

Московский государственный технический университет
имени Н.Э. Баумана



Учебный практикум на ЭВМ

Лекция 1

Основы синтаксиса MathWorks MATLAB



Ассистент кафедры «Системы автоматического управления»
Завьялов Р.А.

Осенний семестр 2022

Дисциплина: Профилирующая практика

Итог: оценка

Преподаватель: Завьялов Роман Александрович

Почта: zavialov.ra@gmail.com





– ЭТО:

- Система научных вычислений для инженерных расчётов
- Высокоуровневый интерпретируемый язык программирования
- Богатый набор функций и приложений для моделирования и решения математических задач



- Свободное распространение программы
- Функциональная схожесть с Matlab



- Менее богатая библиотека функций и приложений (в сравнении с Matlab)



Maple™



- Развиты графические средства
- Простота изучения



- Синтаксис языка напоминает Pascal
- Ограниченная интеграция с языками программирования



- Широкий диапазон функциональных возможностей
- Концепция бесконечной точности вычислений
- Простой в изучении



- Специфический синтаксис языка программирования
- Низкое быстродействие

ptc mathcad[®]



- Отличный редактор математических текстов
- Возможность символьных вычислений
- Твои вычисления = твой отчёт



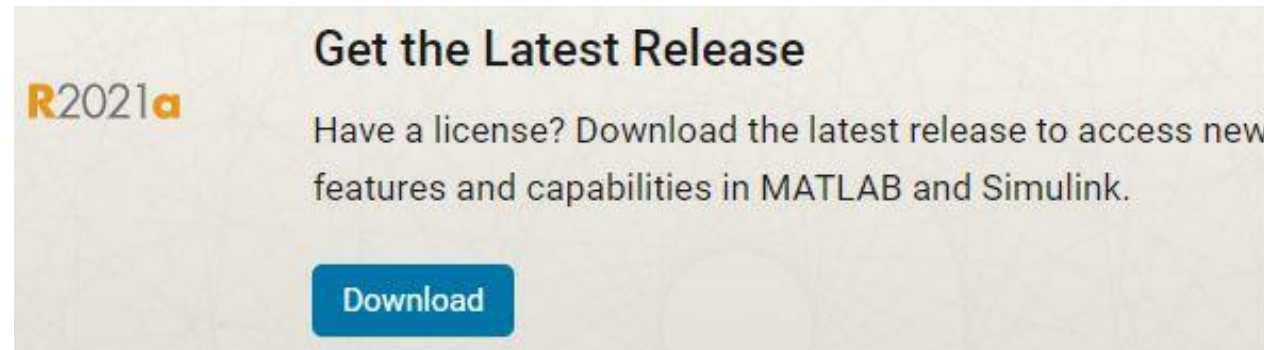
- Именно редактор, не язык программирования



Шаг 1: переходим на сайт <https://www.mathworks.com/>



Шаг 2: скачиваем последнюю версию программы



Шаг 3: создаём и регистрируем аккаунт на почту @bmstu



Email

[No account? Create one!](#)

By signing in you agree to our [privacy policy](#).

Next

Шаг 4: подтверждаем аккаунт и приступаем к установке

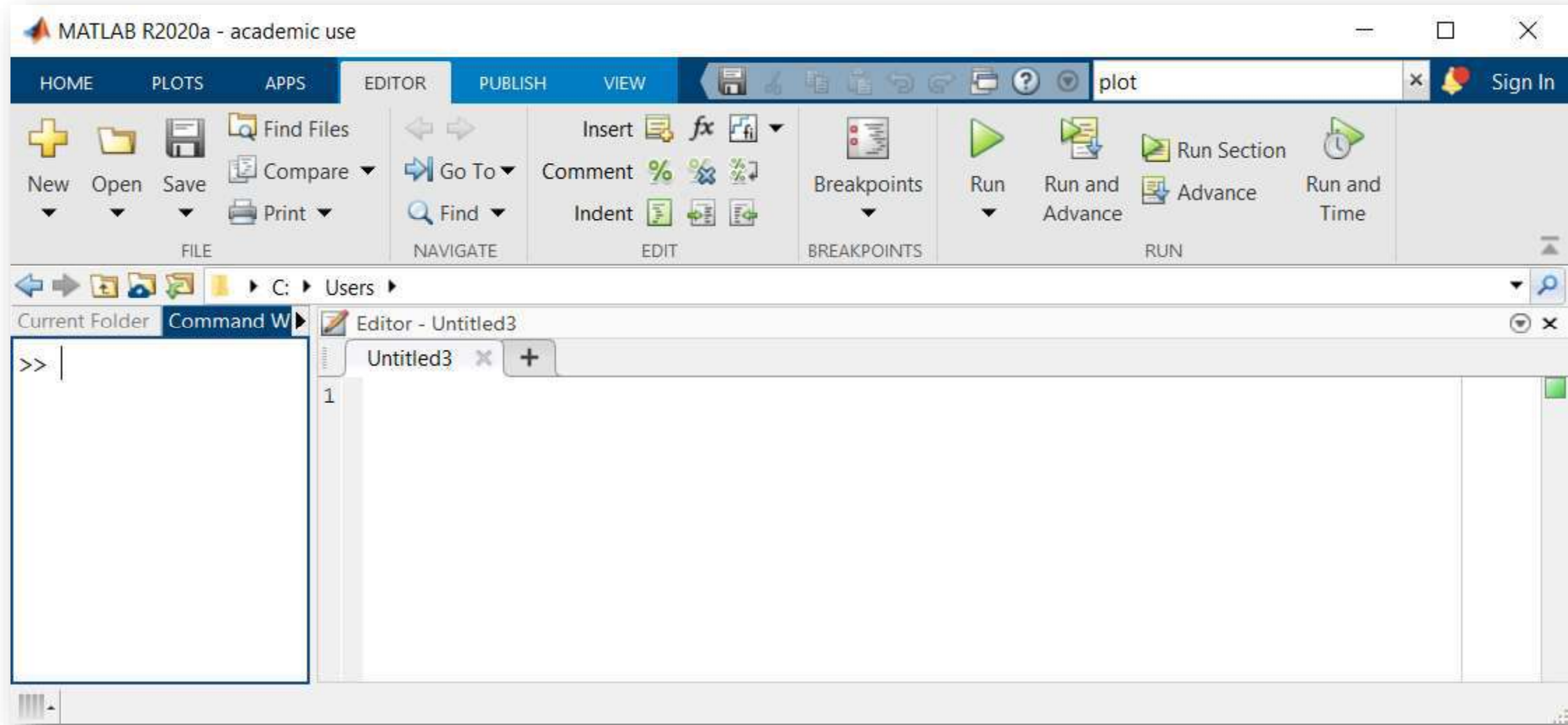
При установке рекомендуется выбрать следующие расширения:

- | | | | |
|---------------------------|-------------------------------|---|---|
| ✓ MATLAB | ✓ Curve Fitting Toolbox | ✓ Navigation Toolbox | ✓ Simspace Driveline |
| ✓ Simulink | ✓ Data Acquisition Toolbox | ✓ Optimization Toolbox | ✓ Simscape Electrical |
| ✓ Aerospace Blockset | ✓ DSP System Toolbox | ✓ Parallel Computing Toolbox | ✓ Simscape Multibody |
| ✓ Aerospace Toolbox | ✓ Global Optimization Toolbox | ✓ Parallel Computing Toolbox | ✓ Simulink Compiler |
| ✓ Audio Toolbox | ✓ Image Acquisition Toolbox | ✓ Partial Differential Equation Toolbox | ✓ Simulink Control Design |
| ✓ Communications Toolbox | ✓ Image Processing Toolbox | ✓ Robust Control Toolbox | ✓ Statistics and Machine Learning Toolbox |
| ✓ Computer Vision Toolbox | ✓ MATLAB Compiler | ✓ Sensor Fusion and Tracking Toolbox | ✓ Symbolic Math Toolbox |
| ✓ Control Systems Toolbox | ✓ Mixed-Signal Blockset | ✓ Signal Processing Toolbox | ✓ System Identification Toolbox |
| | ✓ Model-Predictive Control | ✓ SimEvents | |
| | | ✓ Simspace | |



