Министерство образования Республики Беларусь
Учреждение образования «Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерных систем и сетей
Кафедра информатики
Дисциплина «Методы численного анализа»

ОТЧЕТ

к лабораторной работе №10

на тему:

«МЕТОД АДАМСА»

БГУИР 1-40 04 01

Выполнил студент группы 253505
БЕКАРЕВ Станислав Сергеевич

(дата, подпись студента)

Проверил доцент кафедры информатики
АНИСИМОВ Владимир Яковлевич

(дата, подпись преподавателя)

Минск 2023

Содержание

Цель работы

- изучить численное решение задачи Коши для обыкновенных дифференциальных уравнений методом Адамса;
- проверить метод по трудоемкости, точности;
- составить программу реализации метода;
- выполнить тестовые примеры и проверить правильность работы программы

ЗАДАНИЕ. Найти с точностью до 0.001 решения следующих уравнений на отрезке [0; 1]

$$y' = \frac{a\left(1 - y^2\right)}{(1+m)x^2 + y^2 + 1}, \quad y(0) = 0,$$

где значения параметров a и m принимают следующие значения для вариантов k.

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
m	1.0	1.5	2.0	1.0	1.5	2.0	1.0	1.5	2.0	1.0	1.5	2.0	1.0	2.0
a	0.5	0.7	0.9	1.1	1.3	0.5	0.7	0.9	1.1	1.3	0.5	0.7	0.9	1.0

Вариант 3

Исходные данные

```
Изначальное дифференциальное уравнение:
y' = (0.7 - 0.7*y**2)/(2.5*x**2 + y**2 + 1)
Изначальный шаг: 1.0
Заданая точночть: 0.001
Точка для проверки значений: X0 = 0.5
```

Примечание: x ** n (возвести x в степень n)

Программная реализация

Код метода Адамса:

```
def adams_method(func, f_y0, left, right, step, accuracy):
    tempY = -666.
    while True:
        X, Y = [], []
        n = int((right - left) / step)
        X.append(left)
        Y.append(f_y0)
        X.append(left + step)
        # Y1 находим с помощью модифицированного метода Эйлера
        Y.append(Y[0] + step * func.subs([(x, X[0] + 0.5 *
step),
                                               (y, Y[0] + 0.5 *
step * func.subs([(x, X[0]), (y, Y[0])]))]))
        for i in range(1, n):
            X.append(X[i] + step)
            Y.append(Y[i] + 0.5 * step * (3 * func.subs([(x,
X[i]), (y, Y[i])]) - func.subs([(x, X[i - 1]), (y, Y[i - 1])])))
        if abs(tempY - Y[-1]) < accuracy:
            break
        tempY = Y[-1]
        step /= 2
    return X, Y, step
```

Полученные результаты

```
Изначальное дифференциальное уравнение:
y' = (0.9 - 0.9*y**2.0)/(3.0*x**2.0 + y**2.0 + 1.0)
Изначальный шаг: 1.0
Заданная точность: 0.001
Точка для проверки значений: X0 = 0.5
Метод Адамса
Метод достиг указанной точности при h = 0.0625
Значение в точке X0:
0.346235392082318
```

Тестовый пример 1

```
Изначальное дифференциальное уравнение:
y' = x*y
Решение:
y = exp(x**2/2)
Изначальный шаг: 1.0
Заданная точность: 1e-05
Точка для проверки значений: X0 = 0.5
Значение функции в точке X0:
1.13314845306683
Метод Адамса
Метод достиг указанной точности при h = 0.0009765625
Значение в точке X0:
1.13314827764220
Погрешность:
1.75424626602805e-7
```

Тестовый пример 2

```
Изначальное дифференциальное уравнение:
y' = -tan(x)
Решение:
y = log(cos(x))
Изначальный шаг: 1.0
Заданная точность: 1e-05
Точка для проверки значений: X0 = 0.5
Значение функции в точке X0:
-0.130584240443723
Метод Адамса
Метод достиг указанной точности при h = 0.0009765625
Значение в точке X0:
-0.130584122181750
Погрешность:
-1.18261972714695e-7
```

Вывод

В ходе выполнения лабораторной работы я изучил решение задачи Коши для обыкновенных дифференциальных уравнений методом Адамса, написал программу его реализации на языке Python, правильность работы программы проверил на тестовых примерах.

На основании тестов можно сделать следующие выводы:

- Программа позволяет получить численное решение с заданной точностью (заданная точность в условиях лабораторной работы 10^-3);
- Существенным недостатком метода Адамса второго порядка является то обстоятельство, что для его применения надо знать дополнительно к начальному условию еще Y[-1] = Y(X[0] – h);
- Достоинством метода является то, что значение функции в каждой точке (x, y) вычисляется только один раз.