

Аннотированные статьи и ресурсы по языку программирования Julia:

1. **Официальный сайт Julia**  
На официальном сайте Julia представлено множество учебных материалов, включая вводные курсы, мастер-классы и руководства по применению языка в научных и инженерных задачах.  
<https://julialang.org/learning/>
2. **Книга "Julia. Язык программирования. Быстрый старт"**  
Автор: Вадим Никитин. Книга предлагает простое объяснение ключевых концепций языка, а также примеры решения задач.  
<https://www.litres.ru/book/vadim-nikitin-32700223/julia-yazykprogramirovaniya-bystryy-start-69596290/>
3. **Пособие "Язык Julia как инструмент исследователя"**  
Автор: В. А. Антонюк. Пособие знакомит читателей с основами Julia и включает примеры решения задач.  
[https://cmp.phys.msu.ru/sites/default/files/VA\\_Antonyk\\_Julia\\_2019.pdf](https://cmp.phys.msu.ru/sites/default/files/VA_Antonyk_Julia_2019.pdf)
4. **Учебное пособие "Язык программирования математических вычислений Julia. Базовое руководство"**  
Автор: Шиндина. Содержит инструкции по установке интерпретатора, основные принципы работы и возможности языка.  
<https://studfile.net/preview/16721476/>
5. **Книга "Осваиваем язык Julia"**  
Автор: Малькольм Шеррингтон. Книга предназначена для специалистов по анализу данных и разработчиков, желающих изучить Julia.  
<https://www.litres.ru/book/malkolm-sherrington/osvaivaem-yazyk-julia-22861130/>
6. **Статья "Научное программирование на языке JULIA"**  
Автор: Евгения Александровна Оконешникова. В статье приведены решения задач математической физики с использованием Julia.  
[https://elar.rsvpu.ru/bitstream/123456789/36394/1/nito\\_2021\\_4\\_019.pdf](https://elar.rsvpu.ru/bitstream/123456789/36394/1/nito_2021_4_019.pdf)
7. **Презентация "Научное программирование на языке Julia"**  
Автор: Камиль Хайруллин. Презентация охватывает особенности языка и средства разработки.  
[https://exponenta.ru/storage/app/media/Conf\\_2023/%B5\\_5.04.2023/khirulin\\_kamil.pdf](https://exponenta.ru/storage/app/media/Conf_2023/%B5_5.04.2023/khirulin_kamil.pdf)

## 1. Сумма элементов массива

- Функция `sum_array(arr)` принимает массив `arr`.
- Переменная `sum` инициализируется значением 0.
- В цикле `for` проходим по каждому элементу массива и добавляем его к `sum`.
- Функция возвращает итоговую сумму.
- Пример использования:

```
1  function sum_array(arr)
2      sum = 0
3      for i in arr
4          sum += i
5      end
6      return sum
7  end
8  arr = [1, 2, 3, 4, 5]
9  println("Сумма элементов массива: ", sum_array(arr))
```

## 2. Факториал числа

- Функция `factorial(n)` рекурсивно вычисляет факториал числа `n`.
- Если `n == 0`, возвращается 1.
- Пример использования:

```
1  function factorial(n)
2      n == 0 ? 1 : n * factorial(n - 1)
3  end
4  num = 5
5  println("Факториал числа 5: ", factorial(num))
```

### 3. Решение квадратного уравнения

- Функция `solve_quadratic(a, b, c)` решает уравнение вида  $ax^2+bx+c=0$ .
- Вычисляется дискриминант  $D=b^2-4ac$ .
- В зависимости от значения дискриминанта возвращаются корни или сообщение об их отсутствии.
- Пример использования:

```
1  function solve_quadratic(a, b, c)
2      discriminant = b^2 - 4*a*c
3      if discriminant < 0
4          return "Нет действительных корней"
5      elseif discriminant == 0
6          x = -b / (2*a)
7          return "Один корень: $x"
8      else
9          x1 = (-b + sqrt(discriminant)) / (2*a)
10         x2 = (-b - sqrt(discriminant)) / (2*a)
11         return "Два корня: $x1 и $x2"
12     end
13 end
14 println(solve_quadratic(1, -3, 2))
```