Университет ИТМО

Вычислительная математика

Лабораторная работа №6

Численное решение обыкновенных дифференциальных уравнений

ФИО студента: Готовко Алексей Владимирович Направление подготовки: 09.03.04 (СППО) Учебная группа: Р32101 ФИО преподавателей: Малышева Т.А. Рыбаков С.Д.

1 Цель работы

Решить задачу интерполяции, найти значения функции при заданных значениях аргумента, отличных от узловых точек.

2 Программная реализация задачи

```
def step_euler(f: Function, h: float, x: float, y: float, twice=False) -> float:
        for i in range(2 if twice else 1):
2
            y += h * f.value_at(x, y)
3
            x += h
        return y
5
   def step_runge_kutta(f: Function, h: float, x: float, y: float, twice=False) -> float:
        for i in range(2 if twice else 1):
9
            k1 = h * f.value_at(x, y)
10
            k2 = h * f.value_at(x + h / 2, y + k1 / 2)
11
            k3 = h * f.value_at(x + h / 2, y + k2 / 2)
12
            k4 = h * f.value_at(x + h, y + k3)
13
            y += (k1 + 2 * k2 + 2 * k3 + k4) / 6
            x += h
15
        return y
16
18
   def milne_predict(f: Function, h: float, x_arr: list[float], y_arr: list[float]) -> float:
19
        return 4 * h / 3 * (2 * f.value_at(x_arr[-1], y_arr[-1]) - f.value_at(x_arr[-2], y_arr[-2])
20
                             + 2 * f.value_at(x_arr[-3], y_arr[-3])) + y_arr[-4]
21
23
   def milne_correct(f: Function,
24
                      h: float,
25
                      x_arr: list[float],
26
                      y_arr: list[float],
27
                      pred_y: float) -> float:
28
        return h / 3 * (f.value_at(x_arr[-1] + h / 2, pred_y) + 4 * f.value_at(x_arr[-1], y_arr[-1])
29
                        + f.value_at(x_arr[-2], y_arr[-2])) + y_arr[-2]
30
31
32
    class DiffEqSolver:
33
        def __init__(self,
34
                     functions: tuple [Function, Function, Function],
35
                     h: float,
36
                     y0: float,
37
                     x0: float,
38
                     xn: float,
39
                     eps: float) -> None:
40
            self.functions = functions
            self.h = h
42
            self.y0 = y0
43
            self.x0 = x0
44
            self.xn = xn
45
            self.eps = eps
46
47
        def solve(self, func_idx: int, method_name: SolutionMethod) -> Solution:
            if method_name == SolutionMethod.MILNE:
49
                return self._solve_milne(func_idx)
50
            params = get_method_params(method_name)
51
            return self._solve_single_step(func_idx, params["order"], params["function"])
53
```

```
def _solve_single_step(self, func_idx: int, order: int, step_method) -> Solution:
             f: Function
55
             f = self.functions[func_idx]
56
             cur_h = self.h
58
59
             x = self.x0
60
             y = self.y0
62
             exact_y_arr = [y]
63
             x_arr = [x]
64
             y_{arr} = [y]
65
             f_arr = []
66
67
             while x <= self.xn:
68
                 y_halved_step = step_method(f, cur_h / 2, x, y, True)
70
                 f_val = f.value_at(x, y)
71
                 y = step_method(f, cur_h, x, y)
72
73
                 x += cur_h
74
                 if abs(y - y_halved_step) / (2 ** order - 1) > self.eps and cur_h > STEP_MIN:
75
                      exact_y_arr = [f.exact_value_at(self.x0)]
76
                     x_arr = [self.x0]
                     y_{arr} = [self.y0]
78
                     f_arr = []
79
                      y = self.y0
                     x = self.x0
81
                      cur_h /= 2
82
                      continue
83
                 x_arr.append(x)
85
                 y_arr.append(y)
86
                 f_arr.append(f_val)
87
                 exact_y_arr.append(f.exact_value_at(x))
89
             f_arr.append(0)
90
             return Solution(len(exact_y_arr), x_arr, y_arr, exact_y_arr, f_arr)
91
92
         def _solve_milne(self, func_idx: int) -> Solution:
93
             f: Function
94
             f = self.functions[func_idx]
95
             x = self.x0
97
             y = self.y0
98
99
             exact_y_arr = [f.exact_value_at(x)]
100
             x_{arr} = [x]
101
             y_{arr} = [y]
102
             for i in range(3):
104
                 k1 = self.h * f.value_at(x, y)
105
                 k2 = self.h * f.value_at(x + self.h / 2, y + k1 / 2)
106
                 k3 = self.h * f.value_at(x + self.h / 2, y + k2 / 2)
107
                 k4 = self.h * f.value_at(x + self.h, y + k3)
108
109
                 y += (k1 + 2 * k2 + 2 * k3 + k4) / 6
110
                 x += self.h
                 exact_y_arr.append(f.exact_value_at(x))
112
                 y_arr.append(y)
113
                 x_arr.append(x)
114
```

```
115
             while x <= self.xn:
116
                 eps = max([abs(y_arr[i] - exact_y_arr[i]) for i in range(len(y_arr))])
117
118
                 pred_y = milne_predict(f, self.h, x_arr, y_arr)
119
                 corr_y = milne_correct(f, self.h, x_arr, y_arr, pred_y)
120
                 while abs(pred_y - corr_y) > eps > STEP_MIN:
121
                     pred_y = corr_y
122
                     corr_y = milne_correct(f, self.h, x_arr, y_arr, pred_y)
123
124
                 x += self.h
^{125}
                 exact_y_arr.append(f.exact_value_at(x))
126
                 y_arr.append(corr_y)
127
                 x_arr.append(x)
128
^{129}
             return Solution(len(exact_y_arr), x_arr, y_arr, exact_y_arr)
130
131
132
```

