ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ
Федеральное государственное образовательное бюджетное учреждение
высшего профессионального образования
«Санкт-Петербургский государственный университет телекоммуникаций
им. проф. М.А. Бонч-Бруевича»

Факультет Информационных технологий и программной инженерии

Кафедра Программной инженерии и вычислительной техники

КУРСОВАЯ РАБОТА
по дисциплине:
«Программирование»
тема: Анализ сигнала на выходе электрической цепи

Передаточная характеристика – ___ вариант

Входной сигнал – ___ вариант

Выполнил студент(ка):
_____ _____
(Ф.И.О., № группы) *(подпись)*
Дата выполнения:
«_____» _____2025г
Проверил:
_____ _____
(Ф.И.О.) *(подпись)*

Санкт-Петербург
2025

СОДЕРЖАНИЕ

1 Постановка задачи

В курсовой работе необходимо для заданной электрической цепи по известному входному сигналу определить выходной сигнал для N равностоящих моментов времени, а затем определить некоторые его характеристики с погрешностью не более 1%.

1.1 Ход работы

— Произвести расчет входного и выходного сигнала в контрольных точках, используя при этом математический пакет wxMaxima;
— Написать текст программы на языке Си;
— Произвести запись полученных результатов в файлы данных;
— Используя математический пакет wxMaxima или LibreOffice Calc (электронные таблицы), построить графики зависимости напряжений входных и выходных сигналов от времени.
— Оформить пояснительную записку (doc-файл) по ГОСТ 19.402-78. ЕСПД. Описание программы. Плюс «Заключение» с личными выводами по работе.
— Объединить программу на Си и Wxmaxima (LibreOffice Calc), вызов отчета с помощью скрипта на Bash.
— Защитить работу преподавателю.

1.2 Задание для варианта №16

1) Сигнал на входе:

$$U_{\text{вх}}(t) = \begin{cases} 0 & \text{при } t \le t_1 \\ a(t - t_1) & \text{при } t_1 < t \le t_2 \\ a(t_2 - t_1) - b(t - t_2) & \text{при } t_2 < t \le t_3 \\ a(t_2 - t_1) - b(t_3 - t_2) - c(t - t_3) & \text{при } t > t_3 \end{cases} \quad \begin{vmatrix} a & = 20\ \text{В/с}; \\ b & = 0.5\ \text{В/с}; \\ c & = 17\ \text{В/с}; \\ t_{\text{нач}} & = 5\ \text{с}; \\ t_1 & = 10\ \text{с}; \\ t_2 & = 15\ \text{с}; \\ t_3 & = 45\ \text{с}; \\ t_{\text{кон}} & = 50\ \text{с}; \end{vmatrix}$$

2) Передаточная характеристика:

$$U_{\text{вых}} = \begin{cases} aU_{\text{вх}} + b & \text{при } U_{\text{вх}} \le U_{\text{вх1}} \\ aU_{\text{вх1}} + b & \text{при } U_{\text{вх}} > U_{\text{вх1}} \end{cases} \begin{vmatrix} a & = 2; \\ b & = -5 \text{ В}; \\ U_{\text{вх1}} & = 20 \text{ В}; \end{vmatrix}$$

3) Расчетный параметр:

Найти длительность переднего фронта импульса сигнала для Uвых.

2 Контрольный расчет

В данном разделе представлена реализация контрольных расчётов, выполненная в среде wxMaxima. Программа рассчитывает значения функций Uvx и Uvix на интервале, указанном в постановке задачи, с шагом, обеспечивающим вычисление на 500 точках. Такой подход позволяет достичь оптимальной точности при вычислительном эксперименте.

```
(%i238) /* Определение начальных условий для Uvx(t) */
    fpprintprec:5;
    tnach:5; t1:10; t2:15; t3:45; tkon:50;
    a:20;b:0.5;c:17;

    /* Определение функции Uvx(t) */
    Uvx(t):=if t <= t1 then 0
                   else if t1 < t and t <= t2 then a*(t-t1)
                   else if t2 < t and t <= t3 then a*(t2-t1)-b*(t-t2)
                   else a*(t2-t1) - b*(t3-t1) - c*(t-t3);

    /* Формирование массива времени */
    N:1500; /* Количество точек */
    dt:(tkon-tnach)/(N-1), numer;
    time_array:makelist(tnach + i*dt, i, 0, N-1);

    /* Вычисление значений Uvx(t) для массива времени */
    Uvx_values:map(Uvx, time_array);

    /* Определение начальных условий для Uvix */
    d:2; e:-5;
    Uvx1:20;

    /* Определение функции Uvix */
    Uvix(t):=if Uvx(t) <= Uvx1 then d*Uvx(t) + e
                    else d*Uvx1 + e;

    /* Вычисление значений Uvix для массива времени */
    Uvix_values:map(Uvix, time_array);

    /* Построение графика Uvx(t) */
    wxplot2d([discrete, time_array, Uvx_values], [gnuplot_preamble, "set
grid;"]);
    wxplot2d([discrete, time_array, Uvix_values], [gnuplot_preamble, "set
grid;"]);
(fpprintprec)   5

(tnach) 5

(t1)    10

(t2)    15

(t3)    45

(tkon)  50
```

```
(a) 20

(b) 0.5

(c) 17

(%o227) Uvx(t):=if t<=t1 then 0 else if t1<t and t<=t2 then a*(t-t1) else
if t2<t and t<=t3 then a*(t2-t1)-b*(t-t2) else a*(t2-t1)-b*(t3-t1)+-c*(t-
t3)
(N) 1500

(dt)    0.03002

(time_array)
[5,5.03,5.06,5.0901,5.1201,5.1501,5.1801,5.2101,5.2402,5.2702,5.3002,5.330
2,5.3602,5.3903,5.4203,5.4503,5.4803,5.5103,5.5404,5.5704,5.6004,5.6304,5.
6604,5.6905,5.7205,5.7505,5.7805,5.8105,
5.8406,5.8706,5.9006,5.9306,5.9606,5.9907,6.0207,6.0507,6.0807,6.1107,6.14
08,6.1708,6.2008,6.2308,6.2608,6.2909,6.3209,6.3509,6.3809,6.4109,6.441,6.
471,6.501,6.531,6.561,6.5911,6.6211,6.6511,
6.6811,6.7111,6.7412,6.7712,6.8012,6.8312,6.8612,6.8913,6.9213,6.9513,6.98
13,7.0113,7.0414,7.0714,7.1014,7.1314,7.1614,7.1915,7.2215,7.2515,7.2815,7
.3115,7.3416,7.3716,7.4016,7.4316,7.4616,
7.4917,7.5217,7.5517,7.5817,7.6117,7.6418,7.6718,7.7018,7.7318,7.7618,7.79
19,7.8219,7.8519,7.8819,7.9119,7.942,7.972,8.002,8.032,8.062,8.0921,8.1221
,8.1521,8.1821,8.2121,8.2422,8.2722,8.3022,
8.3322,8.3622,8.3923,8.4223,8.4523,8.4823,8.5123,8.5424,8.5724,8.6024,8.63
24,8.6624,8.6925,8.7225,8.7525,8.7825,8.8125,8.8426,8.8726,8.9026,8.9326,8
.9626,8.9927,9.0227,9.0527,9.0827,9.1127,
9.1428,9.1728,9.2028,9.2328,9.2628,9.2929,9.3229,9.3529,9.3829,9.4129,9.44
3,9.473,9.503,9.533,9.563,9.5931,9.6231,9.6531,9.6831,9.7131,9.7432,9.7732
,9.8032,9.8332,9.8632,9.8933,9.9233,9.9533,
9.9833,10.013,10.043,10.073,10.103,10.133,10.163,10.193,10.223,10.254,10.2
84,10.314,10.344,10.374,10.404,10.434,10.464,10.494,10.524,10.554,10.584,1
0.614,10.644,10.674,10.704,10.734,10.764,
10.794,10.824,10.854,10.884,10.914,10.944,10.974,11.004,11.034,11.064,11.0
94,11.124,11.154,11.184,11.214,11.244,11.274,11.304,11.334,11.364,11.394,1
1.424,11.454,11.484,11.514,11.544,11.574,
11.604,11.634,11.664,11.694,11.724,11.755,11.785,11.815,11.845,11.875,11.9
05,11.935,11.965,11.995,12.025,12.055,12.085,12.115,12.145,12.175,12.205,1
2.235,12.265,12.295,12.325,12.355,12.385,
12.415,12.445,12.475,12.505,12.535,12.565,12.595,12.625,12.655,12.685,12.7
15,12.745,12.775,12.805,12.835,12.865,12.895,12.925,12.955,12.985,13.015,1
3.045,13.075,13.105,13.135,13.165,13.195,
13.225,13.256,13.286,13.316,13.346,13.376,13.406,13.436,13.466,13.496,13.5
26,13.556,13.586,13.616,13.646,13.676,13.706,13.736,13.766,13.796,13.826,1
3.856,13.886,13.916,13.946,13.976,14.006,
14.036,14.066,14.096,14.126,14.156,14.186,14.216,14.246,14.276,14.306,14.3
36,14.366,14.396,14.426,14.456,14.486,14.516,14.546,14.576,14.606,14.636,1
4.666,14.696,14.726,14.757,14.787,14.817,
14.847,14.877,14.907,14.937,14.967,14.997,15.027,15.057,15.087,15.117,15.1
47,15.177,15.207,15.237,15.267,15.297,15.327,15.357,15.387,15.417,15.447,1
5.477,15.507,15.537,15.567,15.597,15.627,
15.657,15.687,15.717,15.747,15.777,15.807,15.837,15.867,15.897,15.927,15.9
57,15.987,16.017,16.047,16.077,16.107,16.137,16.167,16.197,16.227,16.258,1
6.288,16.318,16.348,16.378,16.408,16.438,
16.468,16.498,16.528,16.558,16.588,16.618,16.648,16.678,16.708,16.738,16.7
68,16.798,16.828,16.858,16.888,16.918,16.948,16.978,17.008,17.038,17.068,1
7.098,17.128,17.158,17.188,17.218,17.248,
17.278,17.308,17.338,17.368,17.398,17.428,17.458,17.488,17.518,17.548,17.5
78,17.608,17.638,17.668,17.698,17.728,17.759,17.789,17.819,17.849,17.879,1
```

