|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ «Информатика и системы управления»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Лабораторная работа № \_\_1\_\_**

|  |  |
| --- | --- |
| **Тема \_\_\_\_\_\_\_**Приближенный аналитический метод Пикара**\_\_\_\_\_\_\_**  **Студент \_\_\_\_**Оберган Т.М.**\_\_\_\_\_\_**  **Группа \_\_\_\_**ИУ7-65Б\_\_\_\_\_\_\_\_\_\_\_  **Оценка (баллы) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Преподаватель \_\_\_**Градов В.М.\_\_\_ |  |

Москва.

2020 г.

Целью данной лабораторной работы является анализ и сравнение численных методов и приближенного аналитического метода Пикара.

Существует задача Коши

Аналитического решения нет. Эту задачу можно решить методом Пикара:

Рассмотрим пример:

Тогда

Также эту задачу можно решить, используя численные методы:

Явная схема:

Неявная схема:

Рассмотрим неявную схему на примере:

Ниже приведен листинг реализованных методов.

Листинг 1.1: явная схема Эйлера **def** euler(n, h, x, y):

y\_out = []

**for** i **in** **range**(n):

**try**:

y += h \* func(x, y)

y\_out.append(y)

x += h

**except** OverflowError:

y\_out.append('overflow')

**for** j **in** **range**(i, n-1):

y\_out.append('-----')

**break**

**return** y\_out

Листинг 1.2: неявная схема Эйлера

**def** implicit\_euler(n, h, x, y):

y\_out = [y]

**for** i **in** **range**(n):

D = 1 - 4\*h\*(y + h\*((x + h)\*\*2))

**if** D < 0:

y\_out.append('D < 0')

**for** j **in** **range**(i, n-2):

y\_out.append('-----')

**break**

y = (1 - sqrt(D)) / (2\*h)

x += h

y\_out.append(y)

**return** y\_out

Листинг 1.3: метод Пикара

**def** picar(n, h, x):

**def** f1(a):

**return** a \*\* 3 / 3

**def** f2(a):

**return** f1(a) + a \*\* 7 / 63

**def** f3(a):

**return** f2(a) + (a \*\* 11) \* (2 / 2079) + (a \*\* 15) / 59535

**def** f4(a, f3):

**return** f3 + (a \*\* 15)\*(2 / 93555) + (a \*\* 19)\*(2 / 3393495) +

(a \*\* 19)\*(2 / 2488563) + (a \*\* 23)\*(2 / 86266215) + (a \*\* 23)\*(1 / 99411543) + (a \*\* 27)\*(2 / 3341878155) + (a \*\* 31)\*(1 / 109876902975)

y\_out = [[0, 0]]

**for** i **in** **range**(n-1):

x += h

y\_f3 = f3(x)

y\_out.append([y\_f3, f4(x, y\_f3)])

**return** y\_out