ПРАКТИКУМ

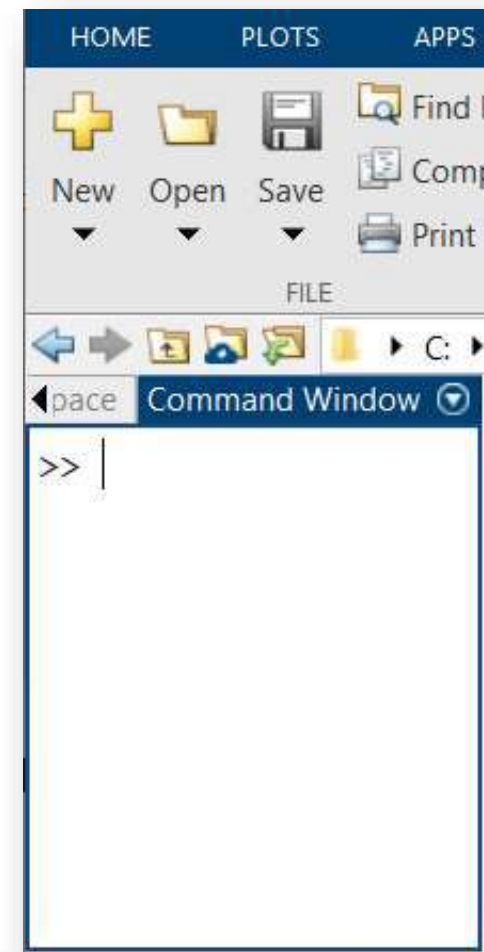
Запуск MATLAB



Описание пользовательского интерфейса MathWorks MATLAB

Command Window

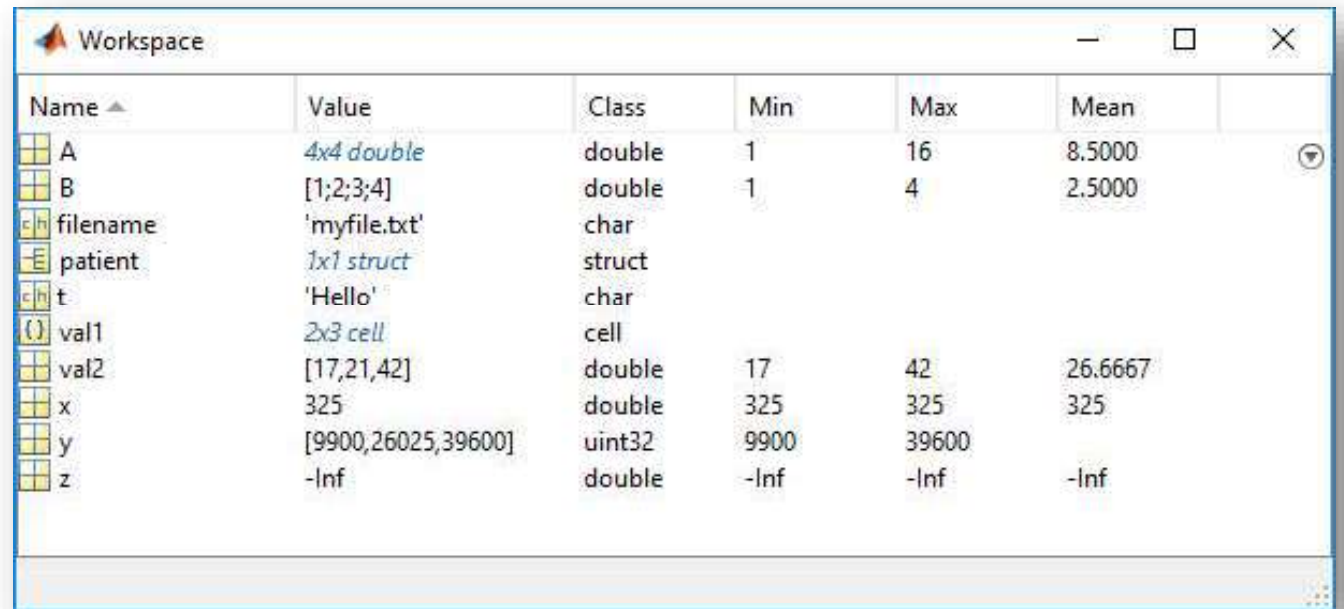
Командное окно позволяет ввести отдельные операторы в командной строке и просмотреть сгенерированные результаты.



Описание пользовательского интерфейса MathWorks MATLAB

Workspace

Позволяет просмотреть и
в интерактивном режиме
и управлять содержимым
рабочей области в
MATLAB.



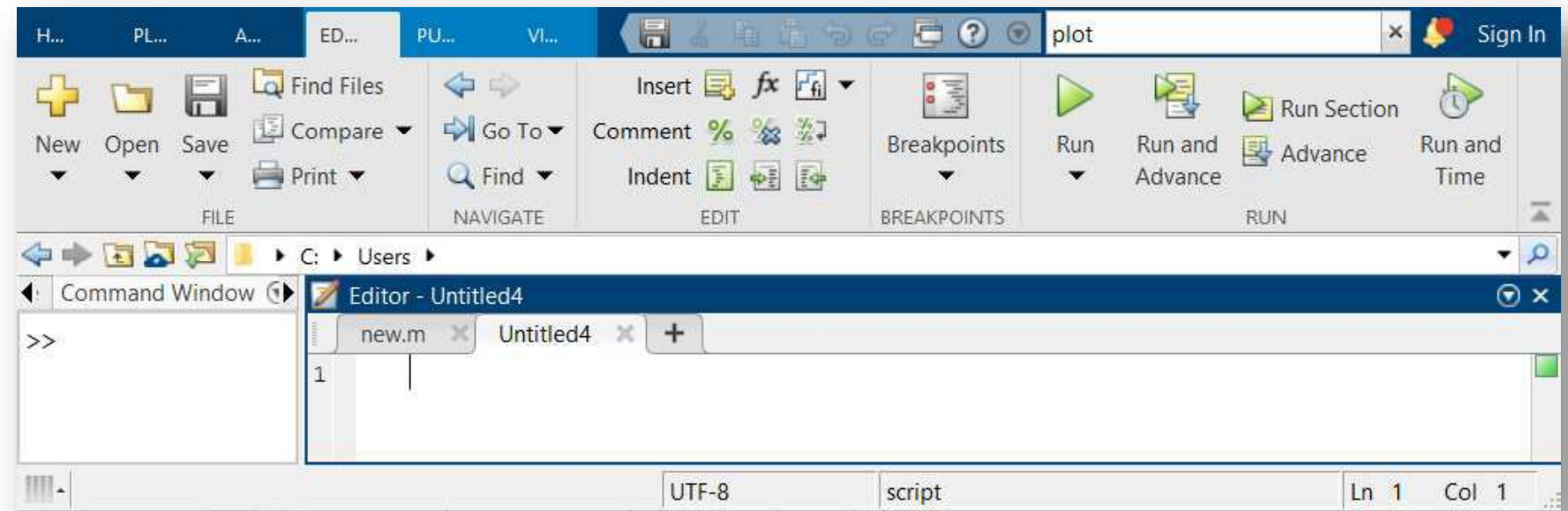
The screenshot shows the MATLAB Workspace window. It contains a table with the following data:

Name	Value	Class	Min	Max	Mean
A	4x4 double	double	1	16	8.5000
B	[1;2;3;4]	double	-1	4	2.5000
filename	'myfile.txt'	char			
patient	1x1 struct	struct			
t	'Hello'	char			
val1	2x3 cell	cell			
val2	[17,21,42]	double	17	42	26.6667
x	325	double	325	325	325
y	[9900,26025,39600]	uint32	9900	39600	
z	-Inf	double	-Inf	-Inf	-Inf

Описание пользовательского интерфейса MathWorks MATLAB

Editor

Здесь можно
вводить команды
и выполнять их
как сценарий.

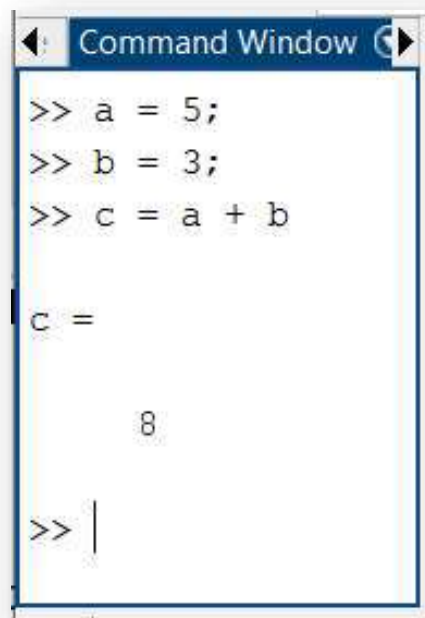


Описание пользовательского интерфейса MathWorks MATLAB

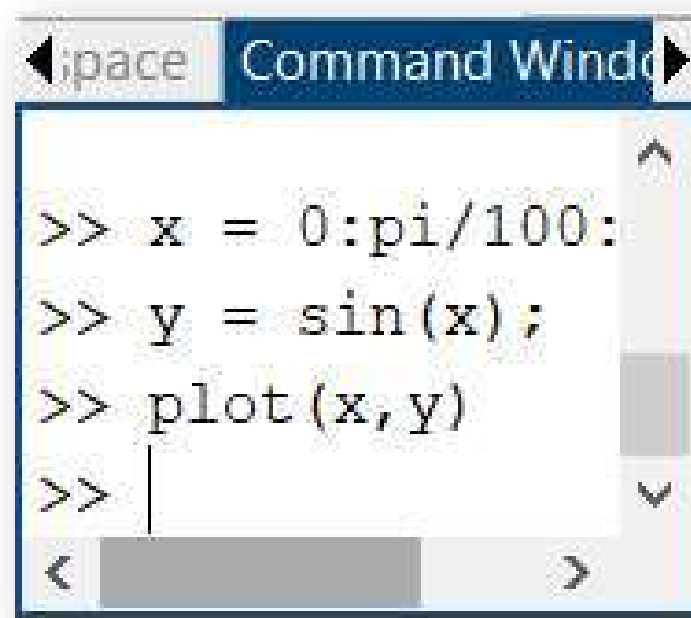
- **clc** - очищает весь текст в командном окне;
- **clear all** - удаление элементов из рабочей области, удаление данных из переменных;
- **clear variables** - удаление переменных из рабочей области;
- **close all force** - закрывает все фигуры.