Полный код программы доступен по ссылке.

3 Результат работы программы

```
Select function
   [1] y' = x - y
   [2] y' = y + (1 + x) * y^2
   [3] y' = x^2 - 2 * y
   [4] y' = 3 * x^2 * y + x^2 * e^(x^3)
6
   Specify [x0, xn] interval and y0 = y(x0) value (format: 'y0 x0 xn'):
   >> -1 1 7
   Specify step (h) value:
   >> 0.1
10
   Specify accuracy (epsilon) value:
11
   >> 0.2
12
13
   Euler's method
14
15
                                   \mid f(x, y) \mid Exact y \mid
                        У
16
             __|___|
17
         0
              | 1.00000 | -1.00000 | 3.00000 | -1.00000
18
         1
              1.10000
                         | -0.70000 | 2.61000 | -0.71841
19
         2
              1.20000
                         | -0.43900
                                    2.31800
                                               | -0.46790
                         | -0.20720
                                     1 2.10440
              1.30000
                                                1 -0.24101
21
         4
              1.40000
                         0.00324
                                     1.95352
                                                | -0.03166
   Т
22
                                     1.85282
         5
              1.50000
                         0.19859
                                                  0.16515
23
   Т
                                                6
              1.60000
                          0.38387
                                     1.79225
                                                   0.35351
24
         7
                 1.70000
                          -
                            0.56310
                                     1
                                       1.76380
                                                   0.53675
25
         8
                 1.80000
                            0.73948
                                     1.76104
                                                1
              1
                         - 1
                                                   0.71763
26
         9
   1
              1
                 1.90000
                         - 1
                            0.91558
                                    1.77883
                                                0.89838
27
        10
                 2.00000
                         1.09347
                                    1.81307
                                                  1.08083
28
   Ι
        11
              2.10000
                         1.27477
                                     1.86045
                                                   1.26650
29
   Ι
        12
              1
                 2.20000
                         1.46082
                                    1.91836
                                                   1.45660
30
                 2.30000
   ١
        13
              1.65265
                                     1.98469
                                                   1.65216
31
        14
                 2.40000
                          1.85112
                                     2.05775
                                                   1.85399
32
   1
        15
              2.50000
                          2.05690
                                     2.13620
                                                1
                                                   2.06277
33
        16
                                     2.21896
   1
              2.60000
                         - 1
                            2.27052
                                                2.27905
34
   Т
        17
              -
                 2.70000
                         - 1
                            2.49242
                                     2.30517
                                                1
                                                   2.50328
        18
                 2.80000
                          2.72293
                                     2.39414
                                                  2.73585
36
        19
                 2.90000
                         2.96235
                                     2.48531
                                                2.97704
              1
37
        20
                 3.00000
                            3.21088
                                    2.57825
                                                   3.22711
                                                38
   П
              -
                         - 1
                                     2.67260
39
   ı
        21
              3.10000
                          3.46870
                                                   3.48626
   1
        22
