

ЛАБОРАТОРНАЯ РАБОТА №1

Введение в базовые операции

ВАРИАНТ 1

Во всех заданиях для генерирования случайных величин можно пользоваться **только** командами **rand** и **randn**. Все задания должны выполняться с минимальным использованием циклов. Каждое задание должно быть оформлено как блок в скрипте. В каждом задании, где нужно что-либо “проверить”, программа должна выводить адекватное сообщение об ошибке в случае невыполнения проверяемого условия.

Любое измерение скорости работы алгоритма должно производиться по большому количеству запусков алгоритма при фиксированных параметрах, с последующим усреднением результатов.

1 [0,5]. Задать два вещественных числа (a и b), натуральное число n и равномерную сетку на $[a, b]$ с n точками. Задать функцию $f(x) = \sin(\ln(1 + |x|) - x^2)$. Нарисовать график её значений на сетке, отметить на графике отдельно глобальные максимум и минимум, а соответствующие значения функции вывести в командное окно.

2 [0,5]. Запросить у пользователя ввод числа n . Проверить, что введенное число — натуральное.

1. Создать вектор из всех нечетных чисел, делящихся на 9, из промежутка от 1 до n .

2. Построить матрицу размера $n \times n$, все элементы i -й строки которой равны i .

3. Создать матрицу B $n \times (n + 1)$ вида

$$B = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$$

Вытянуть матрицу B в вектор s . Присвоить переменной D последние 2 столбца матрицы B .

3 [0,5]. Создать матрицу размера 7×7 , состоящую из случайных элементов с равномерным распределением среди натуральных чисел от 1 до 315, найти максимальный элемент на диагонали этой матрицы, найти максимальное и минимальное отношение произведения к сумме для строк этой матрицы, отсортировать строки матрицы в лексикографическом порядке (то есть строка $[a_1, a_2, a_3, \dots, a_n]$ стоит в матрице выше строки $[b_1, b_2, b_3, \dots, b_n]$, если $a_i = b_i$ при $i = 1, \dots, k - 1$ и $a_k < b_k$ для некоторого k).

4 [0,5]. Построить таблицу умножения всевозможных пар элементов таких, что первый — элемент вектора X , а второй — вектора Y :

$$\begin{bmatrix} x_1 y_1 & x_1 y_2 & x_1 y_3 \\ x_2 y_1 & x_2 y_2 & x_2 y_3 \\ x_3 y_1 & x_3 y_2 & x_3 y_3 \end{bmatrix}$$

5 [0,5]. Запросить у пользователя ввод числа n . Проверить, что введенное число — простое. Создать случайную матрицу $A \in \mathbb{R}^{n \times n}$ и вектор $b \in \mathbb{R}^{n \times 1}$, в случае, если A не вырождена, решить уравнение $Ax = b$ (решить задачу не менее чем двумя способами и вставить проверку возможности решения и правильности решения).

6 [0,5]. Даны векторы a размерности n и b размерности m . Найти, используя только арифметические операции и команды **max** и **min**, максимум функции $|a_i - b_j|$, где a_i — элемент вектора a , b_j — элемент вектора b . Функцию **abs** и дополнительную память не использовать.

7 [0,5]. Пусть у нас задано n точек в пространстве \mathbb{R}^k в виде матрицы **double[n,k]**. Требуется построить матрицу **double[n,n]** расстояний между каждой парой точек. Пользоваться командами **pdist** и **squareform** нельзя.

8 [0,5]. Построить матрицу, в которой по строкам записаны все n -мерные бинарные векторы. Натуральное число n задается пользователем.

9 [0,5]. Реализовать функцию **C = my_multiply(A,B)**, которая выполняет расчет значения $C = AB$ по определению («строка на столбец»). Сравнить быстродействие этой функции и стандартного умножения матриц для матриц различной размерности. Построить график времени работы в зависимости от размера матриц (в случае квадратных матриц).

10 [0,5]. Напишите функцию, которая находит средние значения (по одному направлению) с учётом NaN элементов матрицы. Для

$$X = \begin{bmatrix} NaN & 1 & 2 \\ NaN & 0 & 6 \\ 1 & 5 & NaN \end{bmatrix}$$

ответ [1, 2, 4]. Команду **nanmean** использовать нельзя.

