29.6	369.909	297	29.6	369.905
298	29.7	369.977	298	29.7	373.055	298	29.7	373.051
299	29.8	373.124	299	29.8	376.202	299	29.8	376.198
300	29.9	376.272	300	29.9	379.351	300	29.9	379.347
301	30	379.421	301	30	382.5	301	30	382.496

Как видим, метод Рунге-Кутты намного точнее метода Эйлера, и в достаточной точности описывает нашу функцию.

У нас появляется меню с выбором желания пользователя продолжить.


☐
☒

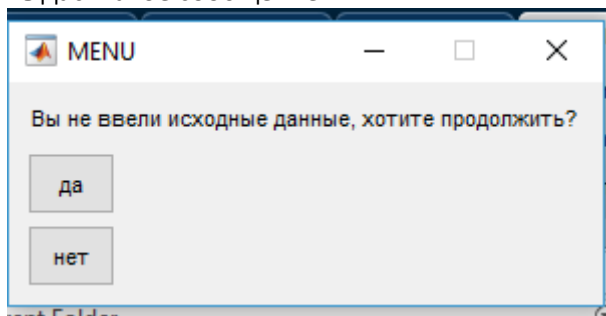
хотите продолжить?

да

нет



Также в программе мы сделали стрессоустойчивость, то есть для избежания ошибки, если мы пытаемся решить наше уравнение не введя достаточное количество исходных данных программа выдаст такое сообщение:



Вывод: Мы сделали программу , решающую дифференциальные уравнения типа  $\frac{dy}{dx} = ax^3 + bx^2 + cx + d$  и сделали вывод, что метод Рунге-Кутты значительно точнее метода Эйлера.