              3.20000
                          1
                            3.73596
                                     - 1
                                       2.76808
                                                   3.75465
40
   1
        23
              3.30000
                         4.01277
                                     2.86446
                                                4.03244
41
        24
                 3.40000
                            4.29922
                                    2.96157
   1
              - 1
                         - 1
                                                  4.31971
42
        25
   1
                 3.50000
                         4.59537
                                    3.05926
                                               4.61658
43
        26
                 3.60000
                            4.90130
                                    3.15740
                                                   4.92310
44
   1
        27
                 3.70000
                         5.21704
                                     3.25592
                                                - 1
                                                   5.23935
              1
45
   Ι
        28
              1
                 3.80000
                          -
                            5.54263
                                     3.35474
                                                   5.56538
46
        29
              3.90000
                          5.87810
                                     3.45379
                                                5.90122
47
        30
                          6.22348
                                     3.55303
                 4.00000
                                                   6.24690
48
        31
                            6.57879
                                     1
                                        3.65243
                                                   6.60246
              1
                 4.10000
                          -
49
                                                -
        32
                 4.20000
                          6.94403
                                     3.75194
                                                   6.96792
        33
                 4.30000
                            7.31922
                                     3.85155
                                                  7.34330
51
                 4.40000
                          7.70438
                                     3.95124
                                                  7.72861
52
        35
                 4.50000
                            8.09950
                                     4.05099
                                                8.12386
   1
              -
53
        36
                 4.60000
                            8.50460
                                     4.15080
   ı
              1
                         -
                                                8.52907
54
        37
              4.70000
                          8.91968
                                     4.25064
                                                   8.94424
55
        38
   1
              4.80000
                          9.34475
                                     4.35051
                                                   9.36937
56
   1
        39
              4.90000
                         9.77980
                                     1 4.45041
                                                1 9.80449
57
```

5.00000 | 10.22484 | 4.55033

1

40

10.24958

59	41		5.10000	-	10.67987	-	4.65026	- [10.70466	- [
60	l 42	-	5.20000	-	11.14490	1	4.75021	-	11.16972	-
61	l 43	-	5.30000	-	11.61992	1	4.85017	-	11.64477	-
62	44	-	5.40000	-	12.10493	1	4.95013	-	12.12981	-
63	l 45		5.50000	-	12.59995	-	5.05011	- [12.62485	- [
64	l 46		5.60000	-	13.10496	-	5.15009	- [13.12987	- [
65	47		5.70000	-	13.61997	-	5.25007	- [13.64490	- [
66	l 48		5.80000	-	14.14497	-	5.35005	- [14.16992	- [
67	l 49		5.90000	-	14.67998	-	5.45004	- [14.70493	- [
68	J 50		6.00000	-	15.22498	-	5.55003	- [15.24994	- [
69	51		6.10000	-	15.77999	-	5.65003	- [15.80495	- [
70	J 52		6.20000	-	16.34499	-	5.75002	- [16.36996	- [
71	J 53		6.30000	-	16.91999	-	5.85002	- [16.94497	- [
72	54		6.40000	- 1	17.50499	- 1	5.95001		17.52997	
73	J 55		6.50000		18.09999	-	6.05001		18.12498	
74	l 56		6.60000		18.70500	-	6.15001		18.72998	
75	J 57		6.70000	-	19.32000	-	6.25001	- [19.34499	- [
76	J 58		6.80000	- 1	19.94500	- 1	6.35001		19.96999	
77	l 59		6.90000		20.58000	-	6.45000		20.60499	
78	l 60		7.00000	-	21.22500	-	6.55000	- [21.24999	- [
79	l 61	- 1	7.10000		21.88000	-	0.00000	- [21.90499	- [
80	I			_1.		_		_1_		_1