Выполнение простого кода в командной строке

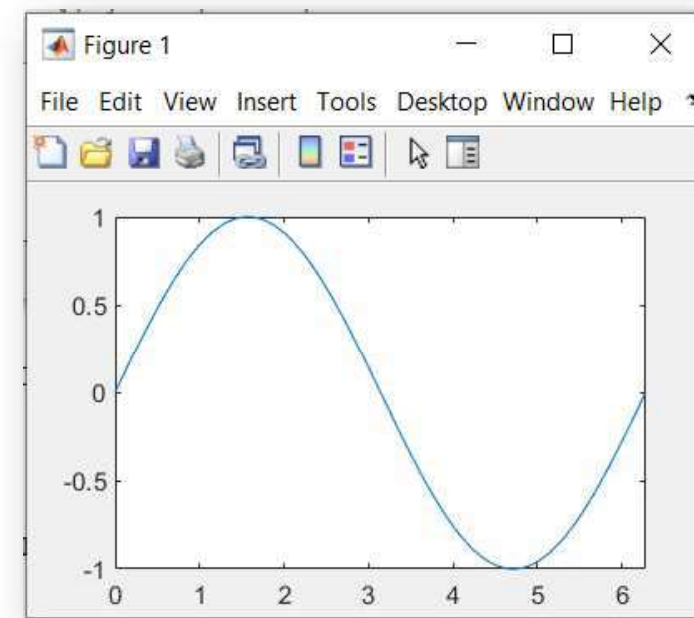
Command Window -Позволяет быстро и легко выполнить необходимую проверку



```
>> a = 5;  
>> b = 3;  
>> c = a + b  
  
c =  
  
      8  
  
>> |
```

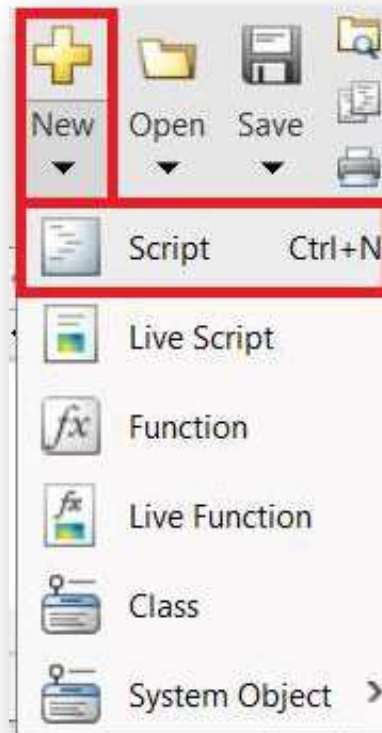


```
>> x = 0:pi/100:  
>> y = sin(x);  
>> plot(x,y)  
>> |
```

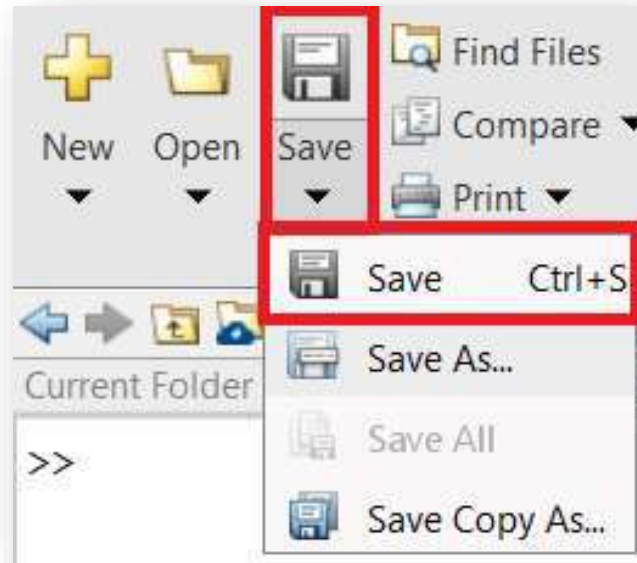


Создание отдельного скрипта и выполнение простого кода

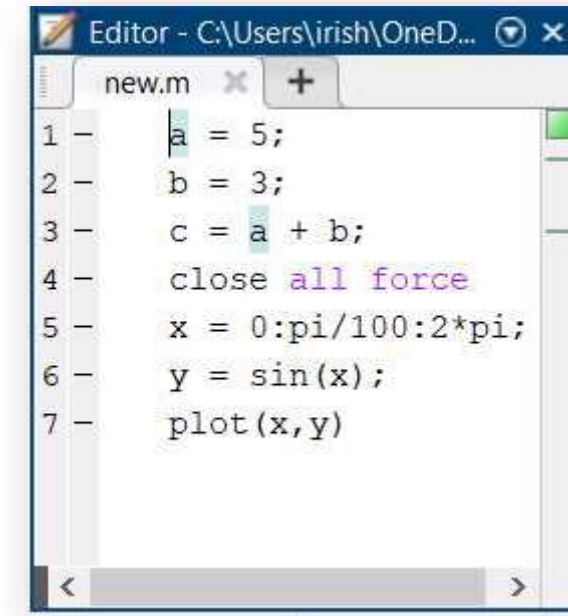
Создать



Сохранить

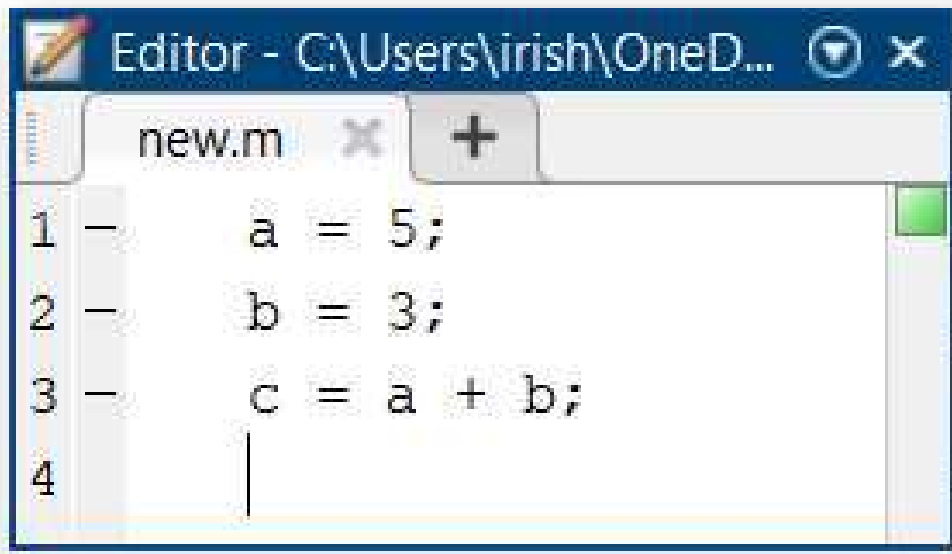


Написать программу



Задание скаляров, векторов и матриц заданным набором значений

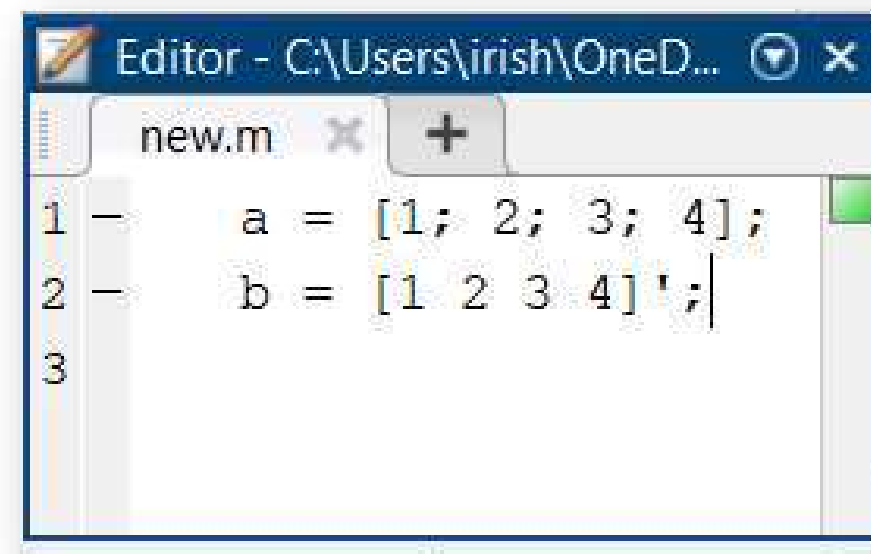
Скаляр



The image shows a MATLAB Editor window titled "Editor - C:\Users\irish\OneD...". The window contains a file named "new.m" with the following code:

```
1 — a = 5;  
2 — b = 3;  
3 — c = a + b;  
4 |
```

