

Лабораторная работа N1

1. На языке Python написать функцию возвращающую объем цилиндра с заданными радиусом R и длины L .

2. Пользуясь методами из модуля `scipy.optimize` найти корень функции $f(x) = \frac{\cos x}{1+x^2}$ на промежутке $[0.1, 2.4]$. Сравнить скорости расчета различными методами: методом деления отрезка (`scipy.optimize.bisect`), методом Ньютона (`scipy.optimize.newton` с заданием производной), методом секущих (`scipy.optimize.newton`) и методом Брента (`scipy.optimize.brentq`).

3. С использованием функций решения системы линейных уравнений из модуля `scipy.linalg` решить систему:

$$\begin{aligned}x_0 &= 0, \\x_1 + x_0 &= 1, \\x_2 + x_1 + x_0 &= 2, \\x_3 + x_2 + x_1 &= 3, \\&\dots \\x_n + x_{n-1} + x_{n-2} &= n\end{aligned}\tag{1}$$

для $n = 100$.

Создать матрицу можно следующим образом. На примере матрицы 5×5 : сначала, создаем матрицу из нулей,

```
A = np.zeros((5,5))
```

Далее, создать массив индексов i и j ,

```
i,j = np.indices(A.shape)
```

которые могут быть использованы для задания нужных нам элементов матрицы A . Например, выражение `A[i==j+1]=1` приравнивает значение 1 всем элементам под диагональю.

4. Случайная матрица A размера $N \times N$ с $N = 50$ задана выражением на языке Python:

```
A = np.random.random((N,N))
```

Зависящий от времени t вектор $\mathbf{b}(t) = \{b_k(t)\}$ дается формулой

$$b_k(t) = \frac{k}{1 + kt}.$$

Найти изменение вектора $\mathbf{x}(t)$ на промежутке $t_{start} = 0$ до $t_{finish} = 10$ с шагом по времени $\Delta t = 0.1$, если эволюция вектора $\mathbf{x}(t)$ описывается системой линейных уравнений,

$$A\mathbf{x}(t) = \mathbf{b}(t).$$

Сравнить и прокомментировать эффективность алгоритмов с применением методов: `scipy.linalg.solve`, `scipy.linalg.lu_solve` и `scipy.linalg.inv` (нахождение обратной матрицы).