Лабораторная работа 2

Структуры данных

Ланцова Я. И.

Российский университет дружбы народов, Москва, Россия

Информация

Докладчик

- Ланцова Яна Игоревна
- студентка
- Российский университет дружбы народов

Цель работы

Основная цель работы – изучить несколько структур данных, реализованных в Julia, научиться применять их и операции над ними для решения задач.

Задание

1. Используя Jupyter Lab, повторите примеры.
2. Выполните задания для самостоятельной работы.

Выполнение лабораторной работы

Выполнение лабораторной работы

```
[1]: favoritelang = ("Python","Julia","R", "C++")

[1]: ("Python", "Julia", "R", "C++")

[2]: tuple_1 = (1, 2.0, "tmp", 53)

[2]: (1, 2.0, "tmp", 53)

[3]: tuple_2 = (a=2, b=1+2)

[3]: (a = 2, b = 3)

[5]: in("tmp", tuple_1), 0 in tuple_1

[5]: (true, false)

[6]: phonebook = Dict("Иванов И.И." => ("867-5309","333-5544"), "Бухгалтерия" => "555-2368")

[6]: Dict{String, Any} with 2 entries:
       "Бухгалтерия" => "555-2368"
       "Иванов И.И." => ("867-5309", "333-5544")

[7]: keys(phonebook)

[7]: KeySet for a Dict{String, Any} with 2 entries. Keys:
       "Бухгалтерия"
       "Иванов И.И."

[8]: values(phonebook)

[8]: ValueIterator for a Dict{String, Any} with 2 entries. Values:
       "555-2368"
       ("867-5309", "333-5544")

[9]: haskey(phonebook, "Иванов И.И.")

[9]: true
```

Рис. 1: Примеры использования кортежей и словарей

Выполнение лабораторной работы

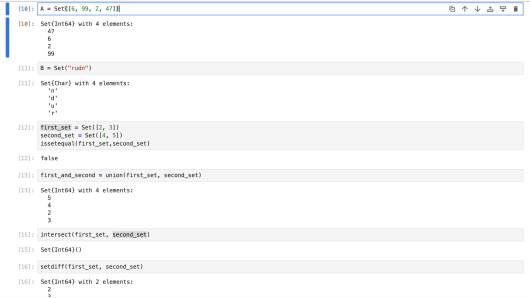


Рис. 2: Примеры использования множеств

Выполнение лабораторной работы

```
[18]: a = [1, 0, 99]
      b = [4, 70, 8]

[18]: 3-element Vector{Int64}:
       4
      70
       8

[19]: c = rand(1,8)

[19]: 1×8 Matrix{Float64}:
       0.4598776  0.475469  0.785701  0.00159815  …  0.311825  0.124381  0.263145

[21]: ones(5)

[21]: 5-element Vector{Float64}:
       1.0
       1.0
       1.0
       1.0
       1.0

[23]: c = transpose(b)

[23]: 1×3 transpose(::Vector{Int64}) with eltype Int64:
       4  70  8

[25]: ar = rand(10:20, 2, 5)

[25]: 2×5 Matrix{Int64}:
       13  19  20  12  16
       12  10  12  13  11

[26]: sort(ar,dims=1)

[26]: 2×5 Matrix{Int64}:
       12  10  12  12  11
       13  19  20  13  16
```