7.909,17.939,17.969,17.999,18.029,18.059,
18.089,18.119,18.149,18.179,18.209,18.239,18.269,18.299,18.329,18.359,18.3
89,18.419,18.449,18.479,18.509,18.539,18.569,18.599,18.629,18.659,18.689,1
8.719,18.749,18.779,18.809,18.839,18.869,
18.899,18.929,18.959,18.989,19.019,19.049,19.079,19.109,19.139,19.169,19.1
99,19.229,19.26,19.29,19.32,19.35,19.38,19.41,19.44,19.47,19.5,19.53,19.56
,19.59,19.62,19.65,19.68,19.71,19.74,19.77,
19.8,19.83,19.86,19.89,19.92,19.95,19.98,20.01,20.04,20.07,20.1,20.13,20.1
6,20.19,20.22,20.25,20.28,20.31,20.34,20.37,20.4,20.43,20.46,20.49,20.52,2
0.55,20.58,20.61,20.64,20.67,20.7,20.73,20.761,
20.791,20.821,20.851,20.881,20.911,20.941,20.971,21.001,21.031,21.061,21.0
91,21.121,21.151,21.181,21.211,21.241,21.271,21.301,21.331,21.361,21.391,2
1.421,21.451,21.481,21.511,21.541,21.571,
21.601,21.631,21.661,21.691,21.721,21.751,21.781,21.811,21.841,21.871,21.9
01,21.931,21.961,21.991,22.021,22.051,22.081,22.111,22.141,22.171,22.201,2
2.231,22.262,22.292,22.322,22.352,22.382,
22.412,22.442,22.472,22.502,22.532,22.562,22.592,22.622,22.652,22.682,22.7
12,22.742,22.772,22.802,22.832,22.862,22.892,22.922,22.952,22.982,23.012,2
3.042,23.072,23.102,23.132,23.162,23.192,
23.222,23.252,23.282,23.312,23.342,23.372,23.402,23.432,23.462,23.492,23.5
22,23.552,23.582,23.612,23.642,23.672,23.702,23.732,23.763,23.793,23.823,2
3.853,23.883,23.913,23.943,23.973,24.003,
24.033,24.063,24.093,24.123,24.153,24.183,24.213,24.243,24.273,24.303,24.3
33,24.363,24.393,24.423,24.453,24.483,24.513,24.543,24.573,24.603,24.633,2
4.663,24.693,24.723,24.753,24.783,24.813,
24.843,24.873,24.903,24.933,24.963,24.993,25.023,25.053,25.083,25.113,25.1
43,25.173,25.203,25.233,25.264,25.294,25.324,25.354,25.384,25.414,25.444,2
5.474,25.504,25.534,25.564,25.594,25.624,
25.654,25.684,25.714,25.744,25.774,25.804,25.834,25.864,25.894,25.924,25.9
54,25.984,26.014,26.044,26.074,26.104,26.134,26.164,26.194,26.224,26.254,2
6.284,26.314,26.344,26.374,26.404,26.434,
26.464,26.494,26.524,26.554,26.584,26.614,26.644,26.674,26.704,26.734,26.7
65,26.795,26.825,26.855,26.885,26.915,26.945,26.975,27.005,27.035,27.065,2
7.095,27.125,27.155,27.185,27.215,27.245,
27.275,27.305,27.335,27.365,27.395,27.425,27.455,27.485,27.515,27.545,27.5
75,27.605,27.635,27.665,27.695,27.725,27.755,27.785,27.815,27.845,27.875,2
7.905,27.935,27.965,27.995,28.025,28.055,
28.085,28.115,28.145,28.175,28.205,28.235,28.266,28.296,28.326,28.356,28.3
86,28.416,28.446,28.476,28.506,28.536,28.566,28.596,28.626,28.656,28.686,2
8.716,28.746,28.776,28.806,28.836,28.866,
28.896,28.926,28.956,28.986,29.016,29.046,29.076,29.106,29.136,29.166,29.1
96,29.226,29.256,29.286,29.316,29.346,29.376,29.406,29.436,29.466,29.496,2
9.526,29.556,29.586,29.616,29.646,29.676,
29.706,29.736,29.767,29.797,29.827,29.857,29.887,29.917,29.947,29.977,30.0
07,30.037,30.067,30.097,30.127,30.157,30.187,30.217,30.247,30.277,30.307,3
0.337,30.367,30.397,30.427,30.457,30.487,
30.517,30.547,30.577,30.607,30.637,30.667,30.697,30.727,30.757,30.787,30.8
17,30.847,30.877,30.907,30.937,30.967,30.997,31.027,31.057,31.087,31.117,3
1.147,31.177,31.207,31.237,31.268,31.298,
31.328,31.358,31.388,31.418,31.448,31.478,31.508,31.538,31.568,31.598,31.6
28,31.658,31.688,31.718,31.748,31.778,31.808,31.838,31.868,31.898,31.928,3
1.958,31.988,32.018,32.048,32.078,32.108,
32.138,32.168,32.198,32.228,32.258,32.288,32.318,32.348,32.378,32.408,32.4
38,32.468,32.498,32.528,32.558,32.588,32.618,32.648,32.678,32.708,32.738,3
2.769,32.799,32.829,32.859,32.889,32.919,
32.949,32.979,33.009,33.039,33.069,33.099,33.129,33.159,33.189,33.219,33.2
49,33.279,33.309,33.339,33.369,33.399,33.429,33.459,33.489,33.519,33.549,3
3.579,33.609,33.639,33.669,33.699,33.729,
33.759,33.789,33.819,33.849,33.879,33.909,33.939,33.969,33.999,34.029,34.0
59,34.089,34.119,34.149,34.179,34.209,34.239,34.27,34.3,34.33,34.36,34.39,
34.42,34.45,34.48,34.51,34.54,34.57,34.6,

```
34.63,34.66,34.69,34.72,34.75,34.78,34.81,34.84,34.87,34.9,34.93,34.96,34.
99,35.02,35.05,35.08,35.11,35.14,35.17,35.2,35.23,35.26,35.29,35.32,35.35,
35.38,35.41,35.44,35.47,35.5,35.53,35.56,35.59,
35.62,35.65,35.68,35.71,35.74,35.771,35.801,35.831,35.861,35.891,35.921,35
.951,35.981,36.011,36.041,36.071,36.101,36.131,36.161,36.191,36.221,36.251
,36.281,36.311,36.341,36.371,36.401,36.431,
36.461,36.491,36.521,36.551,36.581,36.611,36.641,36.671,36.701,36.731,36.7
61,36.791,36.821,36.851,36.881,36.911,36.941,36.971,37.001,37.031,37.061,3
7.091,37.121,37.151,37.181,37.211,37.241,
37.272,37.302,37.332,37.362,37.392,37.422,37.452,37.482,37.512,37.542,37.5
72,37.602,37.632,37.662,37.692,37.722,37.752,37.782,37.812,37.842,37.872,3
7.902,37.932,37.962,37.992,38.022,38.052,
38.082,38.112,38.142,38.172,38.202,38.232,38.262,38.292,38.322,38.352,38.3
82,38.412,38.442,38.472,38.502,38.532,38.562,38.592,38.622,38.652,38.682,3
8.712,38.742,38.773,38.803,38.833,38.863,
38.893,38.923,38.953,38.983,39.013,39.043,39.073,39.103,39.133,39.163,39.1
93,39.223,39.253,39.283,39.313,39.343,39.373,39.403,39.433,39.463,39.493,3
9.523,39.553,39.583,39.613,39.643,39.673,
39.703,39.733,39.763,39.793,39.823,39.853,39.883,39.913,39.943,39.973,40.0
03,40.033,40.063,40.093,40.123,40.153,40.183,40.213,40.243,40.274,40.304,4
0.334,40.364,40.394,40.424,40.454,40.484,
40.514,40.544,40.574,40.604,40.634,40.664,40.694,40.724,40.754,40.784,40.8
14,40.844,40.874,40.904,40.934,40.964,40.994,41.024,41.054,41.084,41.114,4
1.144,41.174,41.204,41.234,41.264,41.294,
41.324,41.354,41.384,41.414,41.444,41.474,41.504,41.534,41.564,41.594,41.6
24,41.654,41.684,41.714,41.744,41.775,41.805,41.835,41.865,41.895,41.925,4
1.955,41.985,42.015,42.045,42.075,42.105,
42.135,42.165,42.195,42.225,42.255,42.285,42.315,42.345,42.375,42.405,42.4
35,42.465,42.495,42.525,42.555,42.585,42.615,42.645,42.675,42.705,42.735,4
2.765,42.795,42.825,42.855,42.885,42.915,
42.945,42.975,43.005,43.035,43.065,43.095,43.125,43.155,43.185,43.215,43.2
45,43.276,43.306,43.336,43.366,43.396,43.426,43.456,43.486,43.516,43.546,4
3.576,43.606,43.636,43.666,43.696,43.726,
43.756,43.786,43.816,43.846,43.876,43.906,43.936,43.966,43.996,44.026,44.0
56,44.086,44.116,44.146,44.176,44.206,44.236,44.266,44.296,44.326,44.356,4
4.386,44.416,44.446,44.476,44.506,44.536,
44.566,44.596,44.626,44.656,44.686,44.716,44.746,44.777,44.807,44.837,44.8
67,44.897,44.927,44.957,44.987,45.017,45.047,45.077,45.107,45.137,45.167,4
5.197,45.227,45.257,45.287,45.317,45.347,
45.377,45.407,45.437,45.467,45.497,45.527,45.557,45.587,45.617,45.647,45.6
77,45.707,45.737,45.767,45.797,45.827,45.857,45.887,45.917,45.947,45.977,4
6.007,46.037,46.067,46.097,46.127,46.157,
46.187,46.217,46.247,46.278,46.308,46.338,46.368,46.398,46.428,46.458,46.4
88,46.518,46.548,46.578,46.608,46.638,46.668,46.698,46.728,46.758,46.788,4
6.818,46.848,46.878,46.908,46.938,46.968,
46.998,47.028,47.058,47.088,47.118,47.148,47.178,47.208,47.238,47.268,47.2
98,47.328,47.358,47.388,47.418,47.448,47.478,47.508,47.538,47.568,47.598,4
7.628,47.658,47.688,47.718,47.748,47.779,
47.809,47.839,47.869,47.899,47.929,47.959,47.989,48.019,48.049,48.079,48.1
09,48.139,48.169,48.199,48.229,48.259,48.289,48.319,48.349,48.379,48.409,4
8.439,48.469,48.499,48.529,48.559,48.589,
48.619,48.649,48.679,48.709,48.739,48.769,48.799,48.829,48.859,48.889,48.9
19,48.949,48.979,49.009,49.039,49.069,49.099,49.129,49.159,49.189,49.219,4
9.249,49.28,49.31,49.34,49.37,49.4,49.43,
49.46,49.49,49.52,49.55,49.58,49.61,49.64,49.67,49.7,49.73,49.76,49.79,49.
82,49.85,49.88,49.91,49.94,49.97,50.0]
(Uvx_values)
[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0.26684,0.86724,1.4676,2.068,
2.6684,3.2688,3.8692,4.4696,5.07,5.6704,6.2708,6.8712,7.4716,8.072,8.6724,
9.2728,9.8732,10.474,11.074,11.674,12.275,12.875,13.476,14.076,14.676,15.2
77,15.877,16.478,17.078,17.678,18.279,
18.879,19.48,20.08,20.68,21.281,21.881,22.482,23.082,23.682,24.283,24.883,
25.484,26.084,26.684,27.285,27.885,28.486,29.086,29.686,30.287,30.887,31.4
88,32.088,32.688,33.289,33.889,34.49,35.09,
35.69,36.291,36.891,37.492,38.092,38.692,39.293,39.893,40.494,41.094,41.69
4,42.295,42.895,43.496,44.096,44.696,45.297,45.897,46.498,47.098,47.698,48
.299,48.899,49.5,50.1,50.7,51.301,51.901,
52.502,53.102,53.702,54.303,54.903,55.504,56.104,56.704,57.305,57.905,58.5
06,59.106,59.706,60.307,60.907,61.508,62.108,62.708,63.309,63.909,64.51,65
.11,65.71,66.311,66.911,67.512,68.112,
68.712,69.313,69.913,70.514,71.114,71.714,72.315,72.915,73.516,74.116,74.7
16,75.317,75.917,76.518,77.118,77.718,78.319,78.919,79.52,80.12,80.72,81.3
21,81.921,82.522,83.122,83.722,84.323,
84.923,85.524,86.124,86.724,87.325,87.925,88.526,89.126,89.726,90.327,90.9
27,91.528,92.128,92.728,93.329,93.929,94.53,95.13,95.73,96.331,96.931,97.5
32,98.132,98.732,99.333,99.933,99.987,
99.972,99.957,99.942,99.927,99.912,99.897,99.882,99.867,99.852,99.837,99.8
22,99.807,99.792,99.777,99.762,99.746,99.731,99.716,99.701,99.686,99.671,9
9.656,99.641,99.626,99.611,99.596,99.581,
99.566,99.551,99.536,99.521,99.506,99.491,99.476,99.461,99.446,99.431,99.4
16,99.401,99.386,99.371,99.356,99.341,99.326,99.311,99.296,99.281,99.266,9
9.251,99.236,99.221,99.206,99.191,99.176,
99.161,99.146,99.131,99.116,99.101,99.086,99.071,99.056,99.041,99.026,99.0
11,98.996,98.981,98.966,98.951,98.936,98.921,98.906,98.891,98.876,98.861,9
8.846,98.831,98.816,98.801,98.786,98.771,
98.756,98.741,98.726,98.711,98.696,98.681,98.666,98.651,98.636,98.621,98.6
06,98.591,98.576,98.561,98.546,98.531,98.516,98.501,98.486,98.471,98.456,9
8.441,98.426,98.411,98.396,98.381,98.366,
98.351,98.336,98.321,98.306,98.291,98.276,98.261,98.245,98.23,98.215,98.2,
98.185,98.17,98.155,98.14,98.125,98.11,98.095,98.08,98.065,98.05,98.035,98
.02,98.005,97.99,97.975,97.96,97.945,97.93,
97.915,97.9,97.885,97.87,97.855,97.84,97.825,97.81,97.795,97.78,97.765,97.
75,97.735,97.72,97.705,97.69,97.675,97.66,97.645,97.63,97.615,97.6,97.585,
97.57,97.555,97.54,97.525,97.51,97.495,97.48,
97.465,97.45,97.435,97.42,97.405,97.39,97.375,97.36,97.345,97.33,97.315,97
.3,97.285,97.27,97.255,97.24,97.225,97.21,97.195,97.18,97.165,97.15,97.135
,97.12,97.105,97.09,97.075,97.06,97.045,97.03
,97.015,97.0,96.985,96.97,96.955,96.94,96.925,96.91,96.895,96.88,96.865,96
.85,96.835,96.82,96.805,96.79,96.775,96.76,96.744,96.729,96.714,96.699,96.
684,96.669,96.654,96.639,96.624,96.609,
96.594,96.579,96.564,96.549,96.534,96.519,96.504,96.489,96.474,96.459,96.4
44,96.429,96.414,96.399,96.384,96.369,96.354,96.339,96.324,96.309,96.294,9
6.279,96.264,96.249,96.234,96.219,96.204,
96.189,96.174,96.159,96.144,96.129,96.114,96.099,96.084,96.069,96.054,96.0
39,96.024,96.009,95.994,95.979,95.964,95.949,95.934,95.919,95.904,95.889,9
5.874,95.859,95.844,95.829,95.814,95.799,
95.784,95.769,95.754,95.739,95.724,95.709,95.694,95.679,95.664,95.649,95.6
34,95.619,95.604,95.589,95.574,95.559,95.544,95.529,95.514,95.499,95.484,9
5.469,95.454,95.439,95.424,95.409,95.394,
95.379,95.364,95.349,95.334,95.319,95.304,95.289,95.274,95.259,95.243,95.2
28,95.213,95.198,95.183,95.168,95.153,95.138,95.123,95.108,95.093,95.078,9
5.063,95.048,95.033,95.018,95.003,94.988,
94.973,94.958,94.943,94.928,94.913,94.898,94.883,94.868,94.853,94.838,94.8
23,94.808,94.793,94.778,94.763,94.748,94.733,94.718,94.703,94.688,94.673,9
4.658,94.643,94.628,94.613,94.598,94.583,
94.568,94.553,94.538,94.523,94.508,94.493,94.478,94.463,94.448,94.433,94.4
18,94.403,94.388,94.373,94.358,94.343,94.328,94.313,94.298,94.283,94.268,9

4.253,94.238,94.223,94.208,94.193,94.178,
94.163,94.148,94.133,94.118,94.103,94.088,94.073,94.058,94.043,94.028,94.0
13,93.998,93.983,93.968,93.953,93.938,93.923,93.908,93.893,93.878,93.863,9
3.848,93.833,93.818,93.803,93.788,93.773,
93.758,93.742,93.727,93.712,93.697,93.682,93.667,93.652,93.637,93.622,93.6
07,93.592,93.577,93.562,93.547,93.532,93.517,93.502,93.487,93.472,93.457,9
3.442,93.427,93.412,93.397,93.382,93.367,
93.352,93.337,93.322,93.307,93.292,93.277,93.262,93.247,93.232,93.217,93.2
02,93.187,93.172,93.157,93.142,93.127,93.112,93.097,93.082,93.067,93.052,9
3.037,93.022,93.007,92.992,92.977,92.962,
92.947,92.932,92.917,92.902,92.887,92.872,92.857,92.842,92.827,92.812,92.7
97,92.782,92.767,92.752,92.737,92.722,92.707,92.692,92.677,92.662,92.647,9
2.632,92.617,92.602,92.587,92.572,92.557,
92.542,92.527,92.512,92.497,92.482,92.467,92.452,92.437,92.422,92.407,92.3
92,92.377,92.362,92.347,92.332,92.317,92.302,92.287,92.272,92.257,92.241,9
2.226,92.211,92.196,92.181,92.166,92.151,
92.136,92.121,92.106,92.091,92.076,92.061,92.046,92.031,92.016,92.001,91.9
86,91.971,91.956,91.941,91.926,91.911,91.896,91.881,91.866,91.851,91.836,9
1.821,91.806,91.791,91.776,91.761,91.746,
91.731,91.716,91.701,91.686,91.671,91.656,91.641,91.626,91.611,91.596,91.5
81,91.566,91.551,91.536,91.521,91.506,91.491,91.476,91.461,91.446,91.431,9
1.416,91.401,91.386,91.371,91.356,91.341,
91.326,91.311,91.296,91.281,91.266,91.251,91.236,91.221,91.206,91.191,91.1
76,91.161,91.146,91.131,91.116,91.101,91.086,91.071,91.056,91.041,91.026,9
1.011,90.996,90.981,90.966,90.951,90.936,
90.921,90.906,90.891,90.876,90.861,90.846,90.831,90.816,90.801,90.786,90.7
71,90.756,90.74,90.725,90.71,90.695,90.68,90.665,90.65,90.635,90.62,90.605
,90.59,90.575,90.56,90.545,90.53,90.515,90.5
,90.485,90.47,90.455,90.44,90.425,90.41,90.395,90.38,90.365,90.35,90.335,9
0.32,90.305,90.29,90.275,90.26,90.245,90.23,90.215,90.2,90.185,90.17,90.15
5,90.14,90.125,90.11,90.095,90.08,90.065,
90.05,90.035,90.02,90.005,89.99,89.975,89.96,89.945,89.93,89.915,89.9,89.8
85,89.87,89.855,89.84,89.825,89.81,89.795,89.78,89.765,89.75,89.735,89.72,
89.705,89.69,89.675,89.66,89.645,89.63,89.615
,89.6,89.585,89.57,89.555,89.54,89.525,89.51,89.495,89.48,89.465,89.45,89.
435,89.42,89.405,89.39,89.375,89.36,89.345,89.33,89.315,89.3,89.285,89.27,
89.255,89.239,89.224,89.209,89.194,89.179,
89.164,89.149,89.134,89.119,89.104,89.089,89.074,89.059,89.044,89.029,89.0
14,88.999,88.984,88.969,88.954,88.939,88.924,88.909,88.894,88.879,88.864,8
8.849,88.834,88.819,88.804,88.789,88.774,
88.759,88.744,88.729,88.714,88.699,88.684,88.669,88.654,88.639,88.624,88.6
09,88.594,88.579,88.564,88.549,88.534,88.519,88.504,88.489,88.474,88.459,8
8.444,88.429,88.414,88.399,88.384,88.369,
88.354,88.339,88.324,88.309,88.294,88.279,88.264,88.249,88.234,88.219,88.2
04,88.189,88.174,88.159,88.144,88.129,88.114,88.099,88.084,88.069,88.054,8
8.039,88.024,88.009,87.994,87.979,87.964,
87.949,87.934,87.919,87.904,87.889,87.874,87.859,87.844,87.829,87.814,87.7
99,87.784,87.769,87.754,87.738,87.723,87.708,87.693,87.678,87.663,87.648,8
7.633,87.618,87.603,87.588,87.573,87.558,
87.543,87.528,87.513,87.498,87.483,87.468,87.453,87.438,87.423,87.408,87.3
93,87.378,87.363,87.348,87.333,87.318,87.303,87.288,87.273,87.258,87.243,8
7.228,87.213,87.198,87.183,87.168,87.153,
87.138,87.123,87.108,87.093,87.078,87.063,87.048,87.033,87.018,87.003,86.9
88,86.973,86.958,86.943,86.928,86.913,86.898,86.883,86.868,86.853,86.838,8
6.823,86.808,86.793,86.778,86.763,86.748,
86.733,86.718,86.703,86.688,86.673,86.658,86.643,86.628,86.613,86.598,86.5
83,86.568,86.553,86.538,86.523,86.508,86.493,86.478,86.463,86.448,86.433,8
6.418,86.403,86.388,86.373,86.358,86.343,
86.328,86.313,86.298,86.283,86.268,86.253,86.237,86.222,86.207,86.192,86.1
77,86.162,86.147,86.132,86.117,86.102,86.087,86.072,86.057,86.042,86.027,8
6.012,85.997,85.982,85.967,85.952,85.937,

85.922,85.907,85.892,85.877,85.862,85.847,85.832,85.817,85.802,85.787,85.7
72,85.757,85.742,85.727,85.712,85.697,85.682,85.667,85.652,85.637,85.622,8
5.607,85.592,85.577,85.562,85.547,85.532,
85.517,85.502,85.487,85.472,85.457,85.442,85.427,85.412,85.397,85.382,85.3
67,85.352,85.337,85.322,85.307,85.292,85.277,85.262,85.247,85.232,85.217,8
5.202,85.187,85.172,85.157,85.142,85.127,
85.112,85.097,85.082,85.067,85.052,85.037,85.022,85.007,82.216,81.706,81.1
96,80.685,80.175,79.665,79.154,78.644,78.134,77.623,77.113,76.603,76.092,7
5.582,75.072,74.561,74.051,73.541,73.03,
72.52,72.01,71.499,70.989,70.479,69.968,69.458,68.948,68.437,67.927,67.417
,66.906,66.396,65.886,65.375,64.865,64.355,63.844,63.334,62.824,62.313,61.
803,61.293,60.782,60.272,59.762,59.251,
58.741,58.23,57.72,57.21,56.699,56.189,55.679,55.168,54.658,54.148,53.637,
53.127,52.617,52.106,51.596,51.086,50.575,50.065,49.555,49.044,48.534,48.0
24,47.513,47.003,46.493,45.982,45.472,
44.962,44.451,43.941,43.431,42.92,42.41,41.9,41.389,40.879,40.369,39.858,3
9.348,38.838,38.327,37.817,37.307,36.796,36.286,35.776,35.265,34.755,34.24
4,33.734,33.224,32.713,32.203,31.693,31.182,
30.672,30.162,29.651,29.141,28.631,28.12,27.61,27.1,26.589,26.079,25.569,2
5.058,24.548,24.038,23.527,23.017,22.507,21.996,21.486,20.976,20.465,19.95
5,19.445,18.934,18.424,17.914,17.403,16.893,
16.383,15.872,15.362,14.852,14.341,13.831,13.321,12.81,12.3,11.79,11.279,1
0.769,10.259,9.7482,9.2378,8.7275,8.2171,7.7068,7.1965,6.6861,6.1758,5.665
4,5.1551,4.6448,4.1344,3.6241,3.1137,2.6034,
2.0931,1.5827,1.0724,0.56204,0.051701,-0.45864,-0.96898,-1.4793,-1.9897,-
2.5]

(d) 2

(e) -5

(Uvx1) 20

(%o235) Uvix(t):=**if** Uvx(t)<=Uvx1 **then** d*Uvx(t)+e **else** d*Uvx1+e

(Uvix_values) [-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-
5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-
5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,
-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-
5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-
5,-5,-5,-5,-5,-5,-5,-
5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-
5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-5,-
5,-5,-5,-5,-5,-5,-5,-5,
-5,-5,-4.4663,-3.2655,-2.0647,-
0.86391,0.33689,1.5377,2.7385,3.9393,5.1401,6.3409,7.5417,8.7425,9.9433,11
.144,12.345,13.546,14.746,15.947,17.148,18.349,19.55,20.751,21.951,23.152,
24.353,
25.554,26.755,27.955,29.156,30.357,31.558,32.759,33.959,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3