Вектор-столбец

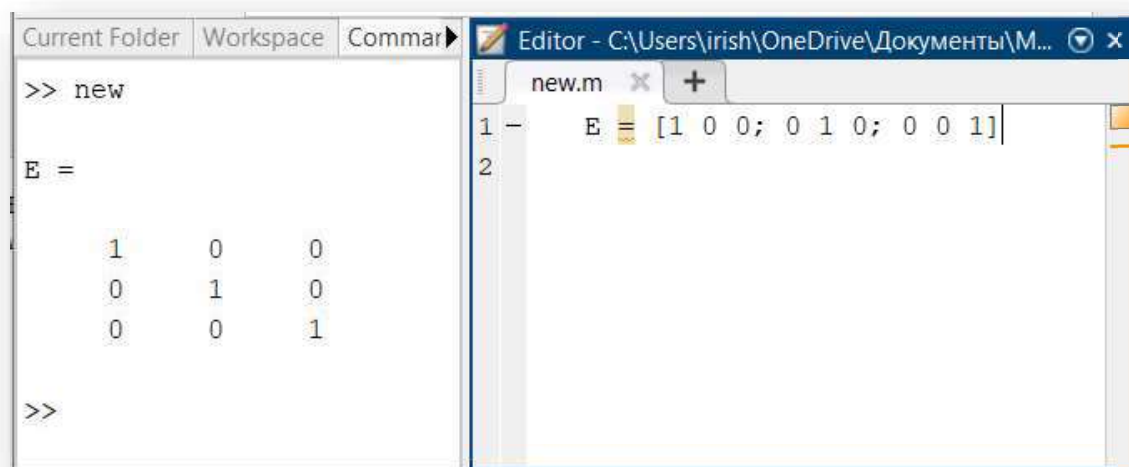


The image shows a MATLAB Editor window titled "Editor - C:\Users\irish\OneD...". The window contains a file named "new.m" with the following code:

```
1 — a = [1; 2; 3; 4];  
2 — b = [1 2 3 4]';  
3 |
```

Задание скаляров, векторов и матриц заданным набором значений

Единичная матрица 3x3



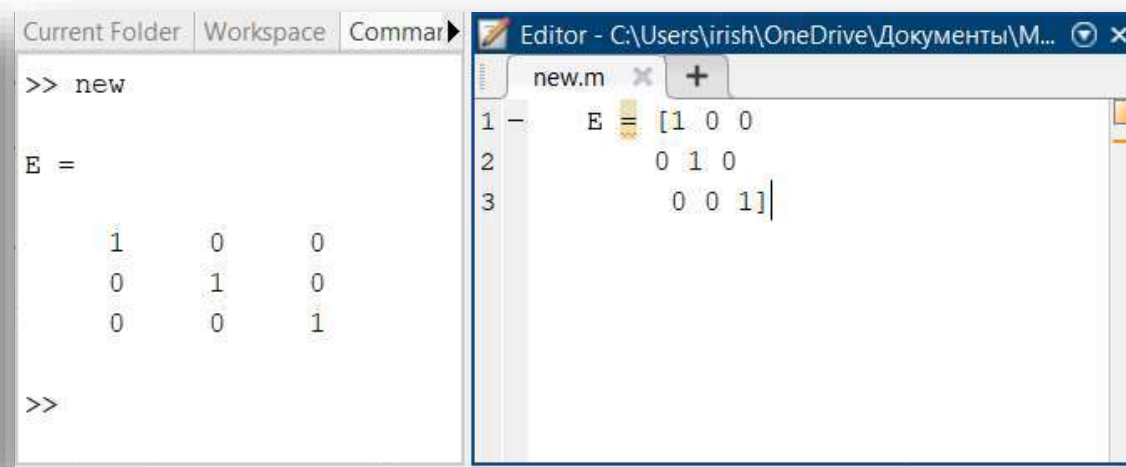
The image shows two windows from the MATLAB environment. The left window is the Command Window, and the right window is the Editor. Both show the creation of a 3x3 identity matrix E .

```
>> new  
  
E =  
  
     1     0     0  
     0     1     0  
     0     0     1  
  
>>
```

The Editor window shows the file `new.m` with the following code:

```
1 - E = [1 0 0; 0 1 0; 0 0 1]  
2
```

Единичная матрица



The image shows two windows from the MATLAB environment. The left window is the Command Window, and the right window is the Editor. Both show the creation of a 3x3 identity matrix E .

```
>> new  
  
E =  
  
     1     0     0  
     0     1     0  
     0     0     1  
  
>>
```

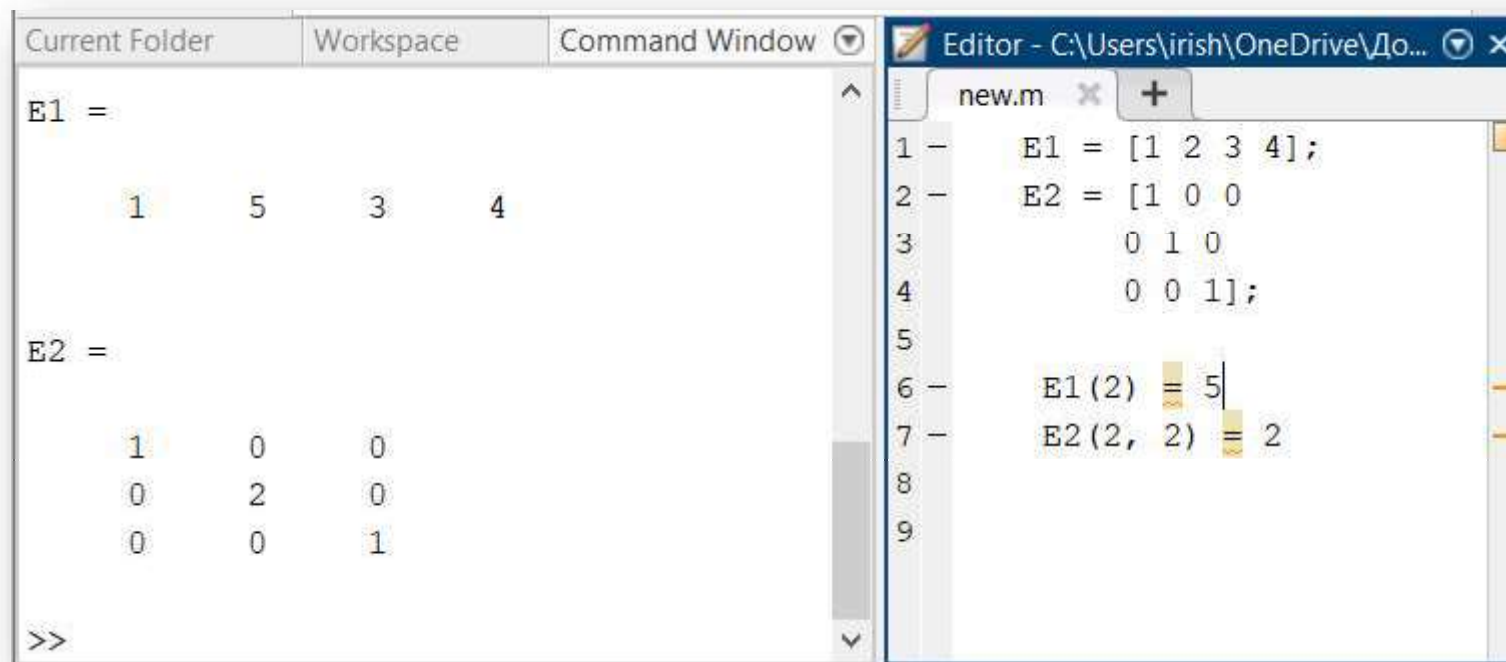
The Editor window shows the file `new.m` with the following code:

```
1 - E = [1 0 0  
2       0 1 0  
3       0 0 1]
```

Обращение к элементам векторов и матриц

Для обращения к элементу вектора/матрицы, в круглых скобках указывается номер элемента к которому обращаются.