${\tt Runge-Kutta's\ method}$

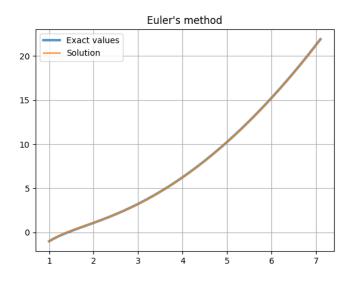
84											
85 86	 	i	 	x	 	У	l I	f(x, y)	 	Exact y	
87	i	0	 I	1.00000	' .	-1.00000	- ' - · 	3.00000	- · ·	-1.00000	- :
88	i	1	İ	1.10000		-0.71842	i	2.64683	i	-0.71841	i
89	i	2	İ	1.20000		-0.46790	i	2.37581	i	-0.46790	i
90	i	3	İ	1.30000		-0.24102	i	2.17204	i	-0.24101	i
91	Ì	4	İ	1.40000		-0.03167	İ	2.02333	ĺ	-0.03166	İ
92	1	5	l	1.50000		0.16515	-	1.91971	- 1	0.16515	1
93	1	6	l	1.60000		0.35350	-	1.85299	- 1	0.35351	1
94	1	7	l	1.70000		0.53675		1.81650	- [0.53675	1
95	1	8	l	1.80000		0.71763		1.80475	-	0.71763	1
96	1	9		1.90000		0.89837		1.81325	-	0.89838	-
97	1	10		2.00000		1.08083		1.83834	- 1	1.08083	-
98	1	11		2.10000		1.26650		1.87701	- 1	1.26650	1
99	I	12		2.20000		1.45660		1.92680	- [1.45660	1
100	1	13		2.30000		1.65216		1.98568		1.65216	-
101	1	14		2.40000		1.85399		2.05202		1.85399	-
102	1	15		2.50000		2.06277		2.12446		2.06277	-
103	1	16	l	2.60000		2.27905		2.20190	- 1	2.27905	-
104	1	17	1	2.70000		2.50329		2.28343	- 1	2.50328	1
105	1	18	1	2.80000		2.73585		2.36830	- 1	2.73585	1
106	1	19	1	2.90000		2.97704		2.45592	- 1	2.97704	1
107	1	20	1	3.00000		3.22711		2.54578	- 1	3.22711	1
108	1	21	1	3.10000		3.48626		2.63748	- 1	3.48626	1
109	I	22		3.20000		3.75466		2.73069		3.75465	-
110	I	23		3.30000		4.03244		2.82512	-	4.03244	I
111	I	24	l	3.40000		4.31972		2.92057	- 1	4.31971	I
112	I	25		3.50000		4.61658		3.01684	-	4.61658	I
113	I	26		3.60000		4.92311		3.11378	- 1	4.92310	I
114	I	27		3.70000		5.23936		3.21128	- 1	5.23935	I
115	I	28		3.80000		5.56538		3.30924	- 1	5.56538	I
116	I	29	l	3.90000		5.90122		3.40756	I	5.90122	I
117	1	30	1	4.00000		6.24691	-	3.50619	-	6.24690	-
118	Į.	31		4.10000		6.60247		3.60506		6.60246	
119	1	32	İ	4.20000	l	6.96793	ı	3.70415	I	6.96792	ı