```
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35
,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,35,3
5,35,35,35,35,35,35,35,35,35,35,
35,35,35,35,35,35,35,35,35,35,35,35,35,34.91,33.889,32.869,31.848,30.827,2
9.807,28.786,27.765,26.744,25.724,24.703,23.682,22.662,21.641,20.62,19.6,1
8.579,17.558,16.538,15.517,14.496,13.476,
12.455,11.434,10.414,9.3929,8.3722,7.3516,6.3309,5.3102,4.2895,3.2688,2.24
82,1.2275,0.2068,-0.81388,-1.8346,-2.8552,-3.8759,-4.8966,-5.9173,-6.938,-
7.9586,-8.9793,-10.0]
(%t237)
(%o237)

(%t238)
(%o238)
```

3 Таблица идентификаторов

Идентификатор	Тип	Назначение / Описание
Константы/Макросы		
N	int (макрос)	Максимальный размер массивов данных (t, Uvx, Uvix).
INPUT_SIZE	int (макрос)	Максимальный размер буфера для строкового ввода (input).
Переменные и Массивы		
t	float[]	Массив значений времени.
Uvx	float[]	Массив значений входного напряжения Uвх(t).
Uvix	float[]	Массив значений выходного напряжения Uвых(t).
n	int	Количество точек данных (размер массивов); параметр функций или локальная переменная.
dt	float	Шаг дискретизации по времени; параметр или локальная переменная.
choice	int	Переменная для хранения выбора пользователя в меню (main).
continueProgram	bool	Флаг продолжения работы основного цикла программы (main).
epsilon	float	Заданная точность для итерационного расчета (calculate_with_precision).
current_precision	float	Текущая рассчитанная погрешность (calculate_with_precision).
current_parameter	float	Текущее рассчитанное значение параметра (calculate_with_precision).
prev_parameter	float	Значение параметра на предыдущей итерации (calculate_with_precision).
duration	float	Рассчитанная длительность фронта (calc_leading_edge).
Umax, Umin	float	Максимальное/минимальное значение сигнала (calc_leading_edge).
U1, U2	float	Пороговые уровни для расчета фронта (calc_leading_edge).

fp, f1, f2, f3	FILE*	Указатели на файлы.
line	char[]	Буфер для чтения строки из файла (print_banner).
input	char[]	Буфер для строкового ввода пользователя (ask_user_continue).
Пользовательские Функции		
main	int()	Главная функция, точка входа, основной цикл программы.
forming_time	void()	Формирование массива времени t.
forming_Uvx	void()	Формирование массива входного напряжения Uvx.
forming_Uvix	void()	Формирование массива выходного напряжения Uvix.
input_n	int()	Ввод целочисленного значения n пользователем.
to_lower_str	void()	Преобразование строки к нижнему регистру.
ask_user_continue	bool()	Запрос у пользователя на продолжение работы программы.
print_banner	void()	Вывод заставки программы из файла zast.txt.
forming_table	void()	Формирование и вывод таблицы результатов в консоль.
output_in_file	void()	Запись массивов данных t, Uvx, Uvix в файлы.
calc_leading_edge	float()	Расчет длительности переднего фронта сигнала.
calcu-late_with_precision	void()	Итерационный расчет параметра с заданной точностью.
Стандартные Функции (Примеры)		
printf	int()	Форматированный вывод данных в консоль.
scanf	int()	Форматированный ввод данных из консоли.
fopen	FILE*()	Открытие файла.

`fclose`	`int()`	Закрытие файла.
`fprintf`	`int()`	Форматированный вывод данных в файл.
`fgets`	`char*()`	Чтение строки из файла.
`perror`	`void()`	Вывод системного сообщения об ошибке.
`getchar`	`int()`	Чтение одного символа из стандартного ввода.
