Язык Julia

Начало работы с Julia:

* Язык Julia - (<https://julialang.org/>) - Официальный веб-сайт Julia с документацией, учебными пособиями и ресурсами.
* Самоучитель Julia - (<https://docs.julialang.org/en/v1/manual/getting-started/>) - Официальный самоучитель Julia, посвященный основам языка.
* Julia на примере - ([ht](http://ihed.ras.ru/~thermo/Julia/Brief%20description%20of%20Julia%20language.pdf)) - Краткое описание языка программирования Julia с примерами его использования.
* JuliaLang на GitHub - (<https://github.com/JuliaLang>) - Официальный репозиторий Julia на GitHub с исходным кодом, проблемами и обсуждениями.

Примеры и фрагменты кода

Базовые операции

**println("Hello, World!")**

Базовая арифметика

**x = 5**

**y = 3**

**println("x + y = ", x + y) # Output: x + y = 8**

Структура данных

Массивы

**arr = [1, 2, 3, 4, 5]**

**println("Array: ", arr) # Output: Array: [1, 2, 3, 4, 5]**

Словари

**dict = Dict("name" => "John", "age" => 30)**

**println("Dictionary: ", dict) # Output: Dictionary: Dict("name"=>"John", "age"=>30)**

Поток управления

Оператор if-else

**x = 5**

**if x > 10**

**println("x is greater than 10")**

**else**

**println("x is less than or equal to 10") # Output: x is less than or equal to 10**

**end**

Цикл for

**arr = [1, 2, 3, 4, 5]**

**for elem in arr**

**println("Element: ", elem)**

**end**

Функции

Простые функции

**function greet(name)**

**println("Hello, ", name, "!")**

**end**

**greet("John") # Output: Hello, John!**

Рекурсивные функции

**function factorial(n)**

**if n == 0**

**return 1**

**else**

**return n \* factorial(n-1)**

**end**

**end**

**println("Factorial of 5: ", factorial(5))**

Примеры задач

Последовательность Фибоначчи

**function fibonacci(n)**

**if n <= 1**

**return n**

**else**

**return fibonacci(n-1) + fibonacci(n-2)**

**end**

**end**

**for i in 1:10**

**println("Fibonacci number $i: ", fibonacci(i-1))**

**end**

Умножение матриц

**using LinearAlgebra**

**A = [1 2; 3 4]**

**B = [5 6; 7 8]**

**C = A \* B**

**println("Matrix A: ")**

**display(A)**

**println("Matrix B: ")**

**display(B)**

**println("Matrix C (A \* B): ")**

**display(C)**

Сортировка массива

**arr = [5, 2, 8, 3, 1, 6, 4]**

**# Sort the array in ascending order**

**sorted\_arr = sort(arr)**

**println("Original array: ", arr)**

**println("Sorted array: ", sorted\_arr)**

Найти максимальное значение в массиве

**arr = [3, 1, 4, 1, 5, 9, 2]**

**max\_val = maximum(arr)**

**println("Array: ", arr)**

**println("Maximum value: ", max\_val)**