После знака равенства указывается число на которое меняется элемент матрицы/вектора



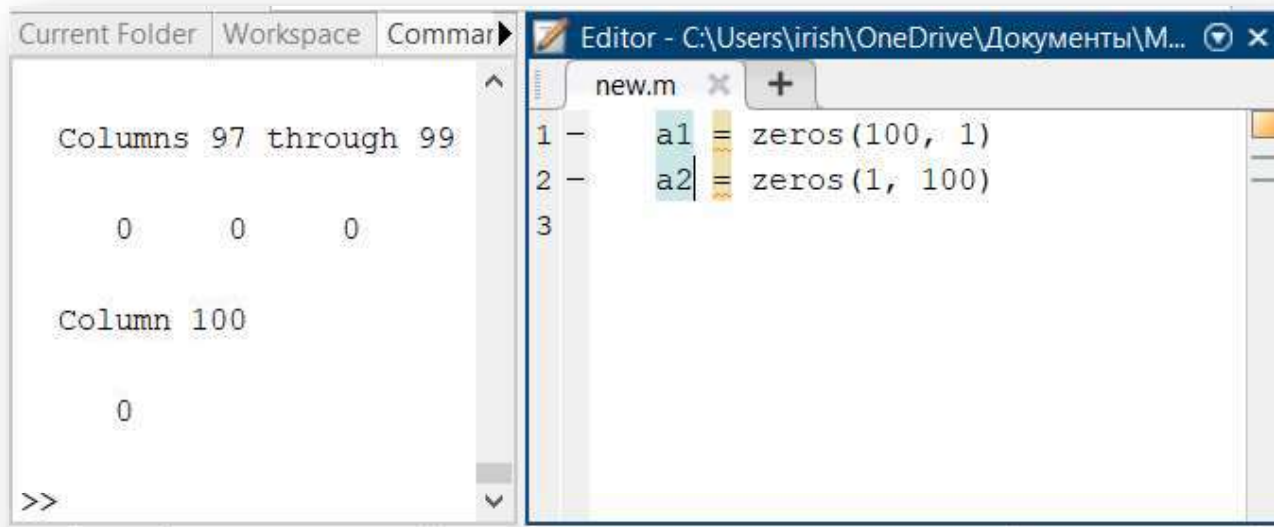
The screenshot displays the MATLAB environment. The 'Workspace' window on the left shows two variables: `E1` and `E2`. `E1` is a 1x4 row vector with values [1, 5, 3, 4], and `E2` is a 3x3 matrix with values [1, 0, 0; 0, 2, 0; 0, 0, 1]. The 'Editor' window on the right shows a script named `new.m` with the following code:

```
1 E1 = [1 2 3 4];  
2 E2 = [1 0 0  
3       0 1 0  
4       0 0 1];  
5  
6 E1(2) = 5;  
7 E2(2, 2) = 2;  
8  
9
```

Задание скаляров, векторов и матриц с использованием `zeros()`, `ones()`, `nan()`, `inf()`

`zeros()`

Вектор-столбец и вектор-строка, 3

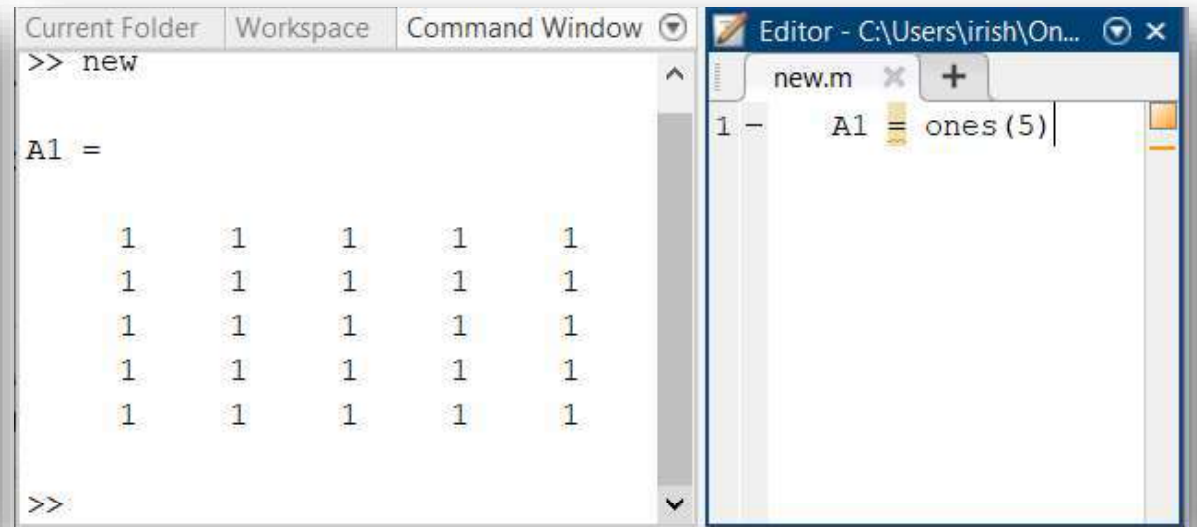


The screenshot shows the MATLAB environment. The Command Window on the left displays the output of the code: three zeros for columns 97-99 and one zero for column 100. The Editor window shows the code in `new.m`:

```
1 - a1 = zeros(100, 1)
2 - a2 = zeros(1, 100)
3
```

`ones()`

Матрица 5x5, состоящая из единиц



The screenshot shows the MATLAB environment. The Command Window on the left displays the output of the code: a 5x5 matrix of ones. The Editor window shows the code in `new.m`:

```
>> new

A1 =

     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1

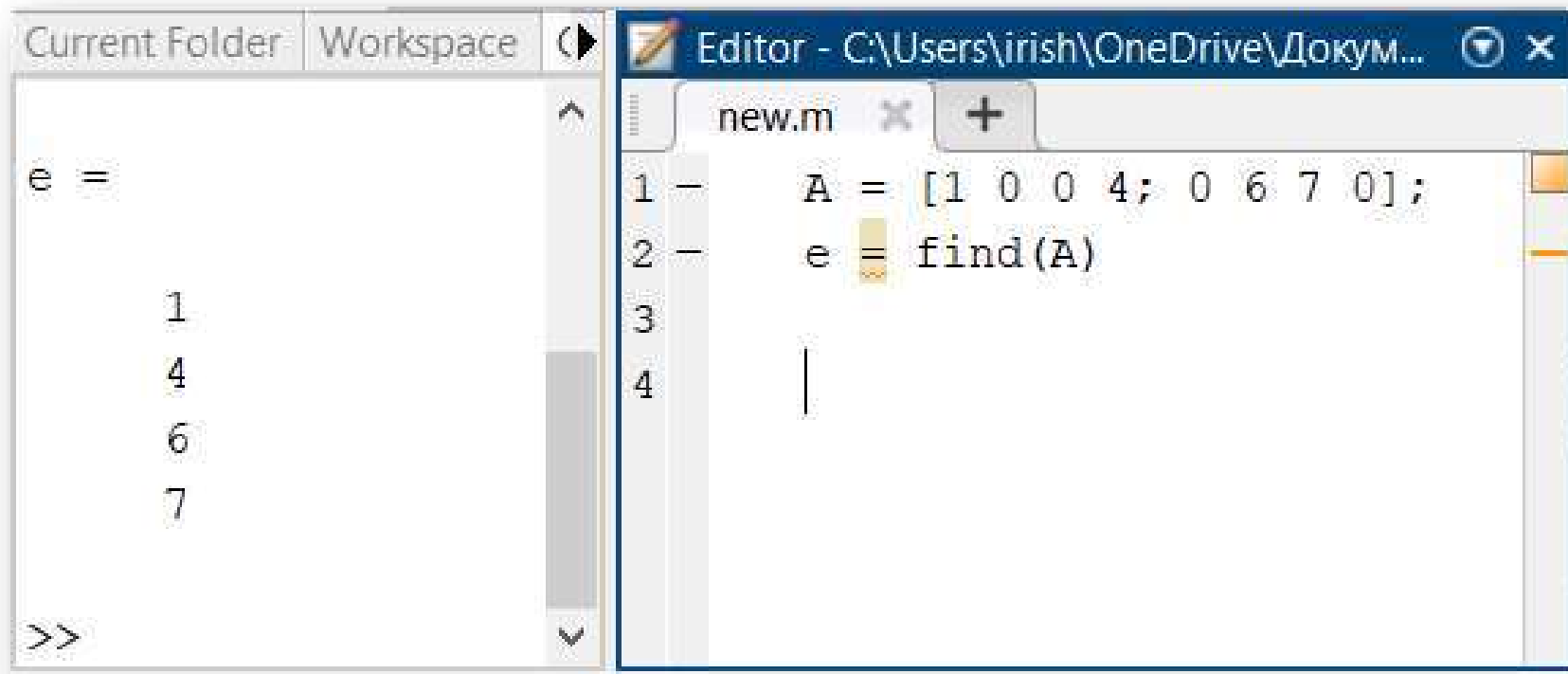
>>
```

The Editor window shows the code in `new.m`:

```
1 - A1 = ones(5)
```

Использование функций find() и nnz()

Функция find() в качестве аргумента принимает некоторое условие, а возвращает набор индексов неравных нулю элементов массива.



