

Министерство высшего образования и науки Российской Федерации  
Национальный научно-исследовательский университет ИТМО  
Факультет программной инженерии и компьютерной техники

Лабораторная работа №6  
по дисциплине  
**«Вычислительная математика»**

Вариант 11

Работу выполнил:  
Макеев Роман Ильич

Группа Р3208

Преподаватель:  
Машина Екатерина Алексеевна

Санкт-Петербург

2024

## Цель работы:

Решить задачу Коши для обыкновенных дифференциальных уравнений численными методами.

## Листинг программы:

### Модифицированный Эйлер

```
def modified_euler_by_x_list(diff_ur: DiffUrDto, y_0: float, x_list:
list[float]) -> list[float]:
    h: float = x_list[1] - x_list[0]
    y_list: list[float] = [y_0]
    for i in range(len(x_list)-1):
        y_i = y_list[i]
        x_i = x_list[i]
        f_i = diff_ur.f_x_y(x_i, y_i)
        y_list.append(y_i + (h / 2) * (f_i + diff_ur.f_x_y(x_list[i+1], y_i +
h * f_i)))

    return y_list
```

### Рунге-Кутт 4-го порядка

```
def runge_kutt4_by_x_list(diff_ur: DiffUrDto, y_0: float, x_list:
list[float]) -> list[float]:
    h: float = x_list[1] - x_list[0]
    y_list: list[float] = [y_0]
    for i in range(len(x_list)-1):
        y_i = y_list[i]
        x_i = x_list[i]
        k1 = h * diff_ur.f_x_y(x_i, y_i)
        k2 = h * diff_ur.f_x_y(x_i + h/2, y_i + k1/2)
        k3 = h * diff_ur.f_x_y(x_i + h/2, y_i + k2/2)
        k4 = h * diff_ur.f_x_y(x_i + h, y_i + k3)
        y_list.append(y_i + (k1 + 2*k2 + 2*k3 + k4) / 6)

    return y_list
```

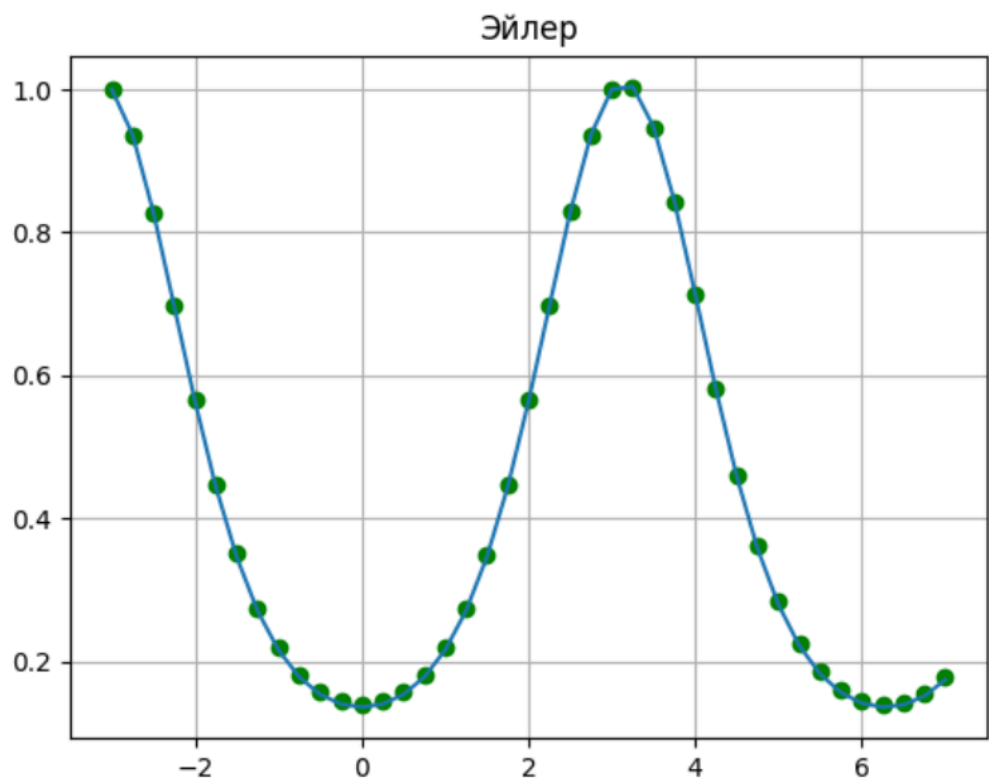
### Адамс

```
def adams_by_x_list(diff_ur: DiffUrDto, y_0: float, x_list: list[float]) ->
list[float]:
    h: float = x_list[1] - x_list[0]
    y_list: list[float] = runge_kutt4_by_x_list(diff_ur, y_0, x_list[:4])
    f_list: list[float] = []
    for i in range(len(y_list) - 1):
        f_list.append(diff_ur.f_x_y(x_list[i], y_list[i]))

    for i in range(len(y_list) - 1, len(x_list) - 1):
        f_list.append(diff_ur.f_x_y(x_list[i], y_list[i]))
        delta_f = f_list[i] - f_list[i - 1]
        delta2_f = f_list[i] - 2 * f_list[i - 1] + f_list[i - 2]
        delta3_f = f_list[i] - 3 * f_list[i - 1] + 3 * f_list[i - 2] -
f_list[i - 3]
        y_list.append(y_list[i] + h * f_list[i] + (h ** 2) * delta_f / 2
+ 5 * (h ** 3) * delta2_f / 12 + 3 * (h ** 4) * delta3_f
/ 8)

    return y_list
```

# Результаты работы программы:



Решение методом Эйлер  
Погрешность: 1.120E-03  
Количество точек: 41  
Шаг: 2.439E-01

i	x	y	Точный y
0	-3.0	1.0	1.0
1	-2.75	0.93634	0.93642
2	-2.5	0.8283	0.82791
3	-2.25	0.69783	0.69641
4	-2.0	0.56607	0.56335
5	-1.75	0.44793	0.44408
6	-1.5	0.35073	0.3462
7	-1.25	0.27577	0.27109
8	-1.0	0.22093	0.21647
9	-0.75	0.18283	0.17877
10	-0.5	0.15816	0.1545
11	-0.25	0.14438	0.14101
12	0.0	0.13991	0.1367
13	0.25	0.14424	0.14101
14	0.5	0.15788	0.1545
15	0.75	0.1824	0.17877
16	1.0	0.2204	0.21647
17	1.25	0.27523	0.27109
18	1.5	0.35033	0.3462
19	1.75	0.44785	0.44408
20	2.0	0.56636	0.56335

**Выводы:**

В ходе работы была написана программа, решающая задачу Коши численными методами Эйлера, Рунге-Кутта и Адамса