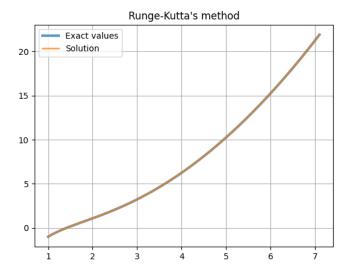
120												
122	120		33		4.30000	-	7.34330	- 1	3.80339		7.34330	
123	121	-	34		4.40000	-	7.72861	- 1	3.90278		7.72861	
124	122	-	35		4.50000	-	8.12386	- 1	4.00227		8.12386	
125	123	-	36		4.60000	-	8.52907	- 1	4.10186		8.52907	
126	124	-	37		4.70000	-	8.94424	- 1	4.20152		8.94424	
127	125	-	38		4.80000	-	9.36938	- 1	4.30124		9.36937	
128	126	-	39		4.90000	-	9.80449	- 1	4.40102		9.80449	
129	127	-	40		5.00000	-	10.24959	- 1	4.50083		10.24958	
130 43 5.30000 11.64477 4.80045 11.64477 131 44 5.40000 12.12982 4.90037 12.12981 132 45 5.50000 12.62485 5.00030 12.62485 133 46 5.60000 13.12988 5.10024 13.12987 134 47 5.70000 13.64490 5.20020 13.64490 135 48 5.80000 14.16992 5.30016 14.16992 136 49 5.90000 14.70494 5.40013 14.70493 137 50 6.00000 15.24995 5.50010 15.24994 138 51 6.10000 15.80496 5.60008 15.80495 139 52 6.20000 16.36997 5.70007 16.36996 140 53 6.30000 16.94497 5.80005 16.94497 141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000 18.12498 6.00003 18.12498 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500	128	1	41	- 1	5.10000	-	10.70466	- [4.60068		10.70466	
131 44 5.40000 12.12982 4.90037 12.12981 132 45 5.50000 12.62485 5.00030 12.62485 133 46 5.60000 13.12988 5.10024 13.12987 134 47 5.70000 13.64490 5.20020 13.64490 135 48 5.80000 14.16992 5.30016 14.16992 136 49 5.90000 14.70494 5.40013 14.70493 137 50 6.00000 15.24995 5.50010 15.24994 138 51 6.10000 15.80496 5.60008 15.80495 139 52 6.20000 16.36997 5.70007 16.36996 140 53 6.30000 17.52998 5.90004 17.52997 141 54 6.40000 17.52998 5.90004	129	-	42		5.20000	-	11.16972	- 1	4.70055		11.16972	
132 45 5.50000 12.62485 5.00030 12.62485 133 46 5.60000 13.12988 5.10024 13.12987 134 47 5.70000 13.64490 5.20020 13.64490 135 48 5.80000 14.16992 5.30016 14.16992 136 49 5.90000 14.70494 5.40013 14.70493 137 50 6.00000 15.24995 5.50010 15.24994 138 51 6.10000 15.80496 5.60008 15.80495 139 52 6.20000 16.36997 5.70007 16.36996 140 53 6.30000 16.94497 5.80005 16.94497 141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000 18.72999 6.10002 18.72998 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.90499	130	1	43	- 1	5.30000	-	11.64477	- [4.80045		11.64477	
133 46 5.60000 13.12988 5.10024 13.12987 13.4 47 5.70000 13.64490 5.20020 13.64490 14.6490 14.6992 13.64490 14.64992 13.64490 14.64992 13.64490 14.64992 13.64490 14.70493	131	1	44	- 1	5.40000	-	12.12982	- [4.90037		12.12981	
134 47 5.70000 13.64490 5.20020 13.64490 135 48 5.80000 14.16992 5.30016 14.16992 136 49 5.90000 14.70494 5.40013 14.70493 137 50 6.00000 15.24995 5.50010 15.24994 138 51 6.10000 15.80496 5.60008 15.80495 139 52 6.20000 16.36997 5.70007 16.36996 140 53 6.30000 16.94497 5.80005 16.94497 141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000 18.12498 6.00003 18.12498 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002	132	1	45	- 1	5.50000	-	12.62485	- [5.00030		12.62485	
135 48 5.80000 14.16992 5.30016 14.16992 136 49 5.90000 14.70494 5.40013 14.70493 137 50 6.00000 15.24995 5.50010 15.24994 138 51 6.10000 15.80496 5.60008 15.80495 139 52 6.20000 16.36997 5.70007 16.36996 140 53 6.30000 16.94497 5.80005 16.94497 141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000	133	-	46		5.60000	-	13.12988	- 1	5.10024		13.12987	