Рис. 3: Примеры использования массивов

Задание 1

Выполнение лабораторной работы

Рис. 4: Работа с множествами

Задание 2

Выполнение лабораторной работы

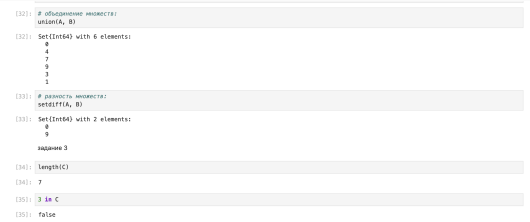


Рис. 5: Примеры операций над множествами элементов разных типов

Задание 3

Выполнение лабораторной работы

```
[37]: arr1 = [i for i in 1:30];

[40]: arr1_1 = [1:30...];

[42]: println(arr1, "\n", arr1_1)
      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]
      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]

[44]: arr2 = reverse(arr1);
      println(arr2)
      [30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1]

[48]: # объединим два массива
      arr3 = vcat(1:30, (30-1):-1:1);
      println(arr3)
      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 29, 28, 27, 26, 25, 24, 23,
      22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1]

[50]: tmp = [4, 6, 3]
      println(tmp)
      [4, 6, 3]

[51]: arr5 = fill(tmp[1], 10);
      println(arr5)
      [4, 4, 4, 4, 4, 4, 4, 4, 4, 4]

[52]: arr6 = fill(tmp, 10);
      println(arr6)
      [[4, 6, 3], [4, 6, 3], [4, 6, 3], [4, 6, 3], [4, 6, 3], [4, 6, 3], [4, 6, 3], [4, 6, 3], [4, 6, 3], [4, 6, 3]]
```

Рис. 6: Работа с массивами по заданиям 3.1-3.6

```
[54]: arr7 = vcat(fill(tmp[1], 11), fill(tmp[2], 10), fill(tmp[3], 10));
      println(arr7)

      [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3]

[55]: arr8 = vcat(fill(tmp[1], 10), fill(tmp[2], 20), fill(tmp[3], 30));
      println(arr8)

      [4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
      3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3]

[62]: elems = [2^tmp[1], 2^tmp[2]];
      arr9 = vcat(elems, fill(2^tmp[3], 4));

      # объединим числа в строку

      arr_str9 = join(arr9)
      count_6 = count(c -> c == '6', arr_str9)

      println(arr9, "\n", count_6)

      [16, 64, 8, 8, 8, 8]
      2

[71]: vec10 = [exp(x) * cos(x) for x in 3:0.1:6];
      println(vec10, "\n", mean(vec10))

      [-19.884530844146987, -22.178753389342127, -24.490696732801293, -26.77318244299338, -28.969237768093574, -31.011186439374516, -32.8197747603
      38504, -34.30336011037369, -35.35719361853035, -35.86283371230767, -35.68773248011913, -34.68504225166807, -32.693695428321746, -29.53881629
      7262983, -25.032529229039966, -18.975233154958957, -11.157417389647478, -1.3620985182057503, 10.632038010191998, 25.046704998273004, 42.0992
      0106253839, 61.99663027669454, 84.92906736250268, 111.0615860420258, 140.5250750527875, 173.40577640857734, 209.73349424783467, 249.46844055
      885668, 292.4867067371223, 338.5643778585117, 387.36034029093076]
      53.11374594642971
```

Рис. 7: Работа с массивами по заданиям 3.7-3.10

Выполнение лабораторной работы

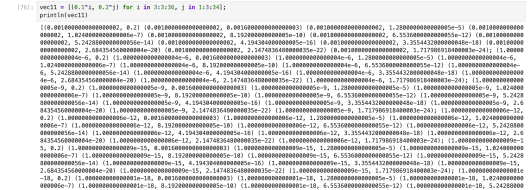
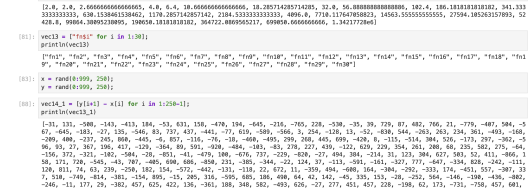


