

Презентация по лабораторной работе №1

Дисциплина: Компьютерный практикум по статистическому анализу данных

Лобанова П.И.

9 сентября 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Лобанова Полина Иннокентьевна
- Учащаяся на направлении “Фундаментальная информатика и информационные технологии”
- Студентка группы НФИбд-02-22
- polla-2004@mail.ru

Цель работы

Основная цель работы — подготовить рабочее пространство и инструментарий для работы с языком программирования Julia, на простейших примерах познакомиться с основами синтаксиса Julia.

Задание

1. Установите под свою операционную систему Julia, Jupyter (разделы 1.3.1 и 1.3.2).
2. Используя Jupyter Lab, повторите примеры из раздела 1.3.3.
3. Выполните задания для самостоятельной работы (раздел 1.3.4).

Выполнение

```
[3]:
typeof(3), typeof(3.5), typeof(3/3.5), typeof(sqrt(3+4im)), typeof(pi)

[3]:
(Int64, Float64, Float64, ComplexF64, Irrational{π})

[4]:
1.0/0.0, 1.0/(-0.0), 0.0/0.0

[4]:
(Inf, -Inf, NaN)

[5]:
typeof(1.0/0.0), typeof(1.0/(-0.0)), typeof(0.0/0.0)

[5]:
(Float64, Float64, Float64)

[2]:
for T in [Int8, Int16, Int32, Int64, Int128, UInt8, UInt16, UInt32, UInt64, UInt128]
    println("${lpad(T,7)}: [$(typemin(T)), $(typemax(T))]" )
end

Int8: [-128,127]
Int16: [-32768,32767]
Int32: [-2147483648,2147483647]
Int64: [-9223372036854775808,9223372036854775807]
Int128: [-170141183460469231731687303715884105728,170141183460469231731687303715884105727]
UInt8: [0,255]
UInt16: [0,65535]
UInt32: [0,4294967295]
UInt64: [0,18446744073709551615]
UInt128: [0,340282366920938463463374607431768211455]
```

Рис. 1: Примеры определения типа числовых величин

```
Int64(2.0), Char(2)
```

```
[6]:
```

```
(2, '\x02')
```

```
[7]:
```

```
convert(Int64, 2.0), convert(Char, 2)
```

```
[7]:
```

```
(2, '\x02')
```

```
[8]:
```

```
Bool(1), Bool(0)
```

```
[8]:
```

```
(true, false)
```

```
[10]:
```

```
typeof(promote(Int8(1), Float16(4.5), Float32(4.1)))
```

```
[10]:
```

```
Tuple{Float32, Float32, Float32}
```

Рис. 2: Примеры приведения аргументов к одному типу

```
function f(x)
    x^2
end
f(4)
```

[12]:

16

[15]:

```
g(x) = x^2
g(8)
```

[15]:

64

Рис. 3: Примеры определения функций

[14]:

```
a = [4 7 6]
b = [1, 2, 3]
a[2], b[2]
```

[14]:

(7, 2)

[18]:

```
a = 1; b = 2; c = 3; d = 4
Am = [a b; c d]
Am[1,1], Am[1,2], Am[2,1], Am[2,2]
```

[18]:

(1, 2, 3, 4)

[20]:

```
aa = [1 2]
AA = [1 2; 3 4]
aa*AA*aa'
```

[20]:

```
1x1 Matrix{Int64}:
 27
```

Рис. 4: Примеры работы с массивами

```
[80]:  
f1 = open("file1.txt")  
  
[80]:  
IOStream(<file file1.txt>)  
  
[63]:  
read(f1, String)  
  
[63]:  
"Hello World\r\nHello Worl\r\nHello Wor"  
  
[77]:  
readline(f1)  
  
[77]:  
"Hello World"  
  
[81]:  
readlines(f1)  
  
[81]:  
3-element Vector{String}:  
"Hello World"  
"Hello Worl"  
"Hello Wor"
```