136 49 5.90000 14.70494 5.40013 14.70493 137 50 6.00000 15.24995 5.50010 15.24994 138 51 6.10000 15.80496 5.60008 15.80495 139 52 6.20000 16.36997 5.70007 16.36996 140 53 6.30000 16.94497 5.80005 16.94497 141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000 18.12498 6.00003 18.12498 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001	134	-	47		5.70000	-	13.64490	- 1	5.20020		13.64490	
137 50 6.00000 15.24995 5.50010 15.24994 1 138 51 6.10000 15.80496 5.60008 15.80495 1 139 52 6.20000 16.36997 5.70007 16.36996 1 140 53 6.30000 16.94497 5.80005 16.94497 1 141 54 6.40000 17.52998 5.90004 17.52997 1 142 55 6.50000 18.12498 6.00003 18.12498 1 143 56 6.60000 18.72999 6.10002 18.72998 1 144 57 6.70000 19.34499 6.20002 19.34499 1 145 58 6.80000 19.96999 6.30001 19.96999 1 146 59 6.90000 20.60500 6.40001 20.60499 1 148 61 7.10000 21.25000 6.50001 21.90499 1	135	-	48		5.80000	-	14.16992	- 1	5.30016		14.16992	
138 51 6.10000 15.80496 5.60008 15.80495 139 52 6.20000 16.36997 5.70007 16.36996 140 53 6.30000 16.94497 5.80005 16.94497 141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000 18.12498 6.00003 18.12498 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000	136	1	49	- 1	5.90000	-	14.70494	- [5.40013		14.70493	
139 52 6.20000 16.36997 5.70007 16.36996 140 53 6.30000 16.94497 5.80005 16.94497 141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000 18.12498 6.00003 18.12498 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000	137	1	50	- 1	6.00000	-	15.24995	- [5.50010		15.24994	
140 53 6.30000 16.94497 5.80005 16.94497 141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000 18.12498 6.00003 18.12498 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000 21.90499	138	1	51	- 1	6.10000	-	15.80496	- [5.60008		15.80495	
141 54 6.40000 17.52998 5.90004 17.52997 142 55 6.50000 18.12498 6.00003 18.12498 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000 21.90499	139	1	52	- 1	6.20000	-	16.36997	- [5.70007		16.36996	
142 55 6.50000 18.12498 6.00003 18.12498 143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000 21.90499	140	1	53	- 1	6.30000	-	16.94497	- [5.80005		16.94497	
143 56 6.60000 18.72999 6.10002 18.72998 144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000 21.90499	141	1	54	- 1	6.40000	-	17.52998	- [5.90004		17.52997	
144 57 6.70000 19.34499 6.20002 19.34499 145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000 21.90499	142	-	55		6.50000	-	18.12498	- 1	6.00003		18.12498	
145 58 6.80000 19.96999 6.30001 19.96999 146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000 21.90499	143	-	56		6.60000	-	18.72999	- 1	6.10002		18.72998	
146 59 6.90000 20.60500 6.40001 20.60499 147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000 21.90499	144	1	57	- 1	6.70000	-	19.34499	- [6.20002		19.34499	
147 60 7.00000 21.25000 6.50001 21.24999 148 61 7.10000 21.90500 0.00000 21.90499	145	1	58	- 1	6.80000	-	19.96999	- [6.30001		19.96999	
148 61 7.10000 21.90500 0.00000 21.90499	146	1	59	- 1	6.90000	-	20.60500	- [6.40001		20.60499	
	147		60	- 1	7.00000	- [21.25000	- 1	6.50001	-	21.24999	-
149	148	1	61	- 1	7.10000	-	21.90500	- [0.0000	- [21.90499	
	149	1		1_		_1		1.		_1.		_