Рис. 8: Работа с массивами по заданию 3.11

Выполнение лабораторной работы



```
[78]: vec12 = [2^i / i for i in 1:25];
      println(vec12)

      [2.0, 2.0, 2.6666666666666665, 4.0, 6.4, 10.666666666666666, 18.285714285714285, 32.0, 56.888888888888886, 102.4, 186.18181818181182, 341.333
      3333333333, 630.1538461538462, 1170.2857142857142, 2184.5333333333333, 4096.0, 7710.117647058823, 14563.555555555555, 27594.105263157893, 52
      428.8, 99864.38095238095, 190650.18181818182, 364722.0869565217, 699050.6666666666, 1.3421772Be6]

[81]: vec13 = ["fn$i" for i in 1:30];
      println(vec13)

      ["fn1", "fn2", "fn3", "fn4", "fn5", "fn6", "fn7", "fn8", "fn9", "fn10", "fn11", "fn12", "fn13", "fn14", "fn15", "fn16", "fn17", "fn18", "fn1
      9", "fn20", "fn21", "fn22", "fn23", "fn24", "fn25", "fn26", "fn27", "fn28", "fn29", "fn30"]

[83]: x = rand(0:999, 250);
      y = rand(0:999, 250);

[88]: vec14_1 = [y[i+1] - x[i] for i in 1:250-1];
      println(vec13_1)

      [-31, 131, -508, -143, -413, 184, -53, 631, 150, -470, 194, -645, -216, -765, 228, -530, -35, 39, 729, 87, 482, 766, 21, -779, -487, 504, -5
      67, -645, -183, -27, 135, -546, 83, 737, 437, -441, -77, 619, -589, -566, 3, 254, -126, 13, -52, -830, 544, -263, 263, 234, 361, -493, -168,
      -209, 480, -237, 245, 808, -445, -6, 857, -115, -76, -39, -408, -495, 299, 288, 465, 699, -420, 8, -115, -514, 304, 526, -173, 297, -302, -5
      96, 93, 27, 367, 196, 417, -129, -364, 89, 591, -920, -484, -103, -83, 278, 227, 439, -122, 629, 229, 354, 261, 280, 68, 235, 582, 275, -64,
      -156, 372, -321, -102, -504, -20, -851, -41, -479, 100, -670, 737, -229, -820, -27, 494, 304, -214, 31, 123, 304, 627, 583, 52, 411, -866, 1
      58, 171, 720, -545, -43, 707, -485, 690, 686, -850, 231, -385, -344, -22, 124, 37, -113, -591, -161, -327, 777, -647, -334, 828, -242, -111,
      120, 811, 74, 63, 239, -250, 182, 154, -572, -442, -331, -110, 22, 672, 11, -359, 494, -606, 164, -304, -292, -333, 174, -651, 557, -307, 33
      7, 510, -749, -814, -381, -154, 895, -15, 205, 316, -595, 685, 186, 490, 64, 42, 142, -45, 335, 153, -28, -252, 564, -146, -198, -436, -802,
      -246, -11, 177, 29, -382, 457, 625, 422, 136, -361, 189, 348, 582, -893, 626, -27, 277, 451, 457, 228, -198, 62, 173, -731, -750, 457, 681,
      709, -92, 278, 68, -23, -205, 651, 397, -116, 3]

[89]: vec14_2 = [x[i] + 2*x[i+1] - x[i+2] for i in 1:250-2];
      println(vec13_2)

      [1024, 2134, 799, 1168, -186, 1344, 156, 985, 1626, 890, 1552, 885, 1718, 1926, 1711, 935, 589, -153, 856, 399, -436, 1030, 1637, 2513, 174,
      651, 1584, 2161, 509, -90, 1278, 1770, 627, 369, 913, 2405, 488, 1069, 1823, 1752, 651, 1376, 2019, 1398, 2103, 934, 1182, 611, -164, 670, 1
      417, 1929, 837, 944, 1298, 1416, -201, 1479, 1012, 222, 932, 440, -47, 1571, 2003, 1647, 568, 939, 344, 1139, 1283, 980, 2251, 1353, 779, 12
      38, 1315, 1523, 1964, 1314, 2116, 1591, 204, 230, 381, 975, 424, -836, 1326, 1481, 1774, 377, 910, 874, 872, 1148, 332, 476, 431, 519, 242,
      103, 1121, 302, 344, 390, 1963, 680, 1247, 693, 1103, 3625, 2476, 627, 1024, 944, 2324, 700, 41, 1274, 2185, 942, -419, 1385, 1429, 2080, 11
      80, 174, -329, 1187, 510, 1865, 681, 1802, 113, 426, 2349, 347, 1850, 1040, -535, 1284, 1510, 2083, 1784, 20, 1144, 542, 241, 1444, 758, 235
      6, 197, 1083, 2692, 730, 817, 569, 1395, 449, 271, 110, 116, 892, 446, 507, 1236, 2064, 481, 789, 1745, 825, -140, 1634, 340, 1370, 988, 127
```

