

# **Отчёт по лабораторной работе №1**

## **Компьютерный практикум по**

### **статистическому анализу данных**

**Julia. Установка и настройка. Основные принципы.**

Выполнил: Махорин Иван Сергеевич,  
НПИбд-02-21, 1032211221

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>5</b>
2.1	Подготовка инструментария к работе . . . . .	5
2.2	Основы синтаксиса Julia на примерах . . . . .	7
2.3	Самостоятельная работа . . . . .	9
<b>3</b>	<b>Вывод</b>	<b>13</b>
<b>4</b>	<b>Список литературы. Библиография</b>	<b>14</b>

## Список иллюстраций

2.1	Установка менеджера пакетов Chocolatey . . . . .	5
2.2	Установка Far Manager . . . . .	6
2.3	Установка Notepad++ . . . . .	6
2.4	Установка Julia . . . . .	6
2.5	Установка Anaconda Distribution (Python 3.x) . . . . .	7
2.6	Установка пакетов для работы с Jupyter . . . . .	7
2.7	Примеры определения типа числовых величин . . . . .	8
2.8	Примеры приведения аргументов к одному типу . . . . .	8
2.9	Примеры определения функций . . . . .	9
2.10	Примеры работы с массивами . . . . .	9
2.11	Примеры работы с функциями для чтения/записи/вывода информации на экран . . . . .	10
2.12	Пример работы с функцией parse . . . . .	10
2.13	Примеры работы базовых математических операций . . . . .	11
2.14	Примеры работы базовых математических операций . . . . .	11
2.15	Примеры работы с операциями над матрицами . . . . .	12