`fabs`	`dou-ble()`	Вычисление абсолютного значения вещественного числа (из `math.h`).
`tolower`	`int()`	Преобразование символа в нижний регистр (из `ctype.h`).
`strcmp`	`int()`	Сравнение двух строк (из `string.h`).

4 Блок-схемы

В разделе представлены схемы алгоритмов ключевых функций программы. Для наглядного представления логики использованы диаграммы активностей стандарта UML, сгенерированные средствами PlantUML и являющиеся функциональным эквивалентом блок-схем по ГОСТ 19.701-90.

Для иллюстрации работы программы выбраны шесть наиболее репрезентативных функций, охватывающих общую структуру (`main`), основные вычислительные алгоритмы (`calculate_with_precision`, `calc_leading_edge`), формирование данных (`forming_Uvx`) и вывод/сохранение результатов (`forming_table`, `output_in_file`). Диаграммы для этих функций представлены ниже.

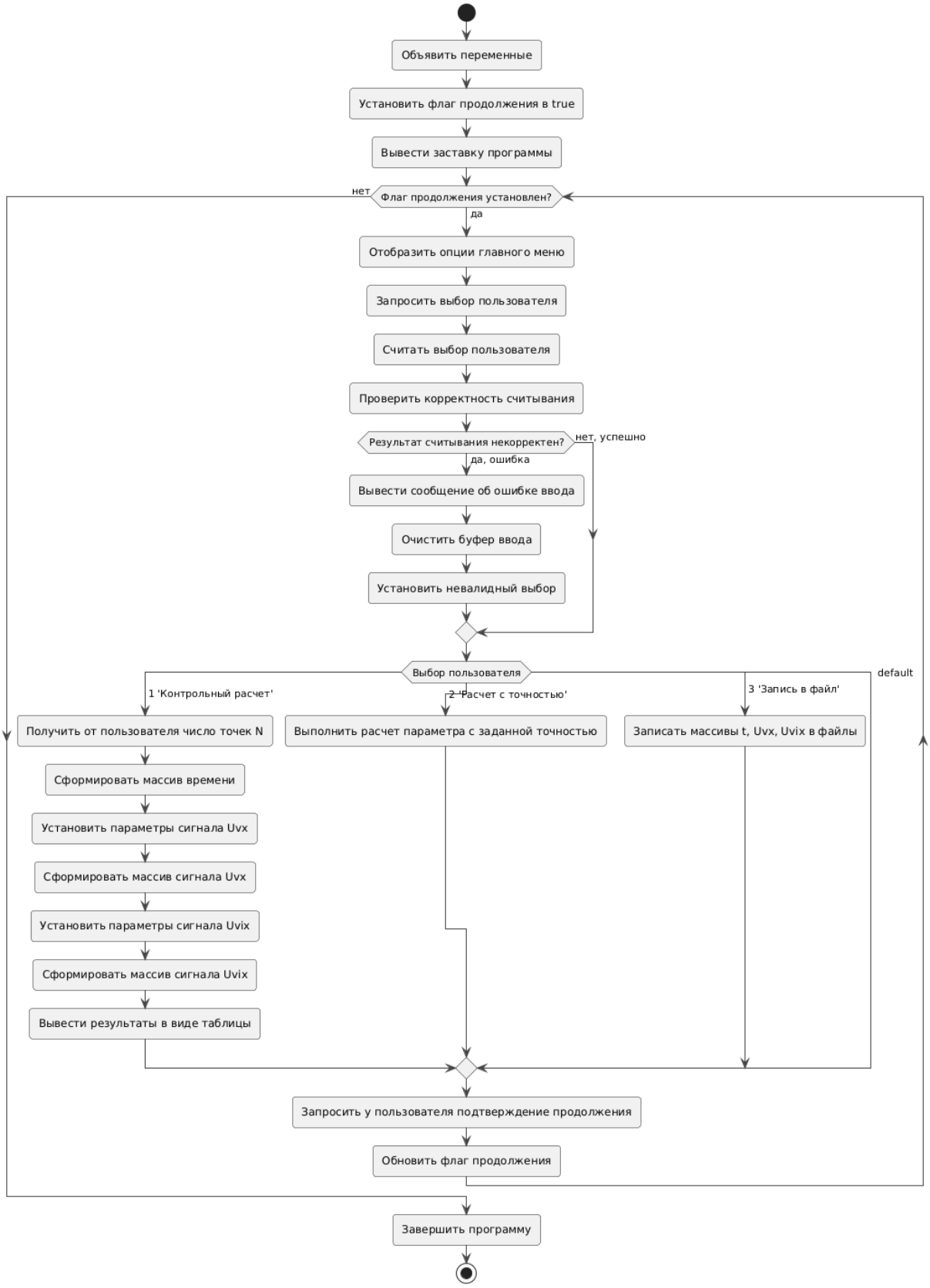

Рисунок 3 - Диаграмма активностей функции `main`

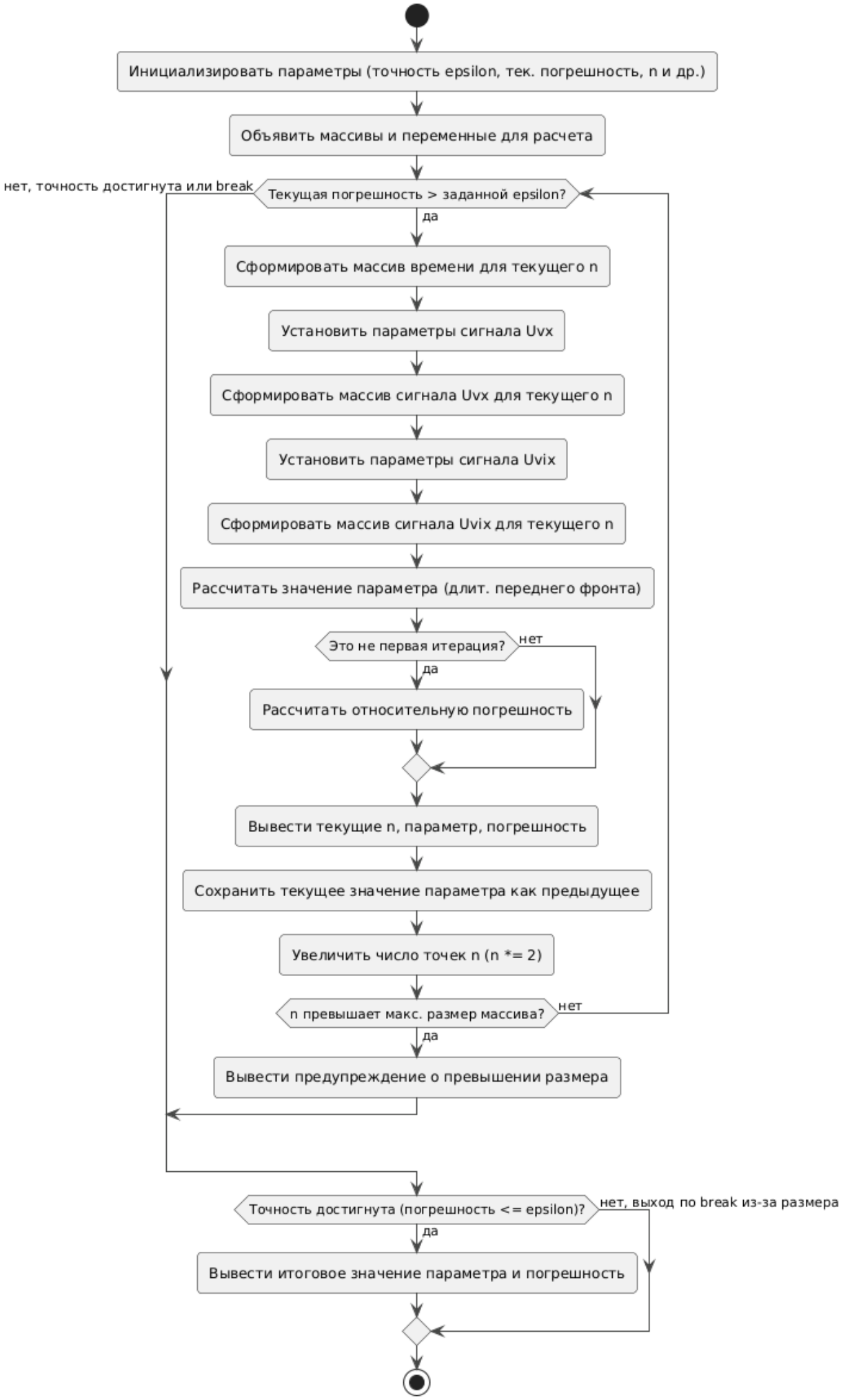
Инициализировать параметры (точность epsilon, тек. погрешность, n и др.)
Объявить массивы и переменные для расчета
Текущая погрешность > заданной epsilon?
нет, точность достигнута или break
да
Сформировать массив времени для текущего n
Установить параметры сигнала Uvx
Сформировать массив сигнала Uvx для текущего n
Установить параметры сигнала Uvix
Сформировать массив сигнала Uvix для текущего n
Рассчитать значение параметра (длит. переднего фронта)
Это не первая итерация?
нет
да
Рассчитать относительную погрешность
Вывести текущие n, параметр, погрешность
Сохранить текущее значение параметра как предыдущее
Увеличить число точек n (n *= 2)
n превышает макс. размер массива?
нет
да
Вывести предупреждение о превышении размера
Точность достигнута (погрешность <= epsilon)?
нет, выход по break из-за размера
да
Вывести итоговое значение параметра и погрешность

Рисунок 4 - Диаграмма активностей функции `calculate_with_precision`

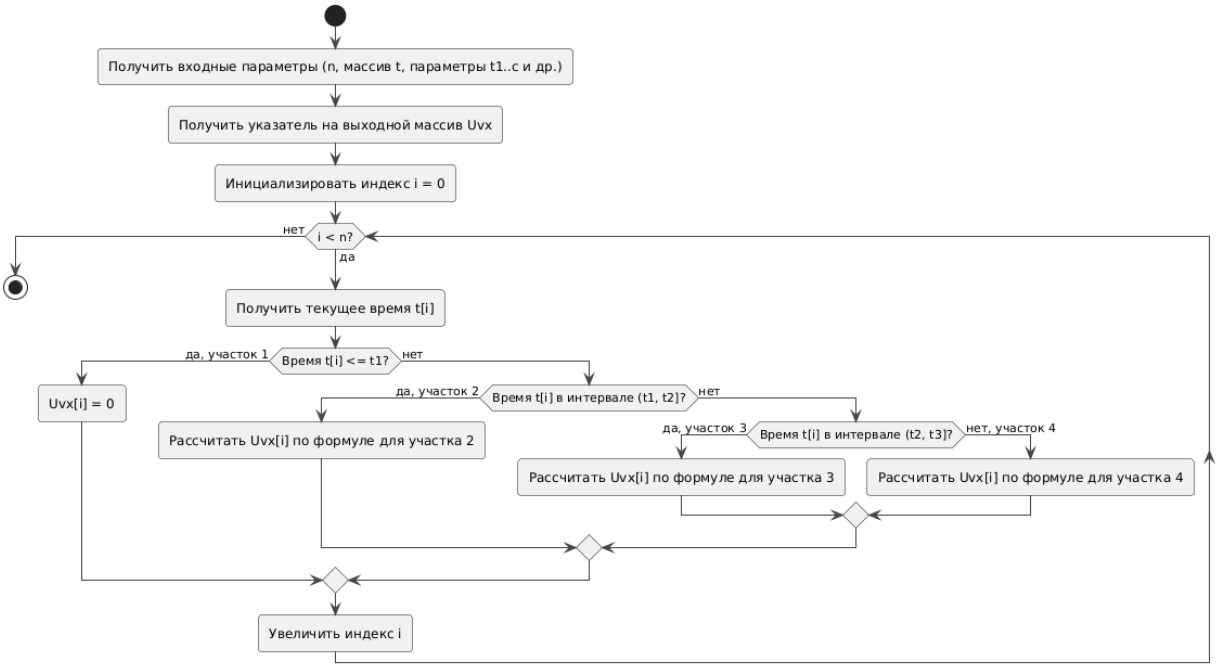

Рисунок 5 - Диаграмма активностей функции `forming_Uvx`

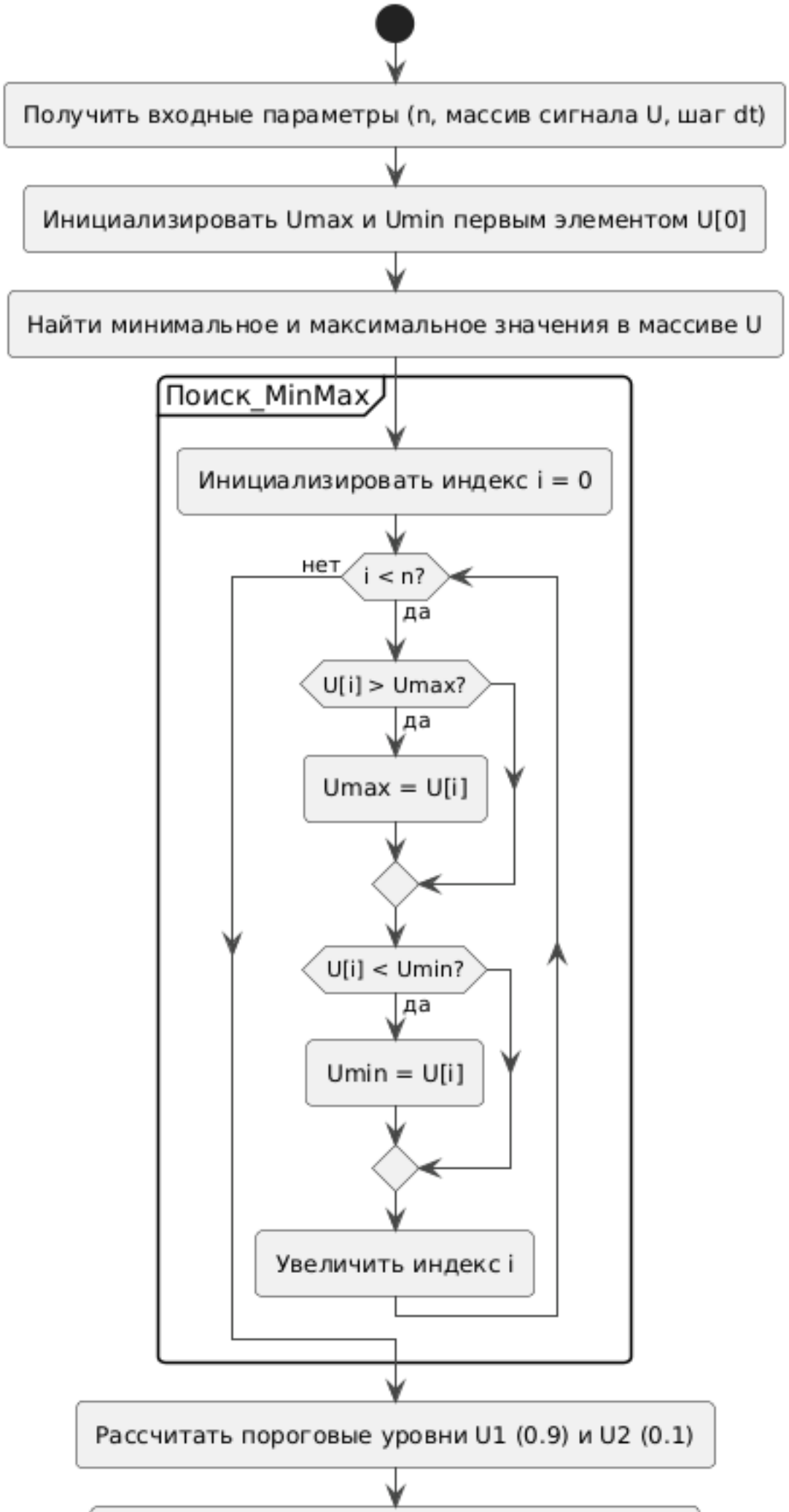
Получить входные параметры (n, массив сигнала U, шаг dt)
Инициализировать Umax и Umin первым элементом U[0]
Найти минимальное и максимальное значения в массиве U
Поиск_MinMax
Инициализировать индекс i = 0
нет
i < n?
да
U[i] > Umax?
да
Umax = U[i]
U[i] < Umin?
да
Umin = U[i]
Увеличить индекс i
Рассчитать пороговые уровни U1 (0.9) и U2 (0.1)

Рисунок 6 - Диаграмма активностей функции `calc_leading_edge`

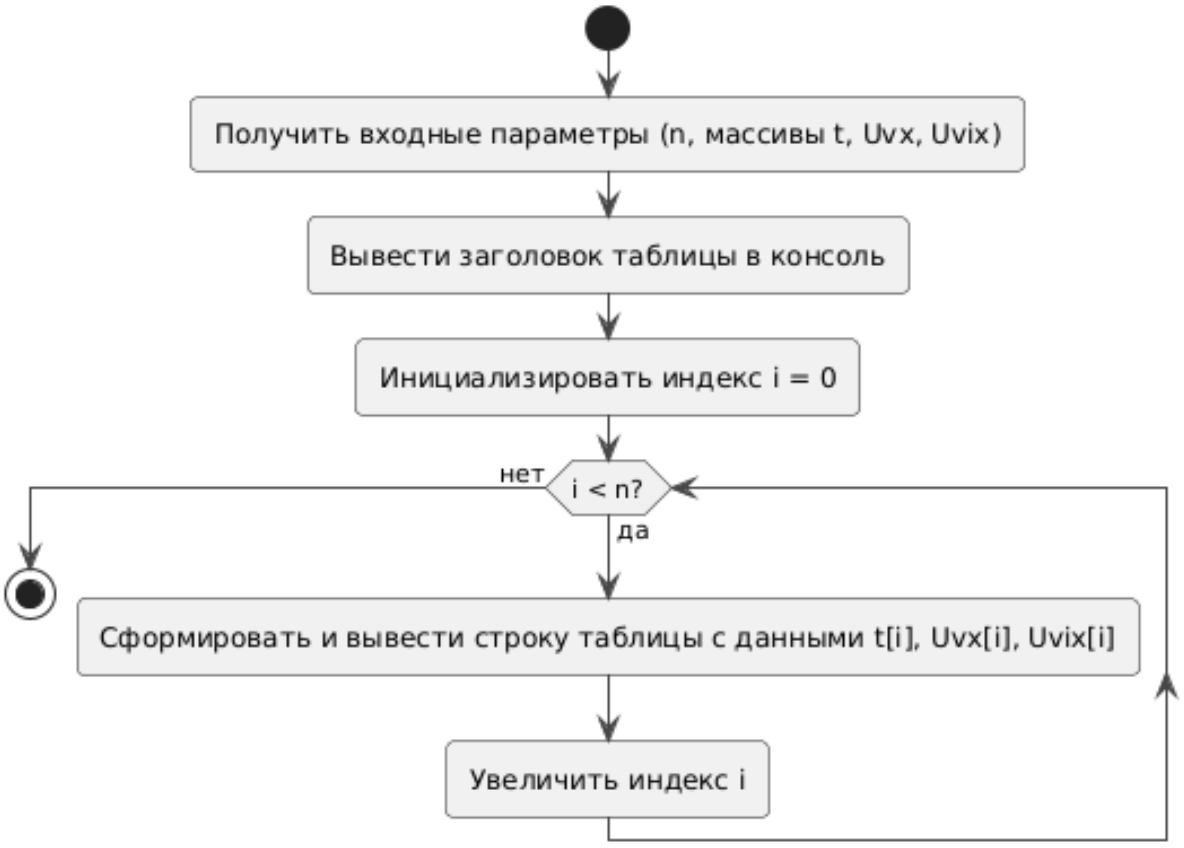

Рисунок 7 - Диаграмма активностей функции `forming_table`

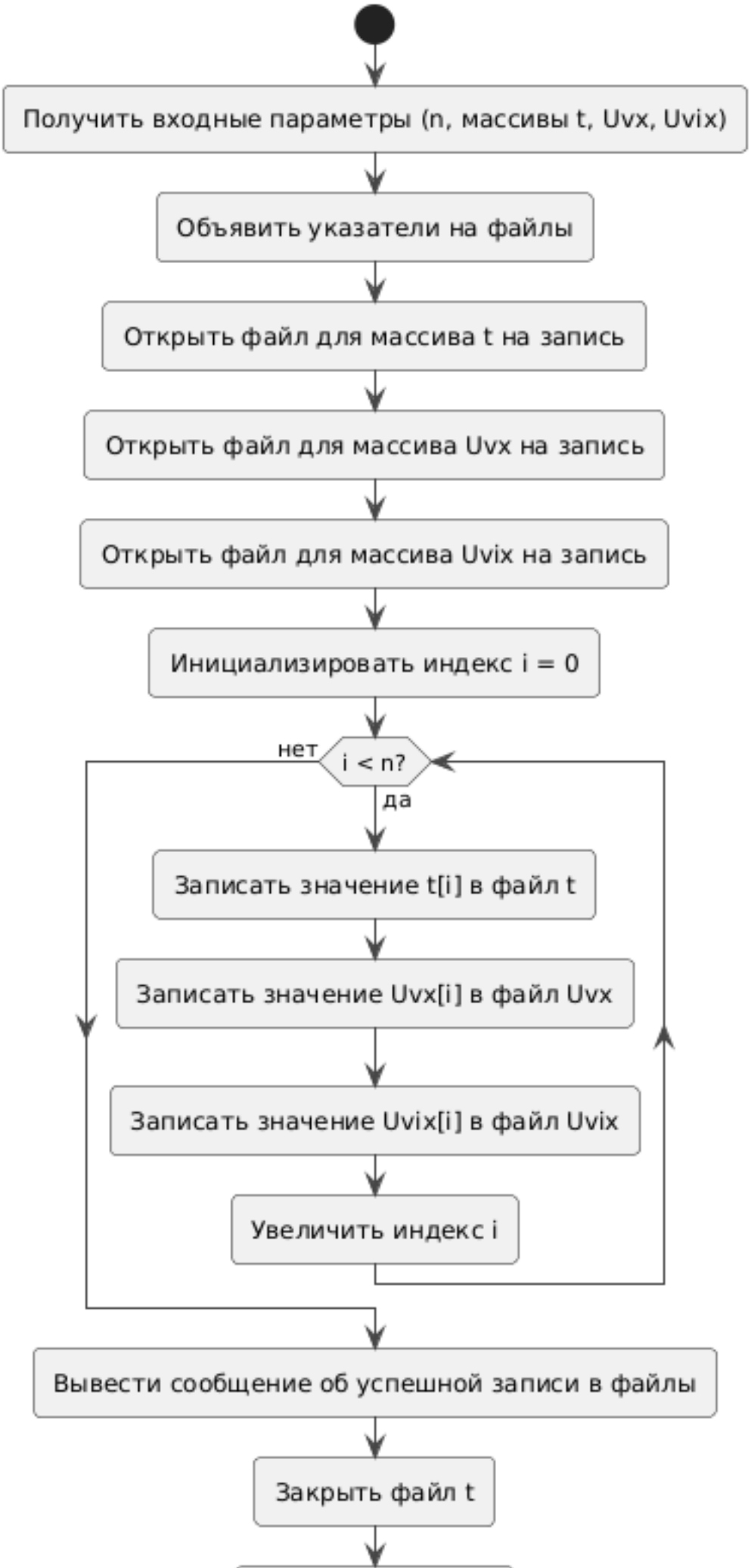
Получить входные параметры (n, массивы t, Uvx, Uvix)
Объявить указатели на файлы
Открыть файл для массива t на запись
Открыть файл для массива Uvx на запись
Открыть файл для массива Uvix на запись
Инициализировать индекс i = 0
нет
i < n?
да
Записать значение t[i] в файл t
Записать значение Uvx[i] в файл Uvx
Записать значение Uvix[i] в файл Uvix
Увеличить индекс i
Вывести сообщение об успешной записи в файлы
Закрыть файл t

Рисунок 8 - Диаграмма активностей функции `output_in_file`

5 Текст программы

5.1 Главный модуль программы

5.1.1 Файл: `signal_analysis.c`

Назначение: Организация основного цикла работы, взаимодействие с пользователем, вызов основных функций расчёта, вывода и сохранения данных.