Рис. 9: Работа с массивами по заданиям 3.12-3.14

```
[90]: vec14_3 = [sin(y[i])/cos(x[i+1]) for i in 1:250-1];
println(vec14_3)
```

```
[0.7473890761102641, 1.3080914156295154, 4.599373499821138, -1.2400887358372137, -0.3566014197704019, -5.9087063246768645, -0.89724985355929
48, 0.8511732921191558, -2.1156099517838056, 14.10775692909659, 0.3191680436602564, -0.7668639440594296, 0.10302303143715785, -0.86817547279
84285, -1.506103499399896, 0.8824296360827248, 1.0811088877801183, 0.3021878117666545, -0.7592741917946998, 0.8558529189323758, 0.6417688328
135184, 0.9798474491088568, -0.31217595859127695, 14.008459081016229, -3.0923579084309636, 0.10702770688635906, -0.5088829453732117, -1.0065
334988144483, 0.6768157283168107, -1.0466712188519456, 1.3421572418876846, -0.6055699644024115, -2.673308178247664, 4.871633694203398, -0.95
32761346734071, -18.153110638613946, -0.5172144176703048, 0.4017151265335428, -3.8432602777803013, 0.8397707605313448, 1.6643263238120536,
4.045961741871783, -1.1260242516457577, -1.8673218284737072, -0.06198568305173133, 0.1559938296012099, -0.7327690587141732, 0.23693688073571
328, 1.2458716095464781, -0.9038382760117768, 0.8537890131682548, -0.7837639195346212, -0.8291926863172305, 1.0010685051286343, -0.859745678
5034667, 1.3135113746588682, 2.169801906609702, 8.734077425140043, -0.9804112335599723, -0.5286891783796658, 0.0383614002508579, 0.11751558
335395076, 0.6322839867314325, 0.7330497581375609, -0.30409526806440595, 0.8056016482882896, 9.760057482228735, 0.1145696930278182, 0.49017
585590942786, -0.8209364253549514, -24.777312422859488, -2.2542747966852827, -0.1294303322677087, 0.12178786649325718, -2.150231746897391,
1.031235855722456, 0.1253023628166324, 0.4534253267163959, 0.5231844383447154, 0.8845756200187418, -0.027184053889779618, 0.273195681398424
5, -6.790371568543003, 0.7955001486101527, -0.46248315548339136, -1.6795890218114278, -0.08950184411342728, 0.6323258473043197, 1.1423086624
34277, 0.38607218513534575, -0.13715914969727888, 0.9010879074552461, -1.1231607510709631, -1.7637788544788142, -0.7180477396326548, -0.1861
485416293908, -3.641890322226735, 0.42073643065214333, -1.034391538933087, 0.08941079620962127, -5.125205895004963, -1.0025712268360631, 0.8
474004715326545, 1.456258311555779, -1.2891764606309082, -1.0274735749675783, -0.6437185521307713, 0.6796179236971972, -0.5531790435228858,
-3.0825558930378225, 1.3587478501960284, -0.23509675735730173, 0.18958848897906366, -10.3464820294, -0.6180820652100812, 1.0163651232396411,
-3.531331618343402, 1.300702942567278, -1.0335506595950608, -0.23683535507284167, 0.1989061367390097, 0.0, 0.6506498112190067, 1.71111353942
22565, -0.894529632673156, 0.5880772710662155, 0.2888724089447745, 0.08904733407858888, 3.9595381788593755, -7.241383523048887, 1.8375247353
510624, -1.3774860422830657, -1.1655027868854833, 0.30105977605935325, -1.0908341239462571, -1.4584181394675446, 0.1853215460859822, 0.70029
37908303628, 0.7115708372613735, 1.7804993169575731, 0.8205905797271945, 0.8756857612452192, 3.7629216713208553, 1.8348166590628272, -0.2830
465105778692, 3.1405597107784593, 2.738007897431789, 1.0585055469032172, -0.42417144859274575, -1.1252452298521796, 0.8536123159979188, -2.5
982808126901156, -0.961802122338875 1, 1.1130207923773194, 0.8111731040684318, 0.34017181035740707, 1.290904308785016, -1.4219353136734487, -
1.2787239514881616, 1.3103303694071806, -0.4890325666837324, 0.8161652471110976, 1.3056506979033837, -0.3431248107421011, 0.507422222103284,
0.7374084839733527, -29.073476810961207, -1.0125862645507808, 2.7839175237873954, -1.051329799766767, 0.3163645068282446, 1.992065703930372
9, 0.5840888138183655, -1.2328024653451195, 1.0799596474611954, -1.2292858817398804, -2.0618347442276455, -2.212740263446289, -0.92453987677
78192, -1.0978380166919903, -35.672703186071786, -1.6544427376302204, -0.33286216457649254, -0.4183616423465848, 0.7687543185066473, 0.99654
55347984983, 1.1322884133226294, -0.887549117836715, -0.9288534901213675, 0.8095742416443351, -1.9970976837709988, -0.03876466537155388, -1.
0997607141385712, -0.1792385183034481, -0.9729923573736801, -0.40337914115347656, 0.9944390598528742, -0.4014803891858844, -0.28999338496886
28, 0.3912825146642829, -0.7989933213182223, -2.356251008887717, 0.19495615097302713, 0.13358669548556137, -0.6789647284934692, 1.4841538131
```