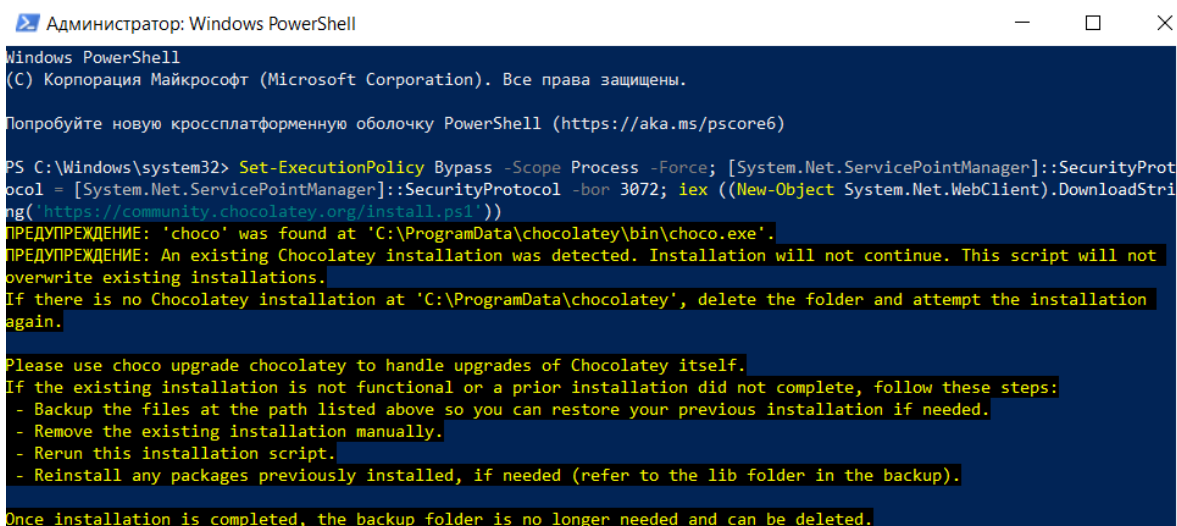
# 1 Цель работы

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

## 2 Выполнение лабораторной работы

### 2.1 Подготовка инструментария к работе

Так как мы используем ОС типа Windows для различных установок будем использовать менеджер пакетов Chocolatey (<https://chocolatey.org/>), который установим через Administrative Shell (рис. 2.1):



```
Администратор: Windows PowerShell
Windows PowerShell
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

Попробуйте новую кроссплатформенную оболочку PowerShell (https://aka.ms/pscore6)

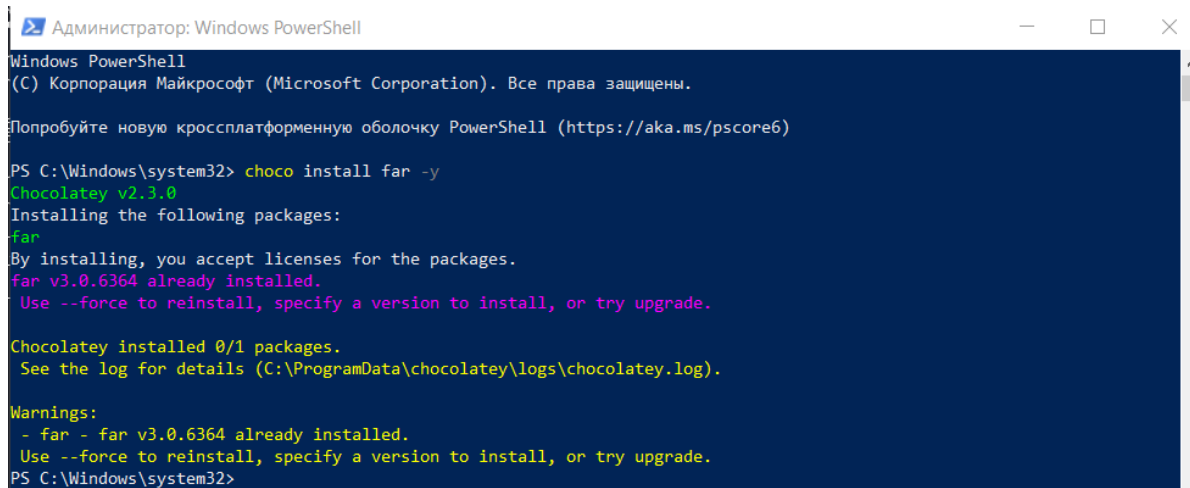
PS C:\Windows\system32> Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))
ПРЕДУПРЕЖДЕНИЕ: 'choco' was found at 'C:\ProgramData\chocolatey\bin\choco.exe'.
ПРЕДУПРЕЖДЕНИЕ: An existing Chocolatey installation was detected. Installation will not continue. This script will not
overwrite existing installations.
If there is no Chocolatey installation at 'C:\ProgramData\chocolatey', delete the folder and attempt the installation
again.

Please use choco upgrade chocolatey to handle upgrades of Chocolatey itself.
If the existing installation is not functional or a prior installation did not complete, follow these steps:
- Backup the files at the path listed above so you can restore your previous installation if needed.
- Remove the existing installation manually.
- Rerun this installation script.
- Reinstall any packages previously installed, if needed (refer to the lib folder in the backup).

Once installation is completed, the backup folder is no longer needed and can be deleted.
```

Рис. 2.1: Установка менеджера пакетов Chocolatey

Далее посредством установленного менеджера установим Far Manager, Notepad++, Julia, Anaconda Distribution (Python 3.x) (рис. 2.2 - рис. 2.5):



```
Администратор: Windows PowerShell
Windows PowerShell
(C) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

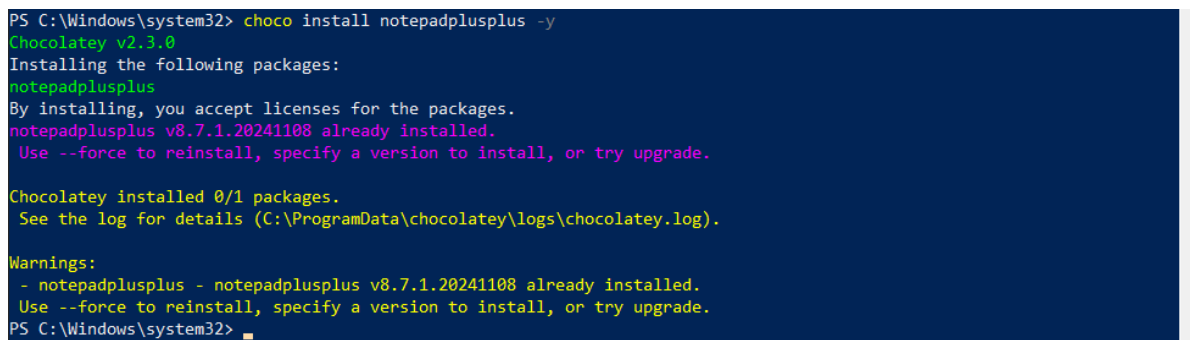
Попробуйте новую кроссплатформенную оболочку PowerShell (https://aka.ms/pscore6)

PS C:\Windows\system32> choco install far -y
Chocolatey v2.3.0
Installing the following packages:
far
By installing, you accept licenses for the packages.
far v3.0.6364 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.

Chocolatey installed 0/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Warnings:
- far - far v3.0.6364 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.
PS C:\Windows\system32>
```