```
#include <stdio.h>
#include <stdbool.h>
#include <math.h>

#include "input.h"
#include "output.h"
#include "forming.h"
#include "parameter.h"

#define N 1500

int main() {
    float t[N], Uvx[N], Uvix[N], dt;
    int n, choice;
    bool continueProgram = true;

    print_banner();

    while (continueProgram) {
        printf("\nМеню:\n");
        printf("1. Контрольный расчет для n точек\n");
        printf("2. Расчет параметра с заданной точностью\n");
        printf("3. Запись данных в файл\n");
        printf("Выберите опцию: ");

        int scanf_result = scanf(" %d", &choice); // Сохраняем результат
scanf

        if (scanf_result != 1) { // Проверяем, прочитала ли scanf ровно
одно целое число
            printf("Ошибка ввода. Пожалуйста, введите целое число.\n");
            // Очистка буфера ввода
            int c;
            while ((c = getchar()) != '\n' && c != EOF) {
                // Просто читаем и выбрасываем символы до конца строки или
файла
            }
            choice = -1; // Присваиваем 'choice' недопустимое значение,
чтобы избежать случайного выхода (если было 0)
                         // или повторного выполнения предыдущего case. -1
попадет в default.
        }

        switch (choice) {
        case 1:
            n = input_n();
```

```
            forming_time(n, t, &dt);

            float t1 = 10, t2 = 15, t3 = 45, a = 20, b = 0.5, c = 17;
            forming_Uvx(n, t, Uvx, t1, t2, t3, a, b, c);

            float Uvx1 = 20, d = 2, e = -5;
            forming_Uvix(n, Uvx, Uvix, Uvx1, d, e);

            forming_table(n, t, Uvx, Uvix);

            break;
        case 2:
            calculate_with_precision();
            break;
        case 3:
            output_in_file(n, t, Uvx, Uvix);
            break;
        }

        continueProgram = ask_user_continue();
    }

    return 0;
}
```

5.2 Модуль формирования данных

5.2.1 Файл: forming.h

Назначение: Заголовочный файл, объявляющий функции формирования временного массива и сигналов.

```
#ifndef FORMING_H
#define FORMING_H

void forming_time(int n, float *t, float *dt);
void forming_Uvx(int n, float *t, float *Uvx, float t1, float t2, float
t3, float a, float b, float c);
void forming_Uvix(int n, float *Uvx, float *Uvix, float Uvx1, float a,
float b);

#endif // FORMING_H
```

5.2.2 Файл: forming.c

Назначение: Реализация функций формирования временного массива и сигналов.

```
#include <stdio.h>
#include "forming.h"

// Функция формирования массива времени
void forming_time(int n, float *t, float *dt) {
    float tn = 5, tk = 50;
```

```
    *dt = (tk - tn) / (n - 1);
    for (int i = 0; i < n; i++) {
        t[i] = tn + i * (*dt);
    }
}

// Функция формирования массива Uvx
void forming_Uvx(int n, float *t, float *Uvx, float t1, float t2, float
t3, float a, float b, float c) {
    for (int i = 0; i < n; i++) {
        if (t[i] <= t1) {
            Uvx[i] = 0;
        } else if (t1 < t[i] && t[i] <= t2) {
            Uvx[i] = a * (t[i] - t1);
        } else if (t2 < t[i] && t[i] <= t3) {
            Uvx[i] = a * (t2 - t1) - b * (t[i] - t2);
        } else {
            Uvx[i] = a * (t2 - t1) - b * (t3 - t1) - c * (t[i] - t3);
        }
    }
}

// Функция формирования массива Uvix
void forming_Uvix(int n, float *Uvx, float *Uvix, float Uvx1, float a,
float b) {
    for (int i = 0; i < n; i++) {
        if (Uvx[i] <= Uvx1) {
            Uvix[i] = a * Uvx[i] + b;
        } else {
            Uvix[i] = a * Uvx1 + b;
        }
    }
}
```

5.3 Модуль расчёта параметров сигнала

5.3.1 Файл: `parameter.h`

Назначение: Заголовочный файл, объявляющий функции анализа параметров сигнала.

