

Лабораторная работа 8. Работа с файлами, произвольное число аргументов в функциях

Работа с файлами

Задание 1

Создать три матрицы A1, A2, A3 порядка 2, 3 и 4, каждая из которых является магическим квадратом.

- 1) Записать матрицы в двоичный файл, загрузить первые две из файла.
- 2) Записать матрицу A3 в файл excel. Считать данные из файла в матрицу B.
- 3) Записать матрицу A2 в текстовый файл. Считать данные из файла в матрицу C.

Задание 2

Записать в текстовый файл таблицу значений векторов x , $\sin(x)$, $\cos(x)$, где $x=[-\pi, \pi]$ с шагом 0.01. Использовать форматированную запись в файл, первая строка должна содержать заголовки столбцов. Импортировать данные из файла (использовать два способа: `importdata` и `textscan`). По полученным данным построить графики в одних осях. Вывести заголовок и легенду.

Задание 3

Записать в текстовый файл таблицу значений векторов x , $\sin(x)$, $\cos(x)$, где $x=[-\pi, \pi]$ с шагом 0.01. Использовать форматированную запись в файл в виде: "x=...; y1=...; y2=.." на каждой строке. Считать данные из файла. По полученным данным построить графики один рядом с другим. Вывести название каждого графика и общий заголовок.

Задание 4

Задать случайную квадратную матрицу A 10 порядка и случайный вектор b. Найти решение СЛАУ $Ax=b$. Записать решение в текстовый файл по три числа в каждой строке. Считать данные из файла в вектор-строку (использовать форматированное чтение из файла, задав размер выходного массива `Inf`).

Произвольное число аргументов в функциях

Каждое задание – отдельная функция (или несколько функций). Вызов всех функций организуйте в основном файле.

Задание 5

Написать функцию `Sum`, которая находит сумму произвольного количества аргументов (использовать `varargin`), а также выводит их количество. Проанализировать результаты, вызвав функцию следующим образом:

```
res1=Sum(1,2,3,4,5)
res2=Sum(10, 'a')%посмотреть ascii-код символа
res3=Sum(10, 'ab')
res4=Sum('abc', 'def')
res5=Sum(1,2,3,4,5, [10 20 30])
res6=Sum(1,2,3,4,5, [10;20;30])
res7=Sum(1,2,3,4,5, [10 20;30 40])
```

Написать выводы в комментариях к каждому вызову.

Задание 6

Написать функцию `MaxMinVector`, которая принимает произвольное количество (использовать `varargin`) векторов и возвращает два выходных параметра: максимальный и минимальный элементы среди элементов всех векторов.

Указание. Организовать проверку, что все входные параметры являются векторами (функция `isvector`). В случае, если хотя бы один входной параметр не является вектором, функция `MaxMinVector` должна выдавать сообщение об ошибке (команда **error**) с указанием, какой из входных параметров не является вектор (см. `help` по команде `error` для создания форматированного сообщения об ошибке).

Задание 7

Написать функцию `MaxMinMatrix`, которая принимает произвольное количество (использовать `varargin`) матриц и возвращает два выходных параметра: максимальный и минимальный элементы среди элементов всех матриц.

Задание 8

Написать функцию `MinMaxAverageV`, которая принимает на вход вектор и возвращает от одного до трех выходных параметров (использовать `varargout`): первый параметр – минимальный элемент, второй параметр – максимальный элемент, третий параметр – среднее значение всех элементов. Вызвать функцию с разным количеством выходных параметров (в том числе и больше трех) и проанализировать результат. Будет ли выводиться ошибка, если количество выходных параметров больше трех?

Указание. Функция должна проверять, что входной параметр является вектором.

Задание 9

Написать функцию `AnalyseV`, которая принимает на вход вектор (с проверкой, является ли аргумент вектором) и возвращает:

- среднее значение, если вызывается с одним выходным параметром или без указания выходных параметров;
- минимум и максимум, если вызывается с двумя выходными параметрами.

В случае, если функция вызывается с тремя и более выходными параметрами, выдавать сообщение об ошибке: «Слишком много выходных параметров».