Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Кафедра вычислительных систем

**ОТЧЕТ**

по курсовой работе

по дисциплине «**Вычислительная математика**»

|  |  |  |
| --- | --- | --- |
| Выполнил:  студент гр. ИС-241  «21» апреля 2024 г. | \_\_\_\_\_\_\_\_\_\_\_\_ | /Кулик П. Е./ |
|  |  |  |
| Проверил:  преподаватель  «\_\_» апреля 2024 г. | \_\_\_\_\_\_\_\_\_\_\_\_ | /Чупрыно Л. А./ |

Оценка «\_\_\_\_\_\_\_\_\_\_\_\_\_»

Новосибирск 2024

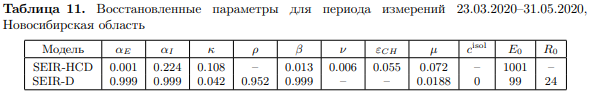
# ПОСТАНОВКА ЗАДАЧИ

В рамках данной курсовой требуется решить систему уравнений SEIR-D, моделирующую распространение инфекции COVID-19 в Новосибирской области. Система уравнений вместе с коэффициентами и начальными данными представлены в статье сибирского журнала вычислительной математики «Математическое моделирование и прогнозирование COVID-19 в Москве и Новосибирской области». Решение должно быть найдено с помощью метода Эйлера на интервале от 0 до 90 дней с точностью до 2 знаков после запятой.

Система уравнений для решения (из статьи):



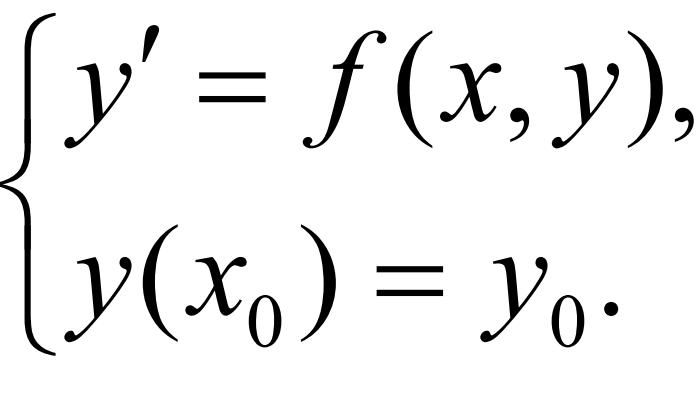
Параметры, используемые в ходе решения (из статьи):



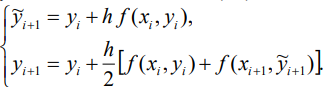
# ОПИСАНИЕ ТЕОРИИ МЕТОДА РЕШЕНИЯ ЗАДАЧИ

Постановка задачи предполагает решение системы с точностью до 2 знаков после запятой методом Эйлера, но так как классический метод Эйлера обладает недостаточной точностью для выполнения задачи (метод первого порядка точности), в решении использовался уточнённый метод Эйлера или же метод Эйлера-Коши, который является методом второго порядка точности и позволяет получить решение с требуемой точностью.

Метод Эйлера-Коши позволяет решать уравнения вида:



Решение задачи Коши выглядит следующим образом:



При данном методе решение ищется на отрезке [x0, b] с разбиением отрезка на n равных частей. При таком подходе , , i = 0, 1, …, n.

# СХЕМА АЛГОРИТМА РЕШЕНИЯ

Алгоритм реализует модель SEIRD, которая используется для моделирования распространения инфекционных заболеваний, таких как эпидемия COVID-19. Описание самого алгоритма:

1. Инициализация параметров модели: В начале кода определены параметры модели SEIRD (alphaE, alphaI, k, ro, beta, mu, c).
2. Функции дифференциальных уравнений: Заданы функции, описывающие изменение каждой из пяти категорий популяции (S – восприимчивые к заражению, E – бессимптомные больные, I – инфицированные с симптомами, R - вылечившиеся, D - умершие) со временем.
3. Вычисление общего количества людей: Функция count\_N() вычисляет общее количество людей в популяции на основе текущего состояния каждой из пяти категорий.
4. Метод Эйлера-Коши для решения дифференциальных уравнений: В основной функции SEIRD() реализован метод Эйлера-Коши для численного решения системы дифференциальных уравнений SEIRD. Он начинается с исходных значений каждой категории (S0, E0, I0, R0, D0) и шагом времени h.
5. Цикл вычислений: В цикле производится вычисление изменений состояний каждой категории с использованием метода Эйлера-Коши.
6. Запись результатов в файл: Результаты вычислений записываются в файл "data.txt" в формате "время от нулевого дня - количество выявленных случаев инфицирования".
7. Основная функция main(): Инициализирует начальные данные и вызывает функцию SEIRD() для каждого временного шага, сохраняя результаты в файл.

Этот алгоритм позволяет моделировать динамику распространения инфекционного заболевания в популяции.

# ЛИСТИНГ ПРОГРАММЫ С КОММЕНТАРИЯМИ

|  |  |  |
| --- | --- | --- |
| |  | | --- | | 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96 | | #include <stdio.h>  // Параметры модели  **double** alphaE = **0.999**;  **double** alphaI = **0.999**;  **double** k = **0.042**;  **double** ro = **0.952**;  **double** beta = **0.999**;  **double** mu = **0.0188**;  **double** c = **1**;  // Функции дифференциальных уравнений  **double** **dS**(**double** N, **double** S, **double** E, **double** I)  {  **return** (-c \* ((alphaI \* S \* I) / N) + ((alphaE \* S \* E) / N));  }  **double** **dE**(**double** N, **double** S, **double** E, **double** I)  {  **return** (c \* ((alphaI \* S \* I) / N) + ((alphaE \* S \* E) / N)  - ((k + ro) \* E));  }  **double** **dI**(**double** E, **double** I)  {  **return** (k \* E - beta \* I - mu \* I);  }  **double** **dR**(**double** E, **double** I)  {  **return** (beta \* I + ro \* E);  }  **double** **dD**(**double** I)  {  **return** (mu \* I);  }  // Вычисление количества всех людей  **double** **count\_N**(**double** S, **double** E, **double** I, **double** R, **double** D)  {  **return** S + E + I + R + D;  }  // Функция для вычисления по методу Эйлера-Коши  **double**  **SEIRD**(**double** S0,  **double** E0,  **double** I0,  **double** R0,  **double** D0,  **int** t0,  **double** t,  **double** h)  {  **double** S = S0, E = E0, I = I0, R = R0, D = D0;  **double** Sn, En, In;  **int** i = t0;  **int** N;  **for** (; i < t; i++) {  // Начальное приближение по методу Эйлера  N = count\_N(S, E, I, R, D);  Sn = S0 + h \* dS(N, S0, E0, I0);  En = E0 + h \* dE(N, S0, E0, I0);  In = I0 + h \* dI(E0, I0);  // Уточнение по методу Эйлера-Коши  S = S0 + h \* **0.5** \* (dS(N, S0, E0, I0) + dS(N, Sn, En, In));  E = E0 + h \* **0.5** \* (dE(N, S0, E0, I0) + dE(N, Sn, En, In));  I = I0 + h \* **0.5** \* (dI(E0, I0) + dI(En, In));  R = R0 + h \* **0.5** \* (dR(E0, I0) + dR(En, In));  D = D0 + h \* **0.5** \* (dD(I0) + dD(In));  S0 = S;  E0 = E;  I0 = I;  R0 = R;  D0 = D;  }  // Количество выявленных случаев инфицирования  **return** k \* E / **0.58**;  }  **int** **main**()  {  // Начальные данные  **double** S0 = **2798047**, E0 = **99**, I0 = **0**, R0 = **24**, D0 = **0**;  **FILE**\* file = fopen("data.txt", "w");  // Вычисление фунции на интервале и сохранение значений в файл  **for** (**int** i = **1**; i <= **100**; i++) {  fprintf(file, "%d %f**\n**", i, SEIRD(S0, E0, I0, R0, D0, **0**, i, **1**));  }  fclose(file);  } |

# РЕЗУЛЬТАТ РАБОТЫ ПРОГРАММЫ

Результатом работы программы является график, построенный на основе данных, посчитанных с помощью программы и записанных в файл “data.txt”. График представлен на рис.1.