```
#ifndef PARAMETER_H
#define PARAMETER_H

float calc_leading_edge(int n, float *U, float dt);
void calculate_with_precision();

#endif // PARAMETER_H
```

5.3.2 Файл: parameter.c

Назначение: Реализация функций анализа параметров сигнала, включая вычисление длительности переднего фронта и итерационный расчёт с заданной точностью.

```c
#include <stdio.h>
#include <math.h>

#include "parameter.h"
#include "forming.h"

#define N 1500

// Функция расчета длительности переднего фронта импульса
float calc_leading_edge(int n, float *U, float dt) {
    // Находим максимальное и минимальное значения
    float Umax = U[0], Umin = U[0];
    for (int i = 0; i < n; i++) {
        if (U[i] > Umax) Umax = U[i];
        if (U[i] < Umin) Umin = U[i];
    }

    // Рассчитываем пороговые уровни
    float U1 = Umin + 0.9 * (Umax - Umin);
    float U2 = Umin + 0.1 * (Umax - Umin);

    // Считаем длительность заднего фронта
    float duration = 0;
    for (int i = 0; i < n - 1; i++) {
        if (U[i] < U1 && U[i] > U2 && U[i + 1] > U[i]) duration += dt;
    }

    return duration;
}

// Функция расчета параметра с заданной точностью
void calculate_with_precision() {
    float epsilon = 0.01; // Требуемая точность (1%)
    float current_precision = 1.0; // Текущая погрешность
    float prev_parameter = 1e10; // Начальное (очень большое) значение
    int n = 11; // Начальное количество точек

    float current_parameter, t[N], Uvx[N], Uvix[N], dt;

    while (current_precision > epsilon) {
        // Формирование массивов
        forming_time(n, t, &dt);

        float t1 = 10, t2 = 15, t3 = 45, a = 20, b = 0.5, c = 17;
        forming_Uvx(n, t, Uvx, t1, t2, t3, a, b, c);
        float Uvx1 = 20, d = 2, e = -5;
        forming_Uvix(n, Uvx, Uvix, Uvx1, d, e);

        // Расчет параметра
        current_parameter = calc_leading_edge(n, Uvix, dt);

        // Расчет погрешности
```

```
        if (prev_parameter != 1e10) {
            current_precision = fabs(prev_parameter - current_parameter) /
current_parameter;
        }

        printf("n = %d, параметр = %f, погрешность = %f\n", n,
current_parameter, current_precision);

        // Обновление значений для следующей итерации
        prev_parameter = current_parameter;
        n *= 2;
    }

    if (n >= N) {
        printf("Предупреждение: достигнут максимальный размер массива без
достижения требуемой точности\n");
    } else {
        printf("Итоговое значение параметра: %f (точность: %f)\n",
current_parameter, current_precision);
    }
}
```

5.4 Модуль ввода данных

5.4.1 Файл: input.h

Назначение: Заголовочный файл, объявляющий функции для ввода данных и обработки пользовательского ввода.

```
#ifndef INPUT_H
#define INPUT_H

int input_n();
void to_lower_str(char *str);
bool ask_user_continue(void);

#endif // INPUT_H
```

5.4.2 Файл: input.c

Назначение: Реализация функций для ввода количества точек, преобразования строки к нижнему регистру и запроса на продолжение работы.

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
#include <stdbool.h>
#include "input.h"

#define INPUT_SIZE 10

// Функция для ввода n
int input_n() {
    int value;
```

```
    int result;

    while (1) {
        printf("Введите целое число: ");
        result = scanf("%d", &value);
        if (result == 1) {
            break;  // Ввод успешно прочитан
        } else {
            printf("Ошибка ввода. Попробуйте еще раз.\n");
            // Очистка буфера ввода до символа новой строки
            while(getchar() != '\n');
        }
    }
    return value;
}

// Функция для преобразования строки к нижнему регистру
void to_lower_str(char *str) {
    for (int i = 0; str[i]; i++) {
        str[i] = tolower((unsigned char)str[i]);
    }
}

// Функция для запроса пользователя на продолжение работы
bool ask_user_continue(void) {
    char input[INPUT_SIZE];
    bool valid = false;

    do {
        printf("Хотите продолжить? (да/нет): ");
        // Считываем строку ввода
        if (scanf("%9s", input) != 1) {
            // Если произошла ошибка ввода, очищаем буфер
            while(getchar() != '\n');
            continue;
        }

        // Преобразуем строку в нижний регистр для корректного сравнения
        to_lower_str(input);

        // Сравниваем введённое значение с допустимыми
        if (strcmp(input, "да") == 0) {
            return true;
        } else if (strcmp(input, "нет") == 0) {
            return false;
        } else {
            printf("Некорректный ввод. Пожалуйста, введите 'да' или
'нет'.\n");
            valid = false;
        }
    } while (!valid);

    return false;
}
```

5.5 Модуль вывода и сохранения данных

5.5.1 Файл: output.h

Назначение: Заголовочный файл, объявляющий функции для вывода и сохранения результатов.

```
#ifndef OUTPUT_H
#define OUTPUT_H

void print_banner();
void forming_table(int n, float *t, float *Uvx, float *Uvix);
void output_in_file(int n, float *t, float *Uvx, float *Uvix);

#endif // BANNER_H
```

5.5.2 Файл: input.c

Назначение: Реализация функций для вывода заставки, вывода таблицы данных и сохранения массивов в файлы.

```
#include <stdio.h>
#include "output.h"

// Функция вывода заставки из файла
void print_banner() {
    FILE *fp = fopen("data/zast.txt", "r");
    if (!fp) {
        perror("Не удалось открыть файл заставки");
        return;
    }
    char line[256];
    while (fgets(line, sizeof(line), fp)) {
        printf("%s", line);
    }
    fclose(fp);
}

// Функция вывода данных в виде таблицы
void forming_table(int n, float *t, float *Uvx, float *Uvix) {
    printf("\n №   t      Uvx     Uvix\n");
    for (int i = 0; i < n; i++) {
        printf(" %3d %6.3f %6.3f %6.3f\n", i, t[i], Uvx[i], Uvix[i]);
    }
}

// Функция для записи данных в файл
void output_in_file(int n, float *t, float *Uvx, float *Uvix) {
    FILE *f1, *f2, *f3;
    f1 = fopen("data/array_t.txt", "w");
    f2 = fopen("data/array_Uvx.txt", "w");
    f3 = fopen("data/array_Uvix.txt", "w");
    for (int i = 0; i < n; i++) {
        fprintf(f1, "\n %6.3f", t[i]);
        fprintf(f2, "\n %6.3f", Uvx[i]);
```

```
            fprintf(f3, "\n %6.3f", Uvix[i]);
        }
        printf("Запись данных в файлы произведена успешно\n");
        fclose(f1);
        fclose(f2);
        fclose(f3);
}
```

6 Графики (обработка полученных результатов)

Графики, полученные в результате обработки данных, представлены ниже. Первый график отображает входной сигнал, второй — выходной сигнал. Оба графика построены на основе данных, рассчитанных программой на языке C. Форма графиков соответствует ожидаемой: входной сигнал имеет трапецеидальную форму, выходной сигнал — прямоугольную с плато. Полученные результаты подтверждают корректность работы программы и соответствие расчетов теоретическим ожиданиям.

```
 -->    /* Считывание данных из файлов, записанных программой, написанной
на C */
    t:read_list("../data/array_t.txt");
    Uvx:read_list("../data/array_Uvx.txt");
    Uvix:read_list("../data/array_Uvix.txt");

    /* Построение по этим данным графиков функций */
    wxplot2d([['discrete, t, Uvx]], [gnuplot_preamble, "set grid;"]);
    wxplot2d([['discrete, t, Uvix]], [gnuplot_preamble, "set grid;"]);
(t)
[5.0,5.03,5.06,5.09,5.12,5.15,5.18,5.21,5.24,5.27,5.3,5.33,5.36,5.39,5.42,
5.45,5.48,5.51,5.54,5.57,5.6,5.63,5.66,5.69,5.72,5.751,5.781,5.811,5.841,5
.871,5.901,5.931,5.961,5.991,6.021,6.051,6.081,6.111,6.141,6.171,6.201,6.2
31,6.261,6.291,6.321,6.351,6.381,6.411,6.441,6.471,6.501,6.531,6.561,6.591
,6.621,6.651,6.681,
6.711,6.741,6.771,6.801,6.831,6.861,6.891,6.921,6.951,6.981,7.011,7.041,7.
071,7.101,7.131,7.161,7.191,7.221,7.252,7.282,7.312,7.342,7.372,7.402,7.43
2,7.462,7.492,7.522,7.552,7.582,7.612,7.642,7.672,7.702,7.732,7.762,7.792,
7.822,7.852,7.882,7.912,7.942,7.972,8.002,8.032,8.062,8.092,8.122,8.152,8.
182,8.212,8.242,
8.272,8.302,8.332,8.362,8.392,8.422,8.452,8.482,8.512,8.542,8.572,8.602,8.
632,8.662,8.692,8.722,8.753,8.783,8.813,8.843,8.873,8.903,8.933,8.963,8.99
3,9.023,9.053,9.083,9.113,9.143,9.173,9.203,9.233,9.263,9.293,9.323,9.353,
9.383,9.413,9.443,9.473,9.503,9.533,9.563,9.593,9.623,9.653,9.683,9.713,9.
743,9.773,9.803,
9.833,9.863,9.893,9.923,9.953,9.983,10.013,10.043,10.073,10.103,10.133,10.
163,10.193,10.223,10.254,10.284,10.314,10.344,10.374,10.404,10.434,10.464,
10.494,10.524,10.554,10.584,10.614,10.644,10.674,10.704,10.734,10.764,10.7
94,10.824,10.854,10.884,10.914,10.944,10.974,11.004,11.034,11.064,11.094,1
1.124,11.154,
11.184,11.214,11.244,11.274,11.304,11.334,11.364,11.394,11.424,11.454,11.4
84,11.514,11.544,11.574,11.604,11.634,11.664,11.694,11.724,11.755,11.785,1
1.815,11.845,11.875,11.905,11.935,11.965,11.995,12.025,12.055,12.085,12.11
5,12.145,12.175,12.205,12.235,12.265,12.295,12.325,12.355,12.385,12.415,12
.445,12.475,
12.505,12.535,12.565,12.595,12.625,12.655,12.685,12.715,12.745,12.775,12.8
05,12.835,12.865,12.895,12.925,12.955,12.985,13.015,13.045,13.075,13.105,1
3.135,13.165,13.195,13.225,13.256,13.286,13.316,13.346,13.376,13.406,13.43
6,13.466,13.496,13.526,13.556,13.586,13.616,13.646,13.676,13.706,13.736,13
.766,13.796,
13.826,13.856,13.886,13.916,13.946,13.976,14.006,14.036,14.066,14.096,14.1
26,14.156,14.186,14.216,14.246,14.276,14.306,14.336,14.366,14.396,14.426,1
4.456,14.486,14.516,14.546,14.576,14.606,14.636,14.666,14.696,14.726,14.75
```