11 [1]. Сгенерировать вектор из n случайных величин с нормальным распределением $N(a, \sigma^2)$. Проверить «правило трёх сигм»: вывести долю элементов вектора, находящихся в интервале $[a - 3\sigma, a + 3\sigma]$.

12 [2]. По аналогии с функцией **trapz** реализовать аналогичные функции **rectangles** (интегрирование методом прямоугольников) и **simpson** (методом Симпсона). С помощью них построить график первообразной функции $f(x) = \sin(x)/x$. Сравнить внутреннюю скорость сходимости при использовании всех трёх методов (внутренняя скорость сходимости определяется с помощью сравнения разностей решений при шаге h и $h/2$, нарисовать график этой ошибки в зависимости от h). Сравнить время вычисления.

13 [1]. Задать формулу для некоторой функции и её производной. На одном графике в логарифмическом масштабе (**loglog**) вывести модули разностей между точным значением производной в некоторой точке и правой и центральной разностной производной в зависимости от шага численного дифференцирования. Для генерации сеток, удобных для визуализации графиков в логарифмическом масштабе, рекомендуется использовать команду **logspace**.

ЛАБОРАТОРНАЯ РАБОТА №1

Введение в базовые операции

ВАРИАНТ 2

Во всех заданиях для генерирования случайных величин можно пользоваться **только** командами **rand** и **randn**. Все задания должны выполняться с минимальным использованием циклов. Каждое задание должно быть оформлено как блок в скрипте. В каждом задании, где нужно что-либо “проверить”, программа должна выводить адекватное сообщение об ошибке в случае невыполнения проверяемого условия.

Любое измерение скорости работы алгоритма должно производиться по большому количеству запусков алгоритма при фиксированных параметрах, с последующим усреднением результатов.

1 [0,5]. Задать два вещественных числа (a и b), натуральное число n и равномерную сетку на $[a, b]$ с n точками. Задать функцию $f(x) = \cos(x^2 - 4|x|)$. Нарисовать график её значений на сетке, отметить на графике отдельно глобальные максимум и минимум, а соответствующие значения функции вывести в командное окно.

2 [0,5]. Запросить у пользователя ввод числа n . Проверить, что введенное число — простое.

1. Создать вектор из всех нечетных чисел, делящихся на 7, из промежутка от 1 до n .

2. Построить матрицу размера $n \times n$, все элементы i -й строки которой равны $i + 1$.

3. Создать матрицу B $(n + 1) \times (n + 1)$ вида

$$B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Вытянуть матрицу B в вектор c . Присвоить переменной D последние 2 столбца матрицы B .

3 [0,5]. Создать матрицу размера 5×8 , состоящую из случайных элементов с равномерным распределением среди целых чисел от -10 до 10, найти максимальный элемент на диагонали этой матрицы, найти максимальное и минимальное отношение произведения к сумме для строк этой матрицы, отсортировать строки матрицы в обратном лексикографическом порядке (то есть строка $[a_1, a_2, a_3, \dots, a_n]$ стоит в матрице ниже строки $[b_1, b_2, b_3, \dots, b_n]$, если $a_i = b_i$ при $i = 1, \dots, k - 1$ и $a_k < b_k$ для некоторого k).

4 [0,5]. Реализовать разбиение произвольной матрицы $A \in \mathbb{R}^{n \times m}$ на матрицы R, G, B по следующему правилу:

$$A = \begin{bmatrix} G_{11} & R_{11} & G_{12} & R_{12} & \dots \\ B_{11} & G_{21} & B_{12} & G_{22} & \dots \\ G_{31} & R_{21} & G_{32} & R_{22} & \dots \\ B_{21} & G_{41} & B_{22} & G_{42} & \dots \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix}$$

Предусмотреть в программе проверки выполнения условий, когда указанное разбиение на матрицы можно построить.