Рис. 10: Работа с массивами по заданию 3.14

Выполнение лабораторной работы

```
[91]: sum14_4 = sum(exp(-x[i+1]) / (x[i] + 10) for i in 1:250-1)

[91]: 1.6198145953442972e-7

[94]: y_bigger_600 = y[y .> 600];
      y_bigger_indices = findall(y .> 600);
      println(y_bigger_600, "\n", y_bigger_indices)

      [739, 778, 718, 665, 738, 934, 692, 998, 777, 936, 863, 689, 962, 866, 867, 816, 686, 696, 788, 911, 641, 826, 995, 994, 717, 972, 996, 842,
      934, 991, 619, 655, 842, 631, 983, 768, 932, 728, 646, 825, 849, 683, 888, 734, 722, 668, 632, 723, 623, 843, 681, 826, 725, 633, 728, 682,
      938, 696, 652, 748, 765, 938, 729, 880, 874, 827, 830, 971, 741, 652, 868, 602, 972, 675, 841, 655, 678, 871, 683, 653, 862, 959, 776, 972,
      815, 873, 673, 638, 635, 727, 685, 783, 839, 660, 776, 810, 860, 638, 731, 711, 843, 806, 885, 817, 769, 862, 881, 636]
      [1, 2, 3, 8, 9, 10, 12, 16, 20, 23, 24, 34, 35, 36, 38, 39, 42, 43, 44, 45, 46, 48, 52, 56, 58, 59, 62, 68, 70, 71, 73, 76, 77, 78, 79, 82,
      83, 84, 86, 90, 95, 96, 97, 99, 105, 106, 107, 109, 110, 114, 118, 120, 123, 124, 126, 127, 128, 130, 131, 132, 133, 136, 137, 139, 140, 14
      2, 143, 145, 149, 154, 155, 157, 158, 161, 162, 168, 173, 174, 177, 179, 187, 188, 192, 193, 195, 196, 198, 199, 200, 202, 204, 205, 207, 20
      8, 209, 220, 221, 223, 224, 228, 229, 232, 233, 236, 240, 241, 247, 250]

[96]: x_corres = x[y_bigger_indices];

[97]: for i in 1:length(y_bigger_indices)
          idx = y_bigger_indices[i]
          println("$idx: x=$(x[idx]), y=$(y[idx])")
      end

      1: x=809, y=739
      2: x=587, y=778
      3: x=959, y=718
      8: x=107, y=665
      9: x=776, y=738
      10: x=674, y=934
      12: x=780, y=692
      16: x=681, y=998
      20: x=437, y=777
      23: x=842, y=936
      24: x=824, y=863
      34: x=225, y=689
      35: x=429, y=962
      36: x=714, y=866
```