7,14.787,14.817,14.847,14.877,14.907,14.937,14.967,14.997,15.027,15.057,15
.087,15.117,
15.147,15.177,15.207,15.237,15.267,15.297,15.327,15.357,15.387,15.417,15.4
47,15.477,15.507,15.537,15.567,15.597,15.627,15.657,15.687,15.717,15.747,1
5.777,15.807,15.837,15.867,15.897,15.927,15.957,15.987,16.017,16.047,16.07
7,16.107,16.137,16.167,16.197,16.227,16.258,16.288,16.318,16.348,16.378,16
.408,16.438,
16.468,16.498,16.528,16.558,16.588,16.618,16.648,16.678,16.708,16.738,16.7
68,16.798,16.828,16.858,16.888,16.918,16.948,16.978,17.008,17.038,17.068,1
7.098,17.128,17.158,17.188,17.218,17.248,17.278,17.308,17.338,17.368,17.39
8,17.428,17.458,17.488,17.518,17.548,17.578,17.608,17.638,17.668,17.698,17
.728,17.759,
17.789,17.819,17.849,17.879,17.909,17.939,17.969,17.999,18.029,18.059,18.0
89,18.119,18.149,18.179,18.209,18.239,18.269,18.299,18.329,18.359,18.389,1
8.419,18.449,18.479,18.509,18.539,18.569,18.599,18.629,18.659,18.689,18.71
9,18.749,18.779,18.809,18.839,18.869,18.899,18.929,18.959,18.989,19.019,19
.049,19.079,
19.109,19.139,19.169,19.199,19.229,19.26,19.29,19.32,19.35,19.38,19.41,19.
44,19.47,19.5,19.53,19.56,19.59,19.62,19.65,19.68,19.71,19.74,19.77,19.8,1
9.83,19.86,19.89,19.92,19.95,19.98,20.01,20.04,20.07,20.1,20.13,20.16,20.1
9,20.22,20.25,20.28,20.31,20.34,20.37,20.4,20.43,20.46,20.49,20.52,20.55,2
0.58,20.61,20.64,
20.67,20.7,20.73,20.761,20.791,20.821,20.851,20.881,20.911,20.941,20.971,2
1.001,21.031,21.061,21.091,21.121,21.151,21.181,21.211,21.241,21.271,21.30
1,21.331,21.361,21.391,21.421,21.451,21.481,21.511,21.541,21.571,21.601,21
.631,21.661,21.691,21.721,21.751,21.781,21.811,21.841,21.871,21.901,21.931
,21.961,
21.991,22.021,22.051,22.081,22.111,22.141,22.171,22.201,22.231,22.262,22.2
92,22.322,22.352,22.382,22.412,22.442,22.472,22.502,22.532,22.562,22.592,2
2.622,22.652,22.682,22.712,22.742,22.772,22.802,22.832,22.862,22.892,22.92
2,22.952,22.982,23.012,23.042,23.072,23.102,23.132,23.162,23.192,23.222,23
.252,23.282,
23.312,23.342,23.372,23.402,23.432,23.462,23.492,23.522,23.552,23.582,23.6
12,23.642,23.672,23.702,23.732,23.763,23.793,23.823,23.853,23.883,23.913,2
3.943,23.973,24.003,24.033,24.063,24.093,24.123,24.153,24.183,24.213,24.24
3,24.273,24.303,24.333,24.363,24.393,24.423,24.453,24.483,24.513,24.543,24
.573,24.603,
24.633,24.663,24.693,24.723,24.753,24.783,24.813,24.843,24.873,24.903,24.9
33,24.963,24.993,25.023,25.053,25.083,25.113,25.143,25.173,25.203,25.233,2
5.264,25.294,25.324,25.354,25.384,25.414,25.444,25.474,25.504,25.534,25.56
4,25.594,25.624,25.654,25.684,25.714,25.744,25.774,25.804,25.834,25.864,25
.894,25.924,
25.954,25.984,26.014,26.044,26.074,26.104,26.134,26.164,26.194,26.224,26.2
54,26.284,26.314,26.344,26.374,26.404,26.434,26.464,26.494,26.524,26.554,2
6.584,26.614,26.644,26.674,26.704,26.734,26.765,26.795,26.825,26.855,26.88
5,26.915,26.945,26.975,27.005,27.035,27.065,27.095,27.125,27.155,27.185,27
.215,27.245,
27.275,27.305,27.335,27.365,27.395,27.425,27.455,27.485,27.515,27.545,27.5
75,27.605,27.635,27.665,27.695,27.725,27.755,27.785,27.815,27.845,27.875,2
7.905,27.935,27.965,27.995,28.025,28.055,28.085,28.115,28.145,28.175,28.20
5,28.235,28.266,28.296,28.326,28.356,28.386,28.416,28.446,28.476,28.506,28
.536,28.566,
28.596,28.626,28.656,28.686,28.716,28.746,28.776,28.806,28.836,28.866,28.8
96,28.926,28.956,28.986,29.016,29.046,29.076,29.106,29.136,29.166,29.196,2
9.226,29.256,29.286,29.316,29.346,29.376,29.406,29.436,29.466,29.496,29.52
6,29.556,29.586,29.616,29.646,29.676,29.706,29.736,29.767,29.797,29.827,29
.857,29.887,
29.917,29.947,29.977,30.007,30.037,30.067,30.097,30.127,30.157,30.187,30.2
17,30.247,30.277,30.307,30.337,30.367,30.397,30.427,30.457,30.487,30.517,3
0.547,30.577,30.607,30.637,30.667,30.697,30.727,30.757,30.787,30.817,30.84
7,30.877,30.907,30.937,30.967,30.997,31.027,31.057,31.087,31.117,31.147,31

.177,31.207,
31.237,31.268,31.298,31.328,31.358,31.388,31.418,31.448,31.478,31.508,31.5
38,31.568,31.598,31.628,31.658,31.688,31.718,31.748,31.778,31.808,31.838,3
1.868,31.898,31.928,31.958,31.988,32.018,32.048,32.078,32.108,32.138,32.16
8,32.198,32.228,32.258,32.288,32.318,32.348,32.378,32.408,32.438,32.468,32
.498,32.528,
32.558,32.588,32.618,32.648,32.678,32.708,32.738,32.769,32.799,32.829,32.8
59,32.889,32.919,32.949,32.979,33.009,33.039,33.069,33.099,33.129,33.159,3
3.189,33.219,33.249,33.279,33.309,33.339,33.369,33.399,33.429,33.459,33.48
9,33.519,33.549,33.579,33.609,33.639,33.669,33.699,33.729,33.759,33.789,33
.819,33.849,
33.879,33.909,33.939,33.969,33.999,34.029,34.059,34.089,34.119,34.149,34.1
79,34.209,34.239,34.27,34.3,34.33,34.36,34.39,34.42,34.45,34.48,34.51,34.5
4,34.57,34.6,34.63,34.66,34.69,34.72,34.75,34.78,34.81,34.84,34.87,34.9,34
.93,34.96,34.99,35.02,35.05,35.08,35.11,35.14,35.17,35.2,35.23,35.26,35.29
,35.32,35.35,
35.38,35.41,35.44,35.47,35.5,35.53,35.56,35.59,35.62,35.65,35.68,35.71,35.
74,35.771,35.801,35.831,35.861,35.891,35.921,35.951,35.981,36.011,36.041,3
6.071,36.101,36.131,36.161,36.191,36.221,36.251,36.281,36.311,36.341,36.37
1,36.401,36.431,36.461,36.491,36.521,36.551,36.581,36.611,36.641,36.671,36
.701,36.731,
36.761,36.791,36.821,36.851,36.881,36.911,36.941,36.971,37.001,37.031,37.0
61,37.091,37.121,37.151,37.181,37.211,37.241,37.272,37.302,37.332,37.362,3
7.392,37.422,37.452,37.482,37.512,37.542,37.572,37.602,37.632,37.662,37.69
2,37.722,37.752,37.782,37.812,37.842,37.872,37.902,37.932,37.962,37.992,38
.022,38.052,
38.082,38.112,38.142,38.172,38.202,38.232,38.262,38.292,38.322,38.352,38.3
82,38.412,38.442,38.472,38.502,38.532,38.562,38.592,38.622,38.652,38.682,3
8.712,38.742,38.773,38.803,38.833,38.863,38.893,38.923,38.953,38.983,39.01
3,39.043,39.073,39.103,39.133,39.163,39.193,39.223,39.253,39.283,39.313,39
.343,39.373,
39.403,39.433,39.463,39.493,39.523,39.553,39.583,39.613,39.643,39.673,39.7
03,39.733,39.763,39.793,39.823,39.853,39.883,39.913,39.943,39.973,40.003,4
0.033,40.063,40.093,40.123,40.153,40.183,40.213,40.243,40.274,40.304,40.33
4,40.364,40.394,40.424,40.454,40.484,40.514,40.544,40.574,40.604,40.634,40
.664,40.694,
40.724,40.754,40.784,40.814,40.844,40.874,40.904,40.934,40.964,40.994,41.0
24,41.054,41.084,41.114,41.144,41.174,41.204,41.234,41.264,41.294,41.324,4
1.354,41.384,41.414,41.444,41.474,41.504,41.534,41.564,41.594,41.624,41.65
4,41.684,41.714,41.744,41.775,41.805,41.835,41.865,41.895,41.925,41.955,41
.985,42.015,
42.045,42.075,42.105,42.135,42.165,42.195,42.225,42.255,42.285,42.315,42.3
45,42.375,42.405,42.435,42.465,42.495,42.525,42.555,42.585,42.615,42.645,4
2.675,42.705,42.735,42.765,42.795,42.825,42.855,42.885,42.915,42.945,42.97
5,43.005,43.035,43.065,43.095,43.125,43.155,43.185,43.215,43.245,43.276,43
.306,43.336,
43.366,43.396,43.426,43.456,43.486,43.516,43.546,43.576,43.606,43.636,43.6
66,43.696,43.726,43.756,43.786,43.816,43.846,43.876,43.906,43.936,43.966,4
3.996,44.026,44.056,44.086,44.116,44.146,44.176,44.206,44.236,44.266,44.29
6,44.326,44.356,44.386,44.416,44.446,44.476,44.506,44.536,44.566,44.596,44
.626,44.656,
44.686,44.716,44.746,44.777,44.807,44.837,44.867,44.897,44.927,44.957,44.9
87,45.017,45.047,45.077,45.107,45.137,45.167,45.197,45.227,45.257,45.287,4
5.317,45.347,45.377,45.407,45.437,45.467,45.497,45.527,45.557,45.587,45.61
7,45.647,45.677,45.707,45.737,45.767,45.797,45.827,45.857,45.887,45.917,45
.947,45.977,
46.007,46.037,46.067,46.097,46.127,46.157,46.187,46.217,46.247,46.278,46.3
08,46.338,46.368,46.398,46.428,46.458,46.488,46.518,46.548,46.578,46.608,4
6.638,46.668,46.698,46.728,46.758,46.788,46.818,46.848,46.878,46.908,46.93
8,46.968,46.998,47.028,47.058,47.088,47.118,47.148,47.178,47.208,47.238,47
.268,47.298,

```
47.328,47.358,47.388,47.418,47.448,47.478,47.508,47.538,47.568,47.598,47.6
28,47.658,47.688,47.718,47.749,47.779,47.809,47.839,47.869,47.899,47.929,4
7.959,47.989,48.019,48.049,48.079,48.109,48.139,48.169,48.199,48.229,48.25
9,48.289,48.319,48.349,48.379,48.409,48.439,48.469,48.499,48.529,48.559,48
.589,48.619,
48.649,48.679,48.709,48.739,48.769,48.799,48.829,48.859,48.889,48.919,48.9
49,48.979,49.009,49.039,49.069,49.099,49.129,49.159,49.189,49.219,49.25,49
.28,49.31,49.34,49.37,49.4,49.43,49.46,49.49,49.52,49.55,49.58,49.61,49.64
,49.67,49.7,49.73,49.76,49.79,49.82,49.85,49.88,49.91,49.94,49.97,50.0]
(Uvx)
[0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,
0.267,0.867,1.468,2.068,2.668,3.269,3.869,4.47,5.07,5.67,6.271,6.871,7.472
,8.072,8.672,9.273,9.873,10.474,11.074,11.674,12.275,12.875,13.476,14.076,
14.676,15.277,15.877,16.478,17.078,17.678,18.279,18.879,19.48,20.08,20.68,
21.281,21.881,22.482,23.082,23.682,24.283,24.883,25.484,26.084,26.684,27.2
85,27.885,
28.486,29.086,29.686,30.287,30.887,31.488,32.088,32.688,33.289,33.889,34.4
9,35.09,35.69,36.291,36.891,37.492,38.092,38.692,39.293,39.893,40.494,41.0
94,41.694,42.295,42.895,43.496,44.096,44.696,45.297,45.897,46.498,47.098,4
7.698,48.299,48.899,49.5,50.1,50.7,51.301,51.901,52.502,53.102,53.702,54.3
03,54.903,
55.504,56.104,56.704,57.305,57.905,58.506,59.106,59.706,60.307,60.907,61.5
08,62.108,62.708,63.309,63.909,64.51,65.11,65.71,66.311,66.911,67.512,68.1
12,68.712,69.313,69.913,70.514,71.114,71.714,72.315,72.915,73.516,74.116,7
4.716,75.317,75.917,76.518,77.118,77.718,78.319,78.919,79.52,80.12,80.72,8
1.321,81.921,
82.522,83.122,83.722,84.323,84.923,85.524,86.124,86.724,87.325,87.925,88.5
26,89.126,89.726,90.327,90.927,91.528,92.128,92.728,93.329,93.929,94.53,95
.13,95.73,96.331,96.931,97.532,98.132,98.732,99.333,99.933,99.987,99.972,9
9.957,99.942,99.927,99.912,99.897,99.882,99.867,99.852,99.837,99.822,99.80
7,99.792,
99.777,99.762,99.746,99.731,99.716,99.701,99.686,99.671,99.656,99.641,99.6
26,99.611,99.596,99.581,99.566,99.551,99.536,99.521,99.506,99.491,99.476,9
9.461,99.446,99.431,99.416,99.401,99.386,99.371,99.356,99.341,99.326,99.31
1,99.296,99.281,99.266,99.251,99.236,99.221,99.206,99.191,99.176,99.161,99
.146,99.131,
99.116,99.101,99.086,99.071,99.056,99.041,99.026,99.011,98.996,98.981,98.9
66,98.951,98.936,98.921,98.906,98.891,98.876,98.861,98.846,98.831,98.816,9
8.801,98.786,98.771,98.756,98.741,98.726,98.711,98.696,98.681,98.666,98.65
1,98.636,98.621,98.606,98.591,98.576,98.561,98.546,98.531,98.516,98.501,98
.486,98.471,
98.456,98.441,98.426,98.411,98.396,98.381,98.366,98.351,98.336,98.321,98.3
06,98.291,98.276,98.261,98.245,98.23,98.215,98.2,98.185,98.17,98.155,98.14
,98.125,98.11,98.095,98.08,98.065,98.05,98.035,98.02,98.005,97.99,97.975,9
7.96,97.945,97.93,97.915,97.9,97.885,97.87,97.855,97.84,97.825,97.81,97.79
5,97.78,97.765,
97.75,97.735,97.72,97.705,97.69,97.675,97.66,97.645,97.63,97.615,97.6,97.5
85,97.57,97.555,97.54,97.525,97.51,97.495,97.48,97.465,97.45,97.435,97.42,
97.405,97.39,97.375,97.36,97.345,97.33,97.315,97.3,97.285,97.27,97.255,97.
24,97.225,97.21,97.195,97.18,97.165,97.15,97.135,97.12,97.105,97.09,97.075
,97.06,97.045,
97.03,97.015,97.0,96.985,96.97,96.955,96.94,96.925,96.91,96.895,96.88,96.8
```