5 [0,5]. Для пар векторов $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ построить матрицу $A \in \mathbb{R}^{nm \times 2}$, строки которой — все пары декартова произведения $x \times y$.

6 [0,5]. Задан $3 \times n$ массив точек, интерпретируемый как координаты векторов $x_1, x_2, \dots, x_n \in \mathbb{R}^3$. Построить матрицу $A \in \mathbb{R}^{n \times n}$, такую, что $a_{ij} = |x_i \times x_j|$ (модуль векторного произведения).

7 [0,5]. Даны векторы a размерности n и b размерности m . Найти, используя только арифметические операции и команды **max** и **min**, максимум функции $|a_i - b_j|$, где a_i — элемент вектора a , b_j — элемент вектора b . Функцию **abs** и дополнительную память не использовать.

8 [0,5]. Пусть у нас задано n точек в пространстве \mathbb{R}^k в виде матрицы **double[n,k]**. Требуется построить матрицу **double[n,n]** расстояний между каждой парой точек. Пользоваться командами **pdist** и **squareform** нельзя.

9 [0,5]. Реализовать функцию **C = my_add(A,B)**, которая выполняет сложение матриц $C = A + B$ по определению. Сравнить быстродействие этой функции и стандартного сложения матриц для матриц различной размерности. Построить график времени работы.

10 [0,5]. Проверить, является ли вектор A симметричным. Например, векторы $A = [3, 4, 5, 4, 3]$, $A = [6, 6]$, $A = [7]$ являются, а векторы $A = [1, 2]$, $A = [1, 2, 3, 4, 1]$ — нет.

11 [1]. Сгенерировать вектор из n случайных величин с равномерным распределением на отрезке $[0, a]$. Проверить неравенство Маркова: для заданного числа $b > 0$ вывести долю элементов вектора, больших b , и сравнить с числом $a/2b$.

12 [2]. По аналогии с функцией **trapz** реализовать аналогичные функции **rectangles** (интегрирование методом прямоугольников) и **simpson** (методом Симпсона). С помощью них построить график первообразной функции $f(x) = \exp(-x^2)$. Сравнить внутреннюю скорость сходимости при использовании всех трёх методов (внутренняя скорость сходимости определяется с помощью сравнения разностей решений при шаге h и $h/2$, нарисовать график этой ошибки в зависимости от h). Сравнить время вычисления.

13 [1]. Задать формулу для некоторой функции и её производной. На одном графике в логарифмическом масштабе (**loglog**) вывести модули разностей между точным значением производной в некоторой точке и правой и центральной разностной производной в зависимости от шага численного дифференцирования. Для генерации сеток, удобных для визуализации графиков в логарифмическом масштабе, рекомендуется использовать команду **logspace**.

ЛАБОРАТОРНАЯ РАБОТА №1

Введение в базовые операции

ВАРИАНТ 3

Во всех заданиях для генерирования случайных величин можно пользоваться **только** командами **rand** и **randn**. Все задания должны выполняться с минимальным использованием циклов. Каждое задание должно быть оформлено как блок в скрипте. В каждом задании, где нужно что-либо “проверить”, программа должна выводить адекватное сообщение об ошибке в случае невыполнения проверяемого условия.

Любое измерение скорости работы алгоритма должно производиться по большому количеству запусков алгоритма при фиксированных параметрах, с последующим усреднением результатов.

1 [0,5]. Задать два вещественных числа (a и b), натуральное число n и равномерную сетку на $[a, b]$ с n точками. Задать функцию $f(x) = |x| \cos(2x^2 - 1)$. Нарисовать график её значений на сетке, отметить на графике отдельно глобальные максимум и минимум, а соответствующие значения функции вывести в командное окно.

2 [0,5]. Запросить у пользователя ввод числа n . Проверить, что введенное число — простое.

1. Создать вектор из всех нечетных чисел, делящихся на 7, из промежутка от 1 до n .

2. Построить матрицу размера $n \times n$, все элементы i -й строки которой равны $i + 1$.

3. Создать матрицу B $(n + 1) \times (n + 1)$ вида

$$B = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Вытянуть матрицу B в вектор s . Присвоить переменной D последние 2 столбца матрицы B .