Рис. 11: Работа с массивами по заданию 3.14

Выполнение лабораторной работы

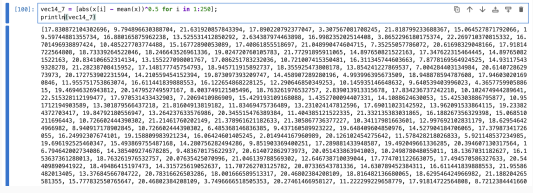


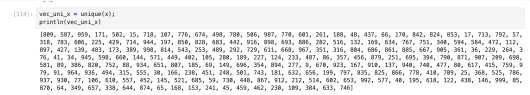
Рис. 12: Работа с массивами по заданию 3.14

Выполнение лабораторной работы

```
[103]: y_close_max = y[abs.(y .- maximum(y)) .<= 200];
       println(length(y_close_max))

       50

[107]: x_even = x[x .% 2 .== 0]
       x_odd = x[x .% 2 .== 1]
       println(length(x_even), " ",length(x_odd))

       123 127

[108]: x_seven = x[x .% 7 .== 0]
       println(length(x_seven))

       32

[109]: println(x[sortperm(y)])

       [752, 41, 598, 502, 277, 318, 967, 853, 632, 282, 209, 617, 874, 571, 525, 261, 15, 748, 610, 389, 660, 65, 338, 923, 34, 594, 80, 506, 376,
       792, 713, 45, 937, 238, 820, 698, 489, 751, 867, 577, 718, 248, 730, 498, 555, 415, 992, 195, 581, 770, 618, 188, 48, 489, 229, 122, 606, 8
       9, 91, 797, 315, 132, 29, 820, 603, 633, 944, 751, 105, 990, 871, 607, 987, 188, 790, 487, 759, 611, 181, 936, 543, 459, 979, 402, 85, 233,
       472, 64, 173, 602, 998, 394, 40, 879, 169, 644, 634, 653, 139, 501, 767, 189, 668, 410, 17, 171, 427, 783, 351, 15, 230, 934, 451, 124, 69,
       885, 137, 304, 109, 656, 349, 713, 489, 866, 168, 66, 477, 462, 57, 814, 227, 178, 967, 930, 144, 778, 105, 360, 340, 709, 685, 824, 146, 44
       9, 668, 695, 945, 686, 456, 88, 521, 746, 145, 543, 886, 264, 696, 91, 825, 316, 743, 212, 107, 357, 30, 199, 910, 807, 835, 448, 442, 225,
       788, 916, 149, 870, 112, 959, 651, 86, 251, 139, 59, 36, 670, 112, 280, 776, 809, 170, 354, 905, 894, 153, 77, 514, 437, 507, 25, 898, 657,
       438, 557, 850, 790, 516, 386, 144, 940, 312, 166, 253, 884, 907, 89, 376, 146, 449, 241, 824, 714, 197, 964, 229, 886, 452, 910, 167, 109, 3
       18, 402, 693, 185, 361, 674, 292, 842, 9, 786, 429, 910, 897, 494, 106, 861, 729, 584, 767, 483, 6011

[110]: reverse(last(sort(x), 10))

[110]: 10-element Vector{Int64}:
        999
        998
        992
        987
        979
        967
        967
        964
        959
```

Рис. 13: Работа с массивами по заданию 3.14

Выполнение лабораторной работы



```
[114]: vec_uni_x = unique(x);
       println(vec_uni_x)
```

```
[809, 587, 959, 171, 502, 15, 718, 107, 776, 674, 498, 700, 506, 987, 770, 601, 261, 188, 48, 437, 66, 170, 842, 824, 853, 17, 713, 792, 57,
318, 783, 606, 225, 429, 734, 944, 197, 850, 828, 683, 442, 916, 898, 693, 886, 282, 516, 132, 169, 634, 767, 751, 340, 594, 584, 472, 112,
897, 427, 139, 483, 173, 389, 998, 814, 543, 253, 489, 292, 729, 611, 668, 967, 351, 316, 804, 606, 861, 885, 667, 905, 361, 36, 229, 264, 3
76, 41, 34, 945, 598, 660, 144, 571, 449, 402, 105, 288, 189, 227, 124, 233, 407, 86, 357, 456, 879, 251, 695, 394, 790, 871, 907, 209, 698,
581, 89, 386, 820, 752, 88, 934, 651, 807, 185, 69, 140, 690, 354, 894, 277, 9, 678, 923, 167, 910, 137, 940, 740, 477, 80, 617, 415, 759, 9
79, 91, 964, 936, 494, 315, 555, 30, 166, 238, 451, 248, 501, 743, 181, 632, 656, 199, 797, 835, 825, 866, 778, 410, 709, 25, 368, 525, 706,
937, 930, 77, 106, 810, 557, 452, 145, 521, 605, 59, 730, 440, 867, 912, 212, 514, 602, 453, 992, 577, 40, 195, 618, 122, 430, 146, 999, 85,
870, 64, 340, 657, 138, 644, 874, 65, 168, 153, 241, 45, 459, 462, 230, 109, 384, 633, 746]
```