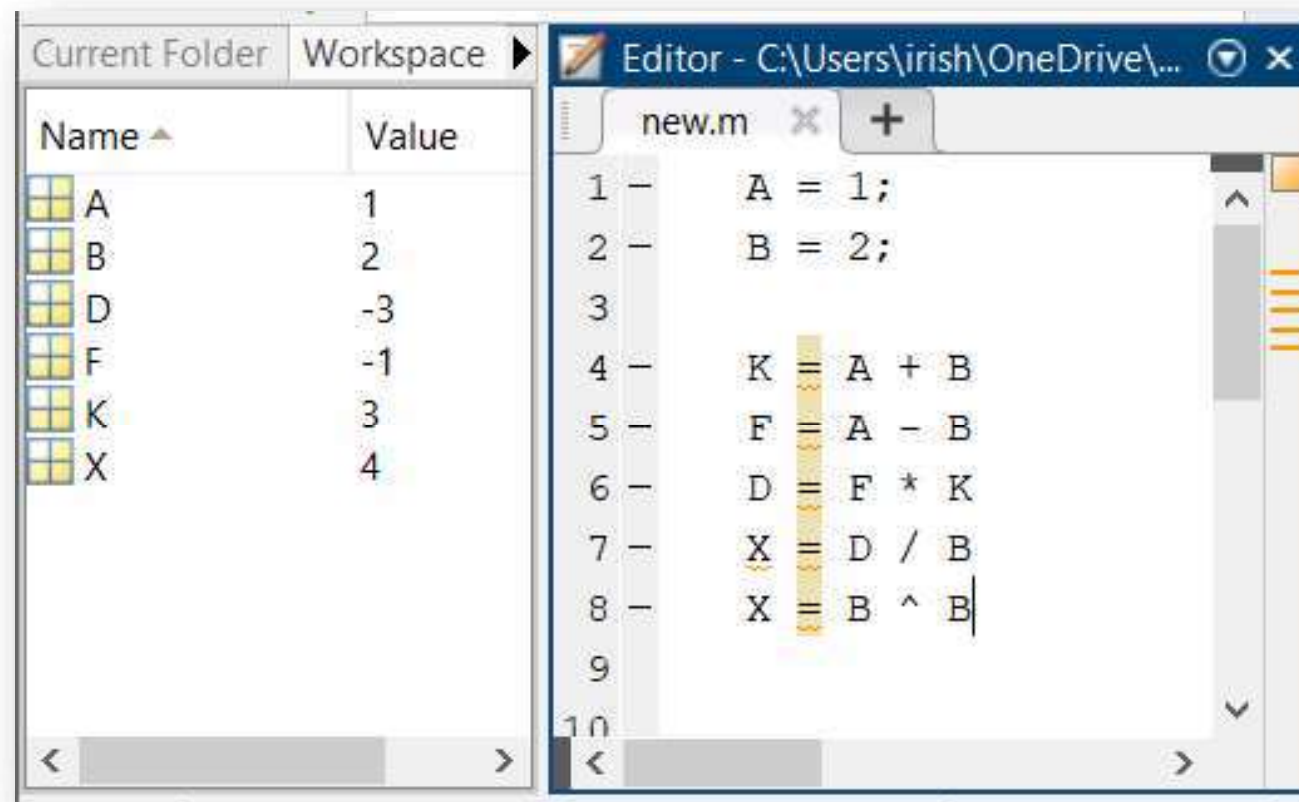
The image shows a MATLAB interface with two main windows. On the left is the 'Workspace' window, which displays the variable 'e' as a column vector containing the values 1, 4, 6, and 7. On the right is the 'Editor' window, which shows a script named 'new.m'. The script contains two lines of code: 'A = [1 0 0 4; 0 6 7 0];' and 'e = find(A)'. The 'find' function is highlighted in yellow in the original image. The 'Current Folder' window is also visible at the top left of the interface.

```
e =  
  
    1  
    4  
    6  
    7  
  
>>
```

```
new.m  
1 - A = [1 0 0 4; 0 6 7 0];  
2 - e = find(A)  
3  
4 |
```

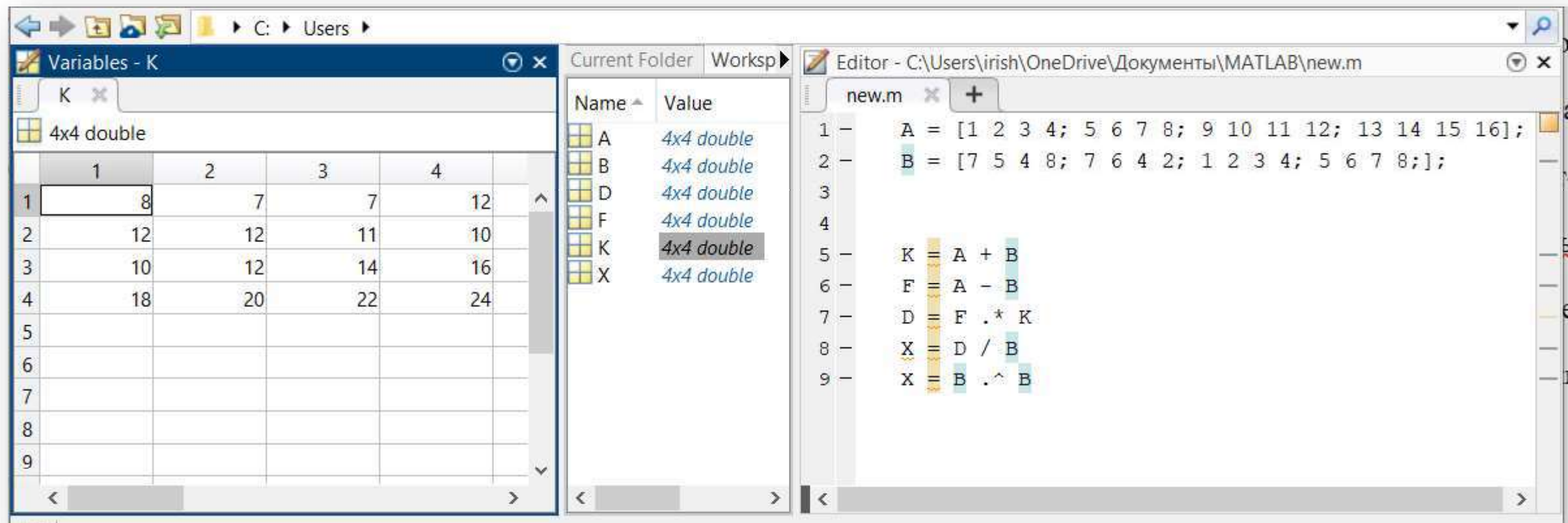

Арифметические операции над скалярами, векторами и матрицами

Арифметические операции над скалярами.



Арифметические операции над скалярами, векторами и матрицами

Арифметические операции над матрицами



The image displays the MATLAB environment with three main windows:

- Variables - K:** A window showing the workspace variables. It lists variables A, B, D, F, K, and X, all of type *4x4 double*. The variable K is highlighted.
- Current Folder / Workspace:** A window showing the current folder and workspace. It lists variables A, B, D, F, K, and X, all of type *4x4 double*. The variable K is highlighted.
- Editor - C:\Users\irish\OneDrive\Документы\MATLAB\new.m:** A window showing the MATLAB script. The script defines matrices A and B, and performs arithmetic operations on them:

```
1 A = [1 2 3 4; 5 6 7 8; 9 10 11 12; 13 14 15 16];  
2 B = [7 5 4 8; 7 6 4 2; 1 2 3 4; 5 6 7 8];  
3  
4  
5 K = A + B  
6 F = A - B  
7 D = F .* K  
8 X = D / B  
9 X = B .^ B
```

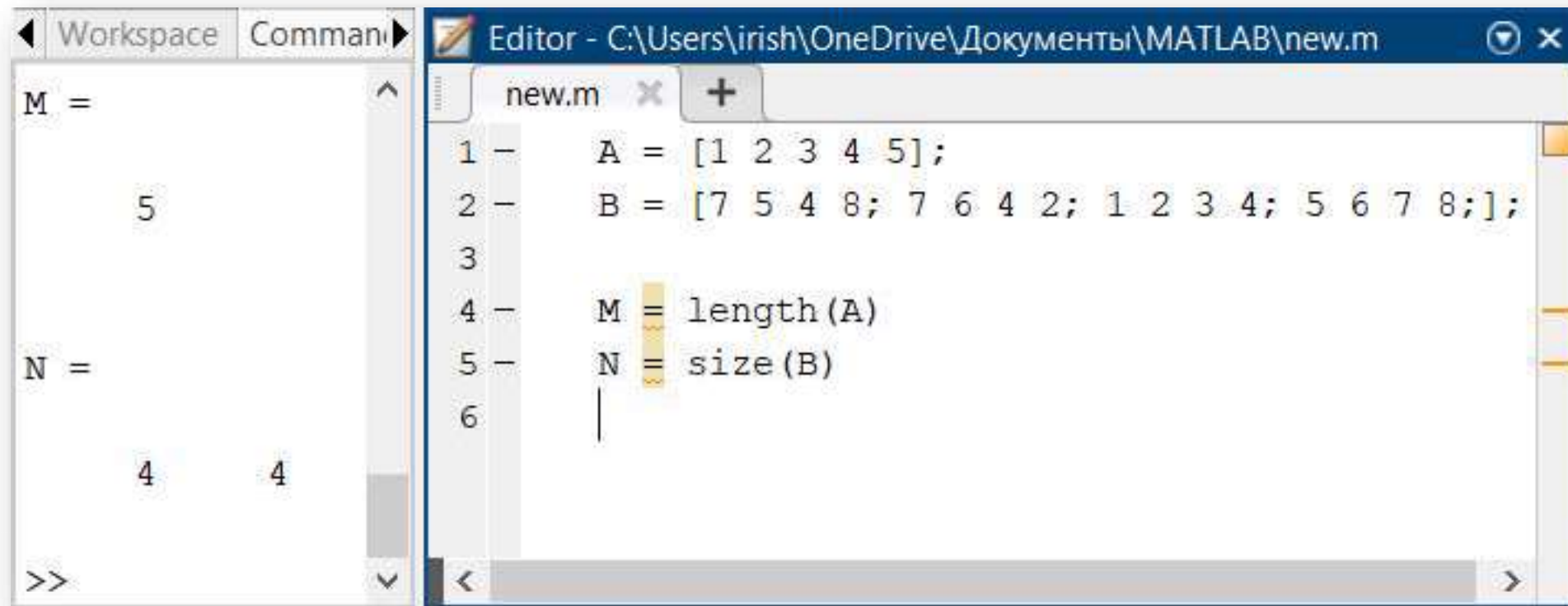
The results of these operations are displayed in the Variables - K window:

	1	2	3	4
1	8	7	7	12
2	12	12	11	10
3	10	12	14	16
4	18	20	22	24
5				
6				
7				
8				
9				

Определение размеров векторов и матриц через `length()` и `size()`

`length()` – возвращает количество элементов в векторе

`size()` – возвращает размер матрицы

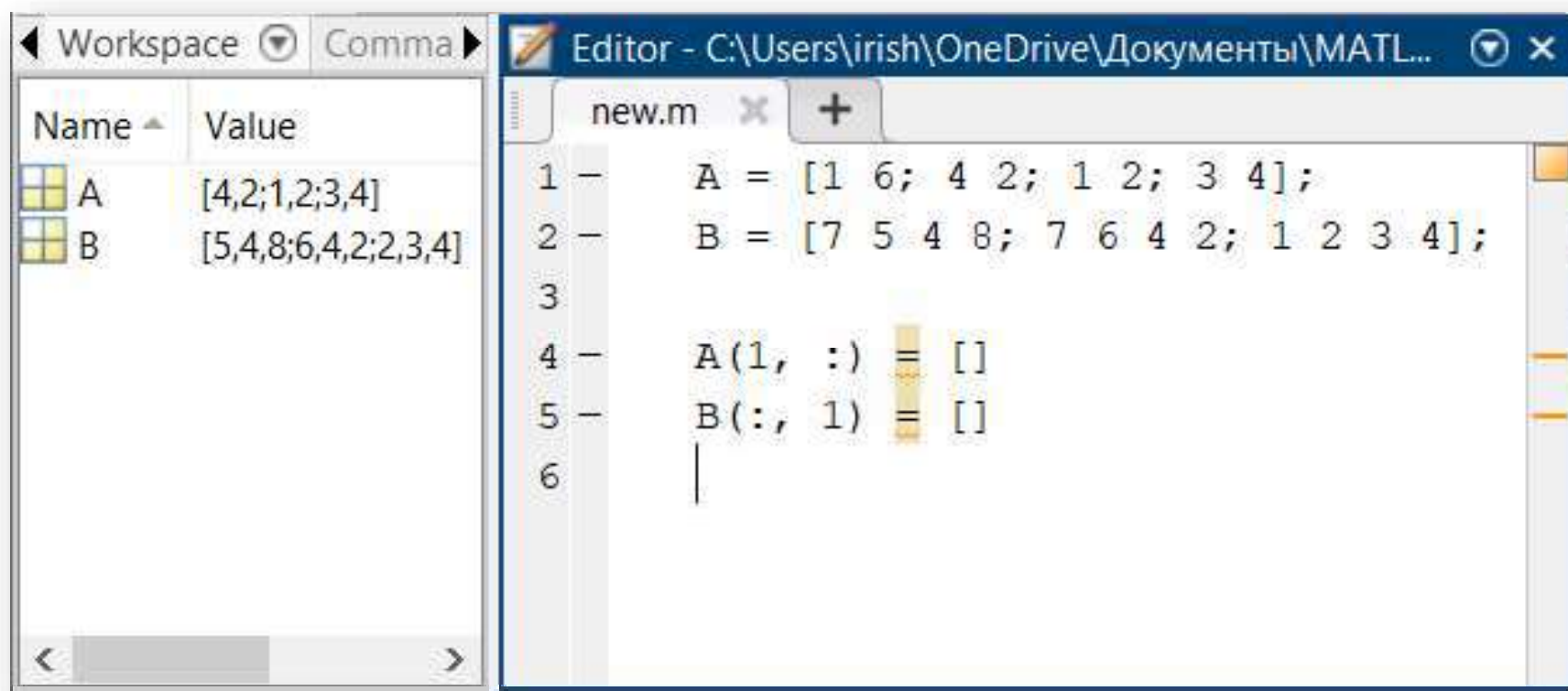


The image shows a MATLAB interface with two windows. The 'Workspace' window on the left displays the results of the code: variable 'M' has a value of 5, and variable 'N' has a value of [4 4]. The 'Editor' window on the right shows the code being executed in a file named 'new.m'.

```
1 - A = [1 2 3 4 5];  
2 - B = [7 5 4 8; 7 6 4 2; 1 2 3 4; 5 6 7 8];  
3  
4 - M = length(A)  
5 - N = size(B)  
6 |
```

Удаление строк и столбцов из матриц

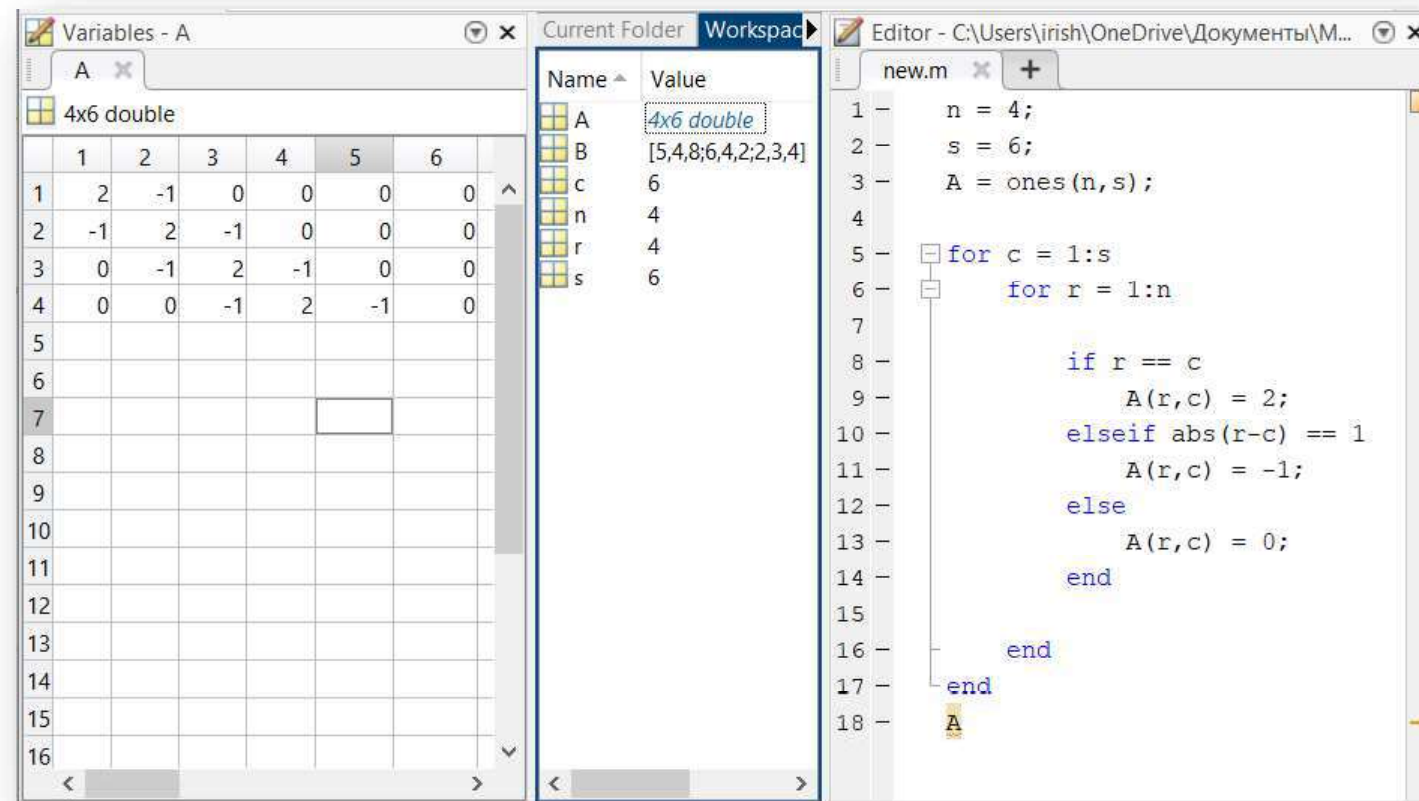
Для удаления строки или столбца из матрицы устанавливается пара квадратных скобок