65,96.85,96.835,96.82,96.805,96.79,96.775,96.76,96.744,96.729,96.714,96.69
9,96.684,96.669,96.654,96.639,96.624,96.609,96.594,96.579,96.564,96.549,96
.534,96.519,96.504,96.489,96.474,96.459,96.444,96.429,96.414,96.399,96.384
,96.369,96.354
,96.339,96.324,96.309,96.294,96.279,96.264,96.249,96.234,96.219,96.204,96.
189,96.174,96.159,96.144,96.129,96.114,96.099,96.084,96.069,96.054,96.039,
96.024,96.009,95.994,95.979,95.964,95.949,95.934,95.919,95.904,95.889,95.8
74,95.859,95.844,95.829,95.814,95.799,95.784,95.769,95.754,95.739,95.724,9
5.709,95.694,
95.679,95.664,95.649,95.634,95.619,95.604,95.589,95.574,95.559,95.544,95.5
29,95.514,95.499,95.484,95.469,95.454,95.439,95.424,95.409,95.394,95.379,9
5.364,95.349,95.334,95.319,95.304,95.289,95.274,95.259,95.243,95.228,95.21
3,95.198,95.183,95.168,95.153,95.138,95.123,95.108,95.093,95.078,95.063,95
.048,95.033,
95.018,95.003,94.988,94.973,94.958,94.943,94.928,94.913,94.898,94.883,94.8
68,94.853,94.838,94.823,94.808,94.793,94.778,94.763,94.748,94.733,94.718,9
4.703,94.688,94.673,94.658,94.643,94.628,94.613,94.598,94.583,94.568,94.55
3,94.538,94.523,94.508,94.493,94.478,94.463,94.448,94.433,94.418,94.403,94
.388,94.373,
94.358,94.343,94.328,94.313,94.298,94.283,94.268,94.253,94.238,94.223,94.2
08,94.193,94.178,94.163,94.148,94.133,94.118,94.103,94.088,94.073,94.058,9
4.043,94.028,94.013,93.998,93.983,93.968,93.953,93.938,93.923,93.908,93.89
3,93.878,93.863,93.848,93.833,93.818,93.803,93.788,93.773,93.758,93.742,93
.727,93.712,
93.697,93.682,93.667,93.652,93.637,93.622,93.607,93.592,93.577,93.562,93.5
47,93.532,93.517,93.502,93.487,93.472,93.457,93.442,93.427,93.412,93.397,9
3.382,93.367,93.352,93.337,93.322,93.307,93.292,93.277,93.262,93.247,93.23
2,93.217,93.202,93.187,93.172,93.157,93.142,93.127,93.112,93.097,93.082,93
.067,93.052,
93.037,93.022,93.007,92.992,92.977,92.962,92.947,92.932,92.917,92.902,92.8
87,92.872,92.857,92.842,92.827,92.812,92.797,92.782,92.767,92.752,92.737,9
2.722,92.707,92.692,92.677,92.662,92.647,92.632,92.617,92.602,92.587,92.57
2,92.557,92.542,92.527,92.512,92.497,92.482,92.467,92.452,92.437,92.422,92
.407,92.392,
92.377,92.362,92.347,92.332,92.317,92.302,92.287,92.272,92.257,92.241,92.2
26,92.211,92.196,92.181,92.166,92.151,92.136,92.121,92.106,92.091,92.076,9
2.061,92.046,92.031,92.016,92.001,91.986,91.971,91.956,91.941,91.926,91.91
1,91.896,91.881,91.866,91.851,91.836,91.821,91.806,91.791,91.776,91.761,91
.746,91.731,
91.716,91.701,91.686,91.671,91.656,91.641,91.626,91.611,91.596,91.581,91.5
66,91.551,91.536,91.521,91.506,91.491,91.476,91.461,91.446,91.431,91.416,9
1.401,91.386,91.371,91.356,91.341,91.326,91.311,91.296,91.281,91.266,91.25
1,91.236,91.221,91.206,91.191,91.176,91.161,91.146,91.131,91.116,91.101,91
.086,91.071,
91.056,91.041,91.026,91.011,90.996,90.981,90.966,90.951,90.936,90.921,90.9
06,90.891,90.876,90.861,90.846,90.831,90.816,90.801,90.786,90.771,90.756,9
0.74,90.725,90.71,90.695,90.68,90.665,90.65,90.635,90.62,90.605,90.59,90.5
75,90.56,90.545,90.53,90.515,90.5,90.485,90.47,90.455,90.44,90.425,90.41,9
0.395,90.38,
90.365,90.35,90.335,90.32,90.305,90.29,90.275,90.26,90.245,90.23,90.215,90
.2,90.185,90.17,90.155,90.14,90.125,90.11,90.095,90.08,90.065,90.05,90.035
,90.02,90.005,89.99,89.975,89.96,89.945,89.93,89.915,89.9,89.885,89.87,89.
855,89.84,89.825,89.81,89.795,89.78,89.765,89.75,89.735,89.72,89.705,89.69
,89.675,89.66,
89.645,89.63,89.615,89.6,89.585,89.57,89.555,89.54,89.525,89.51,89.495,89.
48,89.465,89.45,89.435,89.42,89.405,89.39,89.375,89.36,89.345,89.33,89.315
,89.3,89.285,89.27,89.255,89.239,89.224,89.209,89.194,89.179,89.164,89.149
,89.134,89.119,89.104,89.089,89.074,89.059,89.044,89.029,89.014,88.999,88.
984,88.969,
88.954,88.939,88.924,88.909,88.894,88.879,88.864,88.849,88.834,88.819,88.8
04,88.789,88.774,88.759,88.744,88.729,88.714,88.699,88.684,88.669,88.654,8

```
8.639,88.624,88.609,88.594,88.579,88.564,88.549,88.534,88.519,88.504,88.48
9,88.474,88.459,88.444,88.429,88.414,88.399,88.384,88.369,88.354,88.339,88
.324,88.309,
88.294,88.279,88.264,88.249,88.234,88.219,88.204,88.189,88.174,88.159,88.1
44,88.129,88.114,88.099,88.084,88.069,88.054,88.039,88.024,88.009,87.994,8
7.979,87.964,87.949,87.934,87.919,87.904,87.889,87.874,87.859,87.844,87.82
9,87.814,87.799,87.784,87.769,87.754,87.738,87.723,87.708,87.693,87.678,87
.663,87.648,
87.633,87.618,87.603,87.588,87.573,87.558,87.543,87.528,87.513,87.498,87.4
83,87.468,87.453,87.438,87.423,87.408,87.393,87.378,87.363,87.348,87.333,8
7.318,87.303,87.288,87.273,87.258,87.243,87.228,87.213,87.198,87.183,87.16
8,87.153,87.138,87.123,87.108,87.093,87.078,87.063,87.048,87.033,87.018,87
.003,86.988,
86.973,86.958,86.943,86.928,86.913,86.898,86.883,86.868,86.853,86.838,86.8
23,86.808,86.793,86.778,86.763,86.748,86.733,86.718,86.703,86.688,86.673,8
6.658,86.643,86.628,86.613,86.598,86.583,86.568,86.553,86.538,86.523,86.50
8,86.493,86.478,86.463,86.448,86.433,86.418,86.403,86.388,86.373,86.358,86
.343,86.328,
86.313,86.298,86.283,86.268,86.253,86.237,86.222,86.207,86.192,86.177,86.1
62,86.147,86.132,86.117,86.102,86.087,86.072,86.057,86.042,86.027,86.012,8
5.997,85.982,85.967,85.952,85.937,85.922,85.907,85.892,85.877,85.862,85.84
7,85.832,85.817,85.802,85.787,85.772,85.757,85.742,85.727,85.712,85.697,85
.682,85.667,
85.652,85.637,85.622,85.607,85.592,85.577,85.562,85.547,85.532,85.517,85.5
02,85.487,85.472,85.457,85.442,85.427,85.412,85.397,85.382,85.367,85.352,8
5.337,85.322,85.307,85.292,85.277,85.262,85.247,85.232,85.217,85.202,85.18
7,85.172,85.157,85.142,85.127,85.112,85.097,85.082,85.067,85.052,85.037,85
.022,85.007,
82.216,81.706,81.196,80.685,80.175,79.665,79.154,78.644,78.134,77.623,77.1
13,76.603,76.092,75.582,75.072,74.561,74.051,73.541,73.03,72.52,72.01,71.4
99,70.989,70.479,69.968,69.458,68.948,68.437,67.927,67.417,66.906,66.396,6
5.886,65.375,64.865,64.355,63.844,63.334,62.824,62.313,61.803,61.293,60.78
2,60.272,
59.762,59.251,58.741,58.23,57.72,57.21,56.699,56.189,55.679,55.168,54.658,
54.148,53.637,53.127,52.617,52.106,51.596,51.086,50.575,50.065,49.555,49.0
44,48.534,48.024,47.513,47.003,46.493,45.982,45.472,44.962,44.451,43.941,4
3.431,42.92,42.41,41.9,41.389,40.879,40.369,39.858,39.348,38.838,38.327,37
.817,37.307,
36.796,36.286,35.775,35.265,34.755,34.245,33.734,33.224,32.713,32.203,31.6
93,31.182,30.672,30.162,29.651,29.141,28.631,28.12,27.61,27.1,26.589,26.07
9,25.569,25.058,24.548,24.038,23.527,23.017,22.507,21.996,21.486,20.976,20
.465,19.955,19.445,18.934,18.424,17.914,17.403,16.893,16.383,15.872,15.362
,14.852,
14.341,13.831,13.321,12.81,12.3,11.79,11.279,10.769,10.258,9.748,9.238,8.7
27,8.217,7.707,7.196,6.686,6.176,5.665,5.155,4.645,4.134,3.624,3.114,2.603
,2.093,1.583,1.072,0.562,0.052,-0.459,-0.969,-1.479,-1.99,-2.5]
(Uvix)  [-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-5.0,-
5.0,-5.0,-5.0,-4.466,-3.266,-2.065,-
0.864,0.337,1.538,2.738,3.939,5.14,6.341,7.542,8.743,9.943,
```

11.144,12.345,13.546,14.746,15.947,17.148,18.349,19.55,20.751,21.951,23.15
2,24.353,25.554,26.755,27.955,29.156,30.357,31.558,32.758,33.959,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0

```
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0
,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.
0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35
.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,35.0,3
5.0,35.0,35.0,35.0,35.0,
35.0,35.0,35.0,35.0,34.91,33.889,32.869,31.848,30.827,29.807,28.786,27.765
,26.744,25.724,24.703,23.682,22.662,21.641,20.62,19.6,18.579,17.558,16.538
,15.517,14.496,13.476,12.455,11.434,10.414,9.393,8.372,7.352,6.331,5.31,4.
29,3.269,2.248,1.227,0.207,-0.814,-1.835,-2.855,-3.876,-4.897,-5.917,-
6.938,-7.959,-
8.979,-10.0]
(%t4)
(%o4)

(%t5)
(%o5)
```

ЗАКЛЮЧЕНИЕ

В рамках данной курсовой работы была успешно решена задача анализа сигнала на выходе электрической цепи. Была разработана программа на языке Си, моделирующая прохождение входного сигнала (вариант 16) через цепь с заданной передаточной характеристикой (реализованной в коде) и вычисляющая выходной сигнал (`Uvix`).

Ключевой задачей являлся расчет длительности переднего фронта выходного сигнала с погрешностью не более 1%. Эта задача была выполнена с использованием итерационного алгоритма. Итоговое значение параметра составило 0.896159 с точностью 0.14, что соответствует требованиям задания.

Программа также реализует сохранение результатов расчета в файлы и предоставляет пользователю текстовое меню. Управление проектом осуществляется с помощью скрипта на Bash.

Работа позволила закрепить навыки программирования на языке Си (реализация численных методов, работа с файлами), использования математического пакета wxMaxima для контрольных расчетов и визуализации данных, а также применения скриптового языка Bash для автоматизации задач и организации проекта. Поставленные задачи выполнены успешно.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1) Анализ сигнала на выходе электрической цепи : методические указания к курсовой работе. – [Б. м.], 2022. – 24 с. – Текст : электронный.
2) ГОСТ 19.402-78. Единая система программной документации. Описание программы : [Текст]. – Введ. 1980-01-01. – Москва : Изд-во стандартов, 1979. – 6 с.
3) ГОСТ 19.701-90 (ИСО 5807-85). Единая система программной документации. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения : [Текст]. – Переиздание. – Москва : Стандартинформ, 2010. – 23 с.
4) Керниган, Б. У. Язык программирования С : [Текст] / Б. У. Керниган, Д. М. Ритчи ; пер. с англ. – 2-е изд. – Москва : Вильямс, 2015. – 304 с. – ISBN 978-5-8459-1975-5.
5) Конспекты лекций по программированию : [Электронный ресурс]. – URL: http://docs.basted.ru/ (дата обращения: 14.04.2025). – Текст : электронный.