Рис. 14: Работа с массивами по заданию 3.14

Задание 4

Выполнение лабораторной работы

```
[116]: vec15 = [i^2 for i in 1:100];
       println(vec15)

[1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, 169, 196, 225, 256, 289, 324, 361, 400, 441, 484, 529, 576, 625, 676, 729, 784, 841, 900, 9
61, 1024, 1089, 1156, 1225, 1296, 1369, 1444, 1521, 1600, 1681, 1764, 1849, 1936, 2025, 2116, 2209, 2304, 2401, 2500, 2601, 2704, 2809, 291
6, 3025, 3136, 3249, 3364, 3481, 3600, 3721, 3844, 3969, 4096, 4225, 4356, 4489, 4624, 4761, 4900, 5041, 5184, 5329, 5476, 5625, 5776, 5929,
6084, 6241, 6400, 6561, 6724, 6889, 7056, 7225, 7396, 7569, 7744, 7921, 8100, 8281, 8464, 8649, 8836, 9025, 9216, 9409, 9604, 9801, 10000]
```

Рис. 15: Создание массива квадратов

Задание 5

Выполнение лабораторной работы

```
[119]: using Pkg
       Pkg.add("Primes")

          Updating registry at `~/.julia/registries/General.toml`
         Resolving package versions...
         Installed IntegerMathUtils ─ v0.1.3
         Installed Primes ──────────── v0.5.7
          Updating `~/.julia/environments/v1.11/Project.toml`
        [27ebfcd6] + Primes v0.5.7
          Updating `~/.julia/environments/v1.11/Manifest.toml`
        [18e54dd8] + IntegerMathUtils v0.1.3
        [27ebfcd6] + Primes v0.5.7
      Precompiling project...
          968.7 ms  ✓ IntegerMathUtils
          907.2 ms  ✓ Primes
        2 dependencies successfully precompiled in 3 seconds. 39 already precompiled.

[123]: using Primes
       myprimes = primes(prime(168));
       println(myprimes)

       [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 13
       9, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 29
       3, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 46
       3, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541, 547, 557, 563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641, 643, 64
       7, 653, 659, 661, 673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773, 787, 797, 809, 811, 821, 823, 827, 829, 83
       9, 853, 857, 859, 863, 877, 881, 883, 887, 907, 911, 919, 929, 937, 941, 947, 953, 967, 971, 977, 983, 991, 997]

[124]: myprimes[89]

[124]: 461

[126]: println(myprimes[89:99])

       [461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523]
```

Рис. 16: Работа с пакетом Primes

Задание 6

Выполнение лабораторной работы

```
[127]: sum6_1 = sum(i^3 + 4i^2 for i in 10:100)

[127]: 26052735

[128]: sum6_2 = sum(2^1/i + 3^1/i^2 for i in 1:25)

[128]: 2.1291704368143802e9

[130]: mys_sum = 1
       num = 1
       for i in 2:2:38
           num += 1/(i+1)
           mys_sum += num
       end
       print(mys_sum)

       6.976346137897618
```

Рис. 17: Вычиление сумм

Выводы

Выводы

В результате выполнения данной лабораторной работы я изучила несколько структур данных, реализованных в Julia, научилась применять их и операции над ними для решения задач.