Рис. 2.2: Установка Far Manager

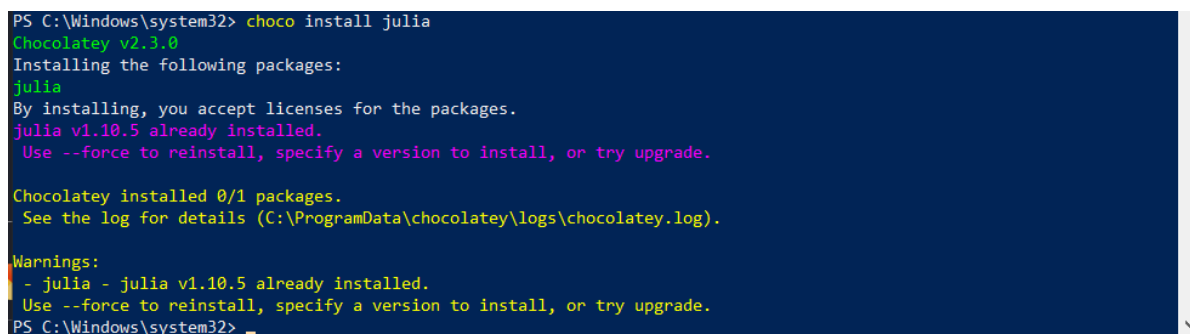


```
PS C:\Windows\system32> choco install notepadplusplus -y
Chocolatey v2.3.0
Installing the following packages:
notepadplusplus
By installing, you accept licenses for the packages.
notepadplusplus v8.7.1.20241108 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.

Chocolatey installed 0/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Warnings:
- notepadplusplus - notepadplusplus v8.7.1.20241108 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.
PS C:\Windows\system32>
```

Рис. 2.3: Установка Notepad++



```
PS C:\Windows\system32> choco install julia
Chocolatey v2.3.0
Installing the following packages:
julia
By installing, you accept licenses for the packages.
julia v1.10.5 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.

Chocolatey installed 0/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Warnings:
- julia - julia v1.10.5 already installed.
Use --force to reinstall, specify a version to install, or try upgrade.
PS C:\Windows\system32>
```

Рис. 2.4: Установка Julia



#### Примеры определения типа числовых величин

```
[3]: typeof(3), typeof(3.5), typeof(3/3.55), 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)

[6]: 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. 2.7: Примеры определения типа числовых величин

После чего приступим к рассмотрению приведения аргументов к одному типу (рис. 2.8):

#### Примеры приведения аргументов к одному типу

```
[7]: Int64(2.0), Char(2), typeof(Char(2))
[7]: (2, '\x02', Char)

[8]: convert(Int64, 2.0), convert(Char,2)
[8]: (2, '\x02')

[9]: typeof(promote(Int8(1), Float16(4.5), Float32(4.1)))
[9]: Tuple{Float32, Float32, Float32}
```

Рис. 2.8: Примеры приведения аргументов к одному типу

И рассмотрим примеры определения функций (рис. 2.9), а также работу с массивами (рис. 2.10):



#### Примеры определения функций

```
[10]: function f(x)
      x^2
      end

[10]: f (generic function with 1 method)

[11]: f(4)

[11]: 16

[12]: g(x)=x^2

[12]: g (generic function with 1 method)

[13]: g(8)

[13]: 64
```

Рис. 2.9: Примеры определения функций

#### Примеры работы с массивами

```
[16]: a = [4 7 6] # вектор-строка
      b = [1, 2, 3] # вектор-столбец
      a[2], b[2] # вторые элементы векторов a и b

[16]: (7, 2)

[18]: a = 1; b = 2; c = 3; d = 4 # присвоение значений
      Am = [a b; c d] # матрица 2 x 2

[18]: 2×2 Matrix{Int64}:
      1  2
      3  4

[19]: Am[1,1], Am[1,2], Am[2,1], Am[2,2] # элементы матрицы Am

[19]: (1, 2, 3, 4)