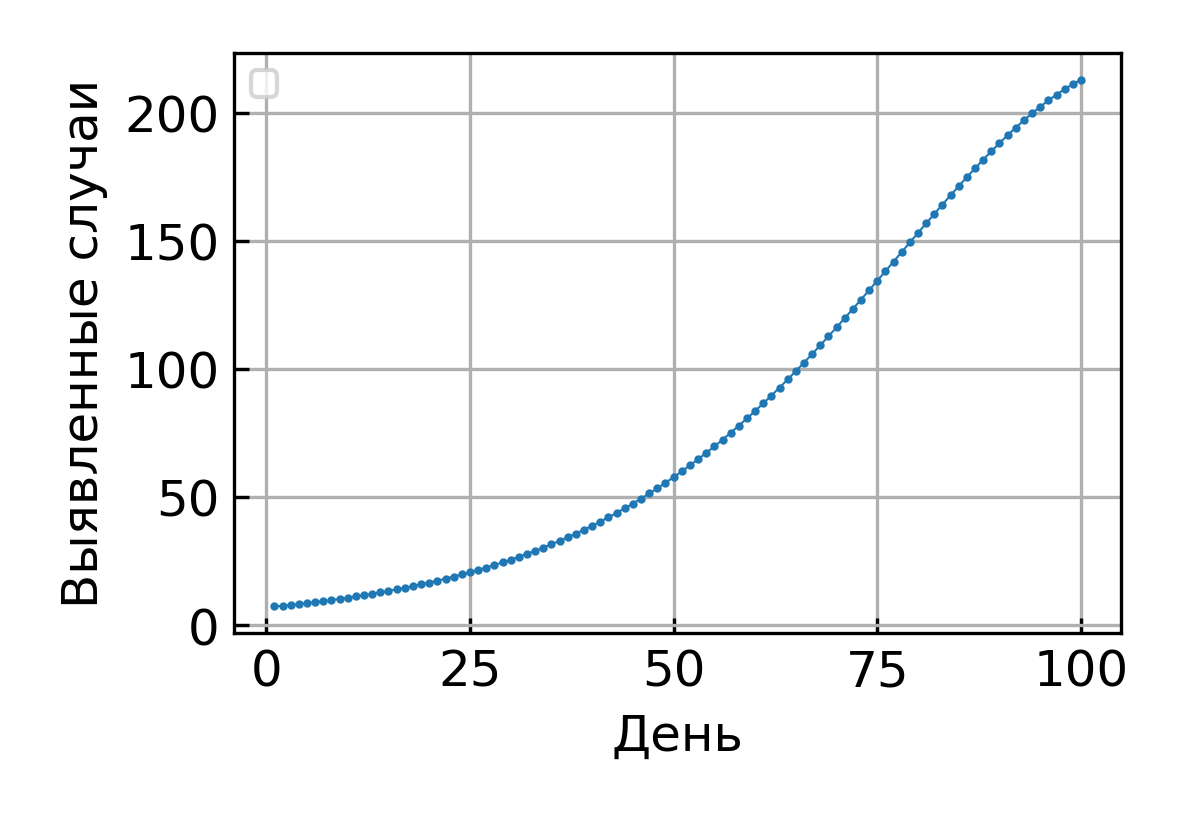


Рисунок 1. Количество выявленных случаев заражения на интервале от 0 до 100 дней.

Полученные данные сходятся с теми, что представлены в статье.