3 [0,5]. Создать матрицу размера 13×7 , состоящую из случайных элементов с нормальным распределением с параметрами $a = 5$, $\sigma^2 = 0.001$, найти элемент с максимальным модулем на диагонали этой матрицы, найти максимальное и минимальное отношение произведения к сумме для столбцов этой матрицы, отсортировать строки матрицы в обратном лексикографическом порядке (то есть строка $[a_1, a_2, a_3, \dots, a_n]$ стоит в матрице ниже строки $[b_1, b_2, b_3, \dots, b_n]$, если $a_i = b_i$ при $i = 1, \dots, k - 1$ и $a_k < b_k$ для некоторого k).

4 [0,5]. Предложить три способа создания матрицы A размера $(2n + 1) \times (2n + 1)$, где $n \geq 5$, следующего вида:

$$A = \{a_{ij}\}, a_{ij} = \begin{cases} 1, & i = 1 \text{ или } (2n + 1), j - \text{чётное}, \\ 1, & i - \text{чётное}, j = 1 \text{ или } (2n + 1), \\ 2, & (i, j) = \{(n, n), (n + 2, n), (n, n + 2), (n + 2, n + 2)\}, \\ -1, & (i, j) = \{(n + 1, n + 1)\}, \\ 0, & \text{иначе.} \end{cases}$$

5 [0,5]. Задан массив $2 \times n$ координат точек на плоскости. Построить матрицу $A \in \mathbb{R}^{n \times n}$, в позиции (i, j) которой будет стоять псевдоскалярное произведение i -го и j -го вектора $(x \bullet y = x_1 y_2 - x_2 y_1)$.

6 [0,5]. Даны векторы a размерности n и b размерности m . Найти, используя только арифметические операции и команды **max** и **min**, максимум функции $|a_i - b_j|$, где a_i — элемент вектора a , b_j — элемент вектора b . Функцию **abs** и дополнительную память не использовать.

7 [0,5]. В каждом столбце матрицы X есть ненулевой элемент. Найти порядковые номера (в столбце) и значения всех первых ненулевых элементов каждого столбца.

8 [0,5]. Пусть у нас задано n точек в пространстве \mathbb{R}^k в виде матрицы **double**[n, k]. Требуется построить матрицу **double**[n, n] расстояний между каждой парой точек. Пользоваться командами **pdist** и **squareform** нельзя.

9 [0,5]. Реализовать функцию $c = \text{my_prod}(x, y)$, которая выполняет скалярное умножение векторов $c = \langle x, y \rangle$ по определению (через цикл). Сравнить быстродействие этой функции, команды **x*y** и команды **dot** для векторов различной размерности. Построить график времени работы.

10 [0,5]. Применяя функцию **Matlab ismember**, реализовать ее версию с ключом **'rows'** для матрицы с неотрицательными целочисленными элементами (можно использовать функцию **ismember** без ключа и функцию **sub2ind**).

11 [1]. Сгенерировать вектор x из n случайных величин с нормальным распределением $N(a, \sigma^2)$. Проверить неравенство Чебышёва: для заданного числа b вывести долю элементов x_i таких, что $|x_i - a| > b$, и сравнить с числом σ^2/b^2 .

12 [2]. По аналогии с функцией **trapz** реализовать аналогичные функции **rectangles** (интегрирование методом прямоугольников) и **simpson** (методом Симпсона). С помощью них построить график первообразной функции $f(x) = \cos(x^2)$. Сравнить внутреннюю скорость сходимости при использовании всех трёх методов (внутренняя скорость сходимости определяется с помощью сравнения разностей решений при шаге h и $h/2$, нарисовать график этой ошибки в зависимости от h). Сравнить время вычисления.

13 [1]. Задать формулу для некоторой функции и её производной. На одном графике в логарифмическом масштабе (**loglog**) вывести модули разностей между точным значением производной в некоторой точке и правой и центральной разностной производной в зависимости от шага численного дифференцирования. Для генерации сеток, удобных для визуализации графиков в логарифмическом масштабе, рекомендуется использовать команду **logspace**.