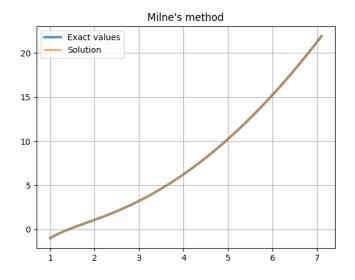
Milne's method

153				
154	l i	l x	I у	Exact y
155	l	l	l	
156	1 0	1.00000	-1.00000	-1.00000
157	1	1.10000	-0.71842	-0.71841
158	1 2	1.20000	-0.46790	-0.46790
159	3	1.30000	-0.24102	-0.24101
160	4	1.40000	-0.03624	-0.03166
161	5	1.50000	0.16171	0.16515
162	1 6	1.60000	0.34548	0.35351
163	7	1.70000	0.53057	0.53675
164	8	1.80000	0.70665	0.71763
165	9	1.90000	0.88991	0.89838
166	10	2.00000	1.06721	1.08083
167	11	2.10000	1.25608	1.26650
168	12	2.20000	1.43973	1.45660
169	13	2.30000	1.64023	1.65216
170	14	1 2.40000	1.83391	1.85399
171	15	2.50000	2.04970	2.06277
172	16	2.60000	2.25581	2.27905
173	17	2.70000	2.48785	2.50328
174	18	2.80000	2.70976	2.73585
175	19	1 2.90000	2.96119	2.97704
176	1 20	3.00000	3.19797	3.22711
177	21	3.10000	3.46833	3.48626
178	22	3.20000	3.72271	3.75465
179	l 23	3.30000	4.01251	4.03244
180	24	3.40000	4.28513	4.31971

181	l 25	I 3.50000	l 4.59480	l 4.61658 l
182		3.60000	1 4.88598	l 4.92310 l
183	1 27	1 3.70000	5.21585	l 5.23935 l
184	28	3.80000	5.52577	5.56538
185	29	3.90000	5.87606	5.90122
186	30	4.00000	6.20484	6.24690
187	31	4.10000	6.57573	6.60246
188	32	4.20000	6.92342	6.96792
189	33	4.30000	7.31504	7.34330
190	l 34	4.40000	7.68166	7.72861
191	35	4.50000	8.09412	8.12386
192	l 36	4.60000	8.47967	8.52907
193	37	4.70000	8.91305	8.94424
194	38	4.80000	9.31751	9.36937
195	39	4.90000	9.77190	9.80449
196	l 40	5.00000	10.19523	10.24958
197	l 41	5.10000	10.67070	10.70466
198	42	5.20000	11.11286	11.16972
199	43	5.30000	11.60948	11.64477
200	44	5.40000	12.07041	12.12981
201	45	5.50000	12.58826	12.62485
202	l 46	5.60000	13.06789	13.12987
203	47	5.70000	13.60705	13.64490
204	48	5.80000	14.10533	14.16992
205	l 49	5.90000	14.66586	14.70493
206	50	6.00000	15.18271	15.24994
207	51	6.10000	15.76469	15.80495
208	52	6.20000	16.30004	16.36996
209	53	6.30000	16.90357	16.94497
210	54	6.40000	17.45733	17.52997
211	55	6.50000	18.08248	18.12498
212	56	6.60000		18.72998
213	57	6.70000		19.34499
214		6.80000		19.96999
215	59	6.90000	•	20.60499
216		7.00000		21.24999
217	61	7.10000	21.85948	21.90499
218	I		_	Il
219				







4 Вывод

Для наиболее оптимального распределения человеческих ресурсов и поддержания количества нервных клеток в головном мозгу, настоятельно рекомендуется использовать готовые библиотеки, содержащие наиболее эффективные реализации алгоритмов, вместо самостоятельной реализации оных.

Иными словами, вообще лучше зачиллиться и не изобретать велосипед.