Рис. 5: Примеры использования функций `open()`, `read()`, `readline()`, `readlines()`

```
[88]: print("Hello World!")
      Hello World!

[89]: println("Hello World!")
      Hello World!

[102]: for i in "Hello World!"
      print(i)
      end
      Hello World!

[103]: for i in "Hello World!"
      println(i)
      end
      H
      e
      l
      l
      o

      H
      o
      r
      d
      !
```

Рис. 6: Примеры использования функций `print()`, `println`

```
[92]: show("Hello World!")  
      "Hello World!"  
[104]: io = IOBuffer()  
[104]: IOBuffer(data=UInt8[...], readable=true, writable=true, seekable=true, append=false, size=0, maxsize=Inf, ptr=1, mark=-1)  
[105]: write(io, "Hello")  
      String(take!(io))  
[105]: "Hello"  
[106]: write(io, "Hello World")  
      String(take!(io))  
[106]: "Hello World"
```

Рис. 7: Примеры использования функций `show()`, `write()`

[115]:

```
parse(Int, "1234")
```

[115]:

1234

[116]:

```
parse(Float64, "0.256")
```

[116]:

0.256

[117]:

```
parse(Int, "1234", base = 5)
```

[117]:

194

[118]:

```
parse(Float64, "1.2e-3")
```

[118]:

0.0012

Рис. 8: Примеры использования функции `parse()`


```
[134]: a = 1+2
b = 0.5896 + 1.28j
c = "hello" * "world"
d = 8-9
e = 0.56-0.04j
print(a, '\n', b, '\n', c, '\n', d, '\n', e)

3
1.8696000000000002j
helloworld
-1
0.52
```

```
[136]: a1 = 58*97
b1 = 0.5*1.6
c1 = 25/5
d1 = 0.687/0.1
print(a1, '\n', b1, '\n', c1, '\n', d1)

5626
0.8
5.0
6.87
```

```
[137]: a2 = 2^3
b2 = 0.25^2
c2 = sqrt(144)
d2 = sqrt(2.89)
print(a2, '\n', b2, '\n', c2, '\n', d2)

8
0.0625
12.0
1.7
```

Рис. 9: Примеры базовых математических операций

```
[138]: a3 = 2<4
      b3 = 25<8
      c3 = 0==0
      d3 = 2!=7
      print(a3, '\n', b3, '\n', c3, '\n', d3)

      true
      false
      true
      true

[141]: a4 = !true
      b4 = !false
      c4 = true && true
      d4 = false && true
      e4 = true || false
      f4 = true || true
      print(a4, '\n', b4, '\n', c4, '\n', d4, '\n', e4, '\n', f4)

      false
      true
      true
      false
      true
      true
```

Рис. 10: Примеры базовых математических операций

[149]:

```
A = [1 2 3; 4 5 6; 7 8 9]
B = [9 8 7; 6 5 4; 3 2 1]
C = A+B
D = A-B
E = A*B
F = A'
print(A, '\n', B, '\n', C, '\n', D, '\n', E, '\n', F)
```

```
[1 2 3; 4 5 6; 7 8 9]
[9 8 7; 6 5 4; 3 2 1]
[10 10 10; 10 10 10; 10 10 10]
[-8 -6 -4; -2 0 2; 4 6 8]
[30 24 18; 84 69 54; 138 114 90]
[1 4 7; 2 5 8; 3 6 9]
```

[164]:

```
using LinearAlgebra
A1 = [1 2 3]
B1 = [4 5 6]
C1 = A1*2
D1 = dot(A1,B1)
print(A1, '\n', B1, '\n', C1, '\n', D1, '\n')
```

```
[1 2 3]
[4 5 6]
[2 4 6]
32
```

Рис. 11: Примеры с операциями над матрицами и векторами

Вывод

Я подготовила рабочее пространство и инструментарий для работы с языком программирования Julia, на простейших примерах познакомилась с основами синтаксиса Julia.