

Лабораторная работа №2

Структуры данных

Ким Реачна¹

17 ноября, 2023, Москва, Россия

¹Российский Университет Дружбы Народов

Цели и задачи

Цель лабораторной работы

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

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

Процесс выполнения лабораторной работы

Операции с кортежами

```
[6]: # Примеры операций над кортежами:  
# длина кортежа x2:  
length(x2)  
  
[6]: 3  
  
[7]: # обратиться к элементам кортежа x2:  
x2[1], x2[2], x2[3]  
  
[7]: (1, 2.0, "tmp")  
  
[8]: # произвести какую-либо операцию (сложение)  
# с вторым и третьим элементами кортежа x1:  
c = x1[2] + x1[3]  
  
[8]: 5  
  
[9]: # обращение к элементам именованного кортежа x3:  
x3.a, x3.b, x3[2]  
  
[9]: (2, 3, 3)  
  
[10]: # проверка вхождения элементов tmp и 0 в кортеж x2  
# (два способа обращения к методу in()):  
in("tmp", x2), 0 in x2  
  
[10]: (true, false)
```

Рис. 1: Примеры операций над кортежами

Операции со словарями

Словари

```
[11]: # создать словарь с именем phonebook:
phonebook = Dict("Иванов И.И." => ("867-5309", "333-5544"), "Бухгалтерия" => "555-2368")

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

[12]: # вывести ключи словаря:
keys(phonebook)

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

[13]: # вывести значения элементов словаря:
values(phonebook)

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

[14]: # вывести заданные в словаре пары "ключ - значение":
pairs(phonebook)

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

[15]: # проверка вхождения ключа в словарь:
haskey(phonebook, "Иванов И.И.")

[15]: true

[16]: # добавить элемент в словарь:
phonebook["Сидоров П.С."] = "555-3344"

[16]: "555-3344"

[17]: # удалить ключ и связанные с ним значения из словаря
pop!(phonebook, "Иванов И.И.")

[17]: ("867-5309", "333-5544")

[18]: # Объединение словарей (функция merge()):
a = Dict{"foo" => 0.0, "bar" => 42.0};
b = Dict{"baz" => 17, "bar" => 13.0};
merge(a, b, merge(b,a))

[18]: (Dict{String, Real}{"bar" => 13.0, "baz" => 17, "foo" => 0.0}, Dict{String, Real}{"bar" => 42.0, "baz" => 17, "foo" => 0.0})
```

Рис. 2: Примеры словарей и операций над словарями

Операции со множествами

```
[61]: # Задания 2
      # Создание множества
      set1 = Set([1, "hello", 3.14, [1, 2, 3]])
      set2 = Set([1, 2, "world", 3.14])

      intersection_set = intersect(set1, set2)
      union_set = union(set1, set2)
      isequal_set = issetequal(set1, set2)
      sub_set = issubset(set1, set2)

      println("Множество 1 : ", set1)
      println("Множество 2 : ", set2)
      println("Пересечение : ", intersection_set)
      println("Объединение : ", union_set)
      println("Эквивалентности множеств : ", isequal_set)
      println("Подмножество : ", sub_set)

      Множество 1 : Set{Any["hello", 3.14, [1, 2, 3], 1]}
      Множество 2 : Set{Any[2, 3.14, 1, "world"]}
      Пересечение : Set{Any[3.14, 1]}
      Объединение : Set{Any[2, "hello", 3.14, [1, 2, 3], 1, "world"]}
      Эквивалентности множеств : false
      Подмножество : false
```

Рис. 3: Задачи с множествами

Операции со массивами

```
[ ]: # Задание 4
squares = [i**2 for i in 1:100]
print("Массив squares: ", squares)

Массив squares: [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, 961, 1024, 1089, 1156, 1225, 1296, 1369, 1444, 1521, 1600, 1681, 1764, 1849, 1936, 2025, 2116, 2209, 2304, 2401, 2500, 2601, 2704, 2809, 2916, 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]

[ ]: # Задание 5
using Primes
myprimes = primes(1000)[1:168]

# Определили 89-е наименьшее простое число.
prime_89 = myprimes[89]
# Получили срез массива с 89-го до 99-го элемента включительно.
slice_89_to_99 = myprimes[89:99]

println("89-е простое число: ", prime_89)
println("Срез массива с 89-го до 99-го элемента myprimes: ", slice_89_to_99)

89-е простое число: 461
Срез массива с 89-го до 99-го элемента myprimes: [461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523]

[ ]: # Задание 6
# 6.1
sum_6_1 = sum(1 -> 1^3 + 41^2, 10:100)
print("6.1. Сумма выражения: ", sum_6_1)

6.1. Сумма выражения: 26852735

[ ]: # 6.2
M = 25
sum_6_2 = sum(1 -> (2^i / i) + (3^i / 1^2), 1:M)
print("6.2. Сумма выражения: ", sum_6_2)

6.2. Сумма выражения: 2.1291704368143802e9

[ ]: # 6.3
sum_6_3 = sum(prod(2 * i / (2 * i - 1) for i in 1:n) for n in 1:20)
println("6.3. Сумма выражения: ", sum_6_3)

6.3. Сумма выражения: 111.00217926389996
```

Рис. 5: Библиотека и вычисление простого числа

Выводы по проделанной работе

Изучила несколько структур данных, реализованных в Julia, научиться применять их и операции над ними для решения задач.