[20]: aa = [1 2] # вектор-строка
      AA = [1 2; 3 4] # матрица 2 x 2
      aa*AA*aa' # умножение вектора-строки на матрицу и на вектор-столбец (операция транспонирования)

[20]: 1×1 Matrix{Int64}:
      27

[21]: aa, AA, aa'

[21]: ([1 2], [1 2; 3 4], [1; 2; ;])
```

Рис. 2.10: Примеры работы с массивами

## 2.3 Самостоятельная работа

В первом задании мы рассмотрим основные функции для чтения / записи / вывода информации на экран. Для этого составим свои примеры (рис. 2.11):

#### Примеры работы с функциями для чтения/записи/вывода информации на экран

```
[53]: write("myfile.txt", "Hello, world!\nMy name is Vanya.\n")
```

```
[53]: 32
```

```
[54]: io = open("myfile.txt", "r")  
      read(io, String)
```

```
[54]: "Hello, world!\nMy name is Vanya.\n"
```

```
[55]: readline("myfile.txt")
```

```
[55]: "Hello, world!"
```

```
[56]: readlines("myfile.txt")
```

```
[56]: 2-element Vector{String}:  
      "Hello, world!"  
      "My name is Vanya."
```

```
[57]: print("Hello, World!")
```

```
Hello, World!
```

```
[58]: println("Hello", ' ', " world.")
```

```
Hello, world.
```

```
[59]: show("Hello, World!")
```

```
"Hello, World!"
```

Рис. 2.11: Примеры работы с функциями для чтения/записи/вывода информации на экран

Во втором задании составим пример для функции `parse()` (рис. 2.12):

#### Пример работы с функцией `parse`

```
[60]: ex1 = Meta.parse("(4 + 4) / 2")
```

```
[60]: :((4 + 4) / 2)
```

Рис. 2.12: Пример работы с функцией `parse`

Далее изучим синтаксис Julia для базовых математических операций с разным типом переменных (рис. 2.13 - рис. 2.14):

#### Примеры работы базовых математических операций

```
[62]: 1 + 1
[62]: 2
[63]: 5 - 1.0
[63]: 4.0
[65]: 2.0 * 3.0
[65]: 6.0
[67]: 6 / 2
[67]: 3.0
[70]: 5 + 2
[70]: 7
[73]: 7.0 % 6
[73]: 1.0
[72]: 2^5
[72]: 32
```

Рис. 2.13: Примеры работы базовых математических операций

```
[74]: 1 == 1
[74]: true
[75]: 2 != 2
[75]: false
[76]: 2.0 < 2
[76]: false
[77]: 3 >= 2.5
[77]: true
[79]: x = true
      !x
[79]: false
[84]: x = true
      y = false
      x || y
[84]: true
```

Рис. 2.14: Примеры работы базовых математических операций

В конце работы приведём несколько примеров с операциями над матрицами (рис. 2.15):

### Примеры работы с операциями над матрицами

```
[92]: using LinearAlgebra  
A = [1 2 3; 4 1 6; 7 8 1]
```

```
[92]: 3×3 Matrix{Int64}:  
 1  2  3  
 4  1  6  
 7  8  1
```

```
[93]: det(A)
```

```
[93]: 104.0
```

```
[94]: tr(A)
```

```
[94]: 3
```

```
[95]: inv(A)
```

```
[95]: 3×3 Matrix{Float64}:  
 -0.451923  0.211538  0.0865385  
  0.365385 -0.192308  0.0576923  
  0.240385  0.0576923 -0.0673077
```

```
[96]: B = [1 1 1; 1 1 1; 1 1 1]  
A - B
```

```
[96]: 3×3 Matrix{Int64}:  
 0  1  2  
 3  0  5  
 6  7  0
```

```
[97]: A + B
```

```
[97]: 3×3 Matrix{Int64}:  
 2  3  4  
 5  2  7  
 8  9  2
```

```
[98]: A * B
```

```
[98]: 3×3 Matrix{Int64}:  
 6  6  6  
 11 11 11  
 16 16 16
```

Рис. 2.15: Примеры работы с операциями над матрицами

## 3 Вывод

В ходе выполнения лабораторной работы были получены навыки по подготовке рабочего пространства и инструментария для работы с языком программирования Julia, а также познакомились на простейших примерах с основами синтаксиса Julia.

## 4 Список литературы. Библиография

[1] Julia Documentation: <https://docs.julialang.org/en/v1/>