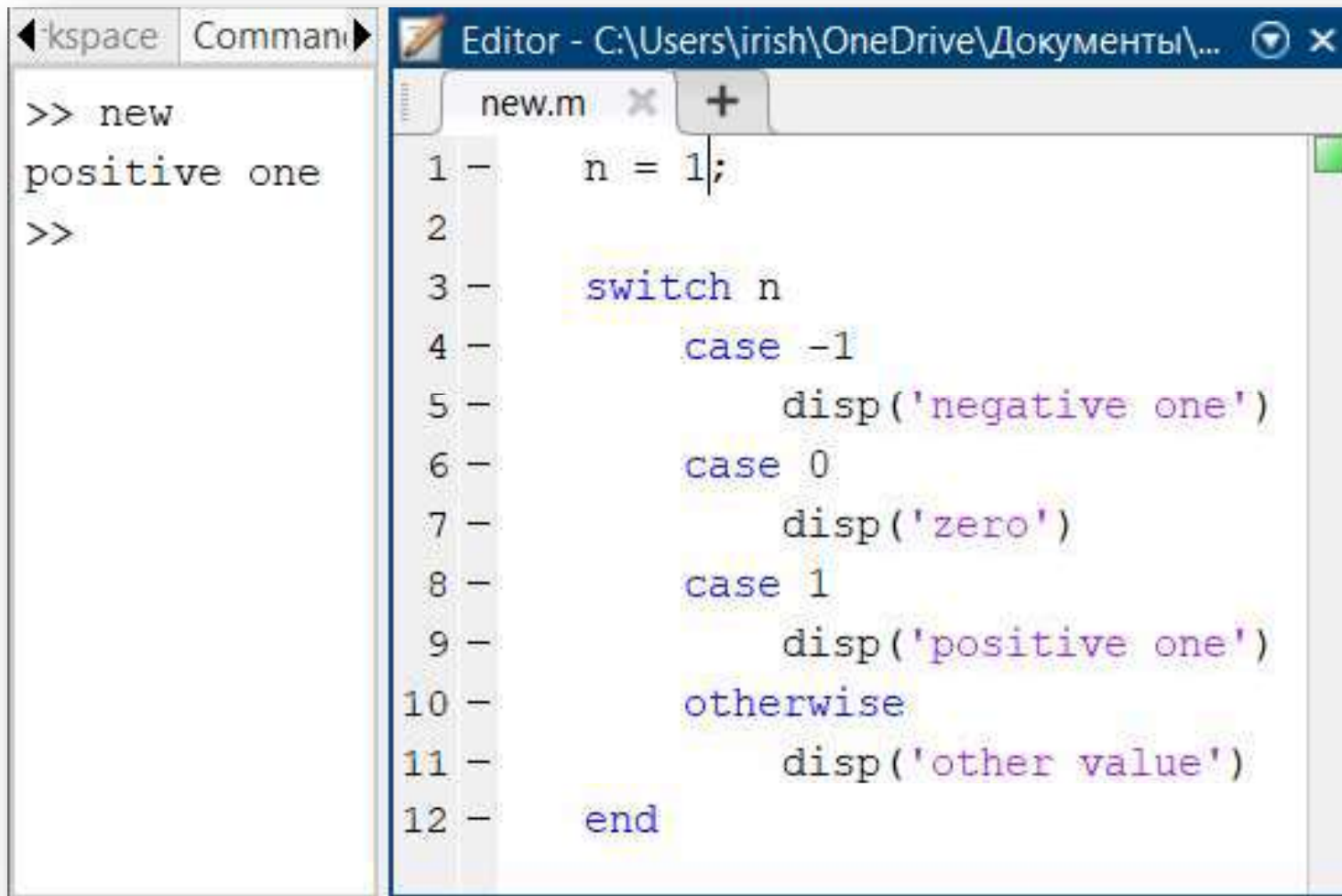
Условные операторы if/else, if/elseif/elseif/else

Оставить просто без циклов и матриц

Дать список $>$, $<$ $=$ $'$ $=$



Оператор множественного выбора switch/case

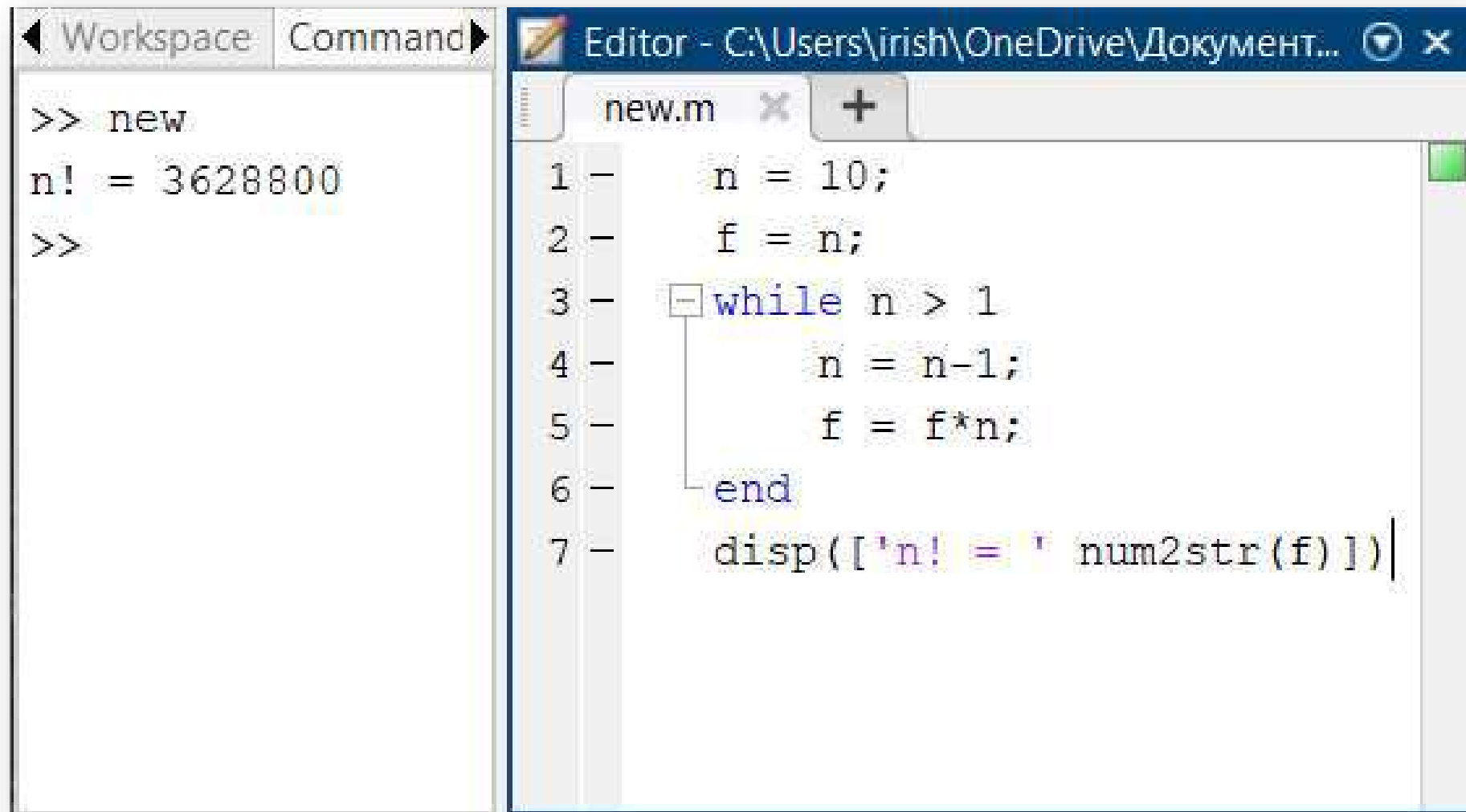


The image shows a MATLAB environment with a Command Window on the left and an Editor on the right. The Command Window shows the execution of a script named 'new'. The Editor shows the code for 'new.m', which uses a switch/case statement to handle different values of 'n'.

```
<kspace Command Window
>> new
positive one
>>

Editor - C:\Users\irish\OneDrive\Документы\...
new.m
1 - n = 1;
2
3 - switch n
4 -     case -1
5 -         disp('negative one')
6 -     case 0
7 -         disp('zero')
8 -     case 1
9 -         disp('positive one')
10 -    otherwise
11 -        disp('other value')
12 -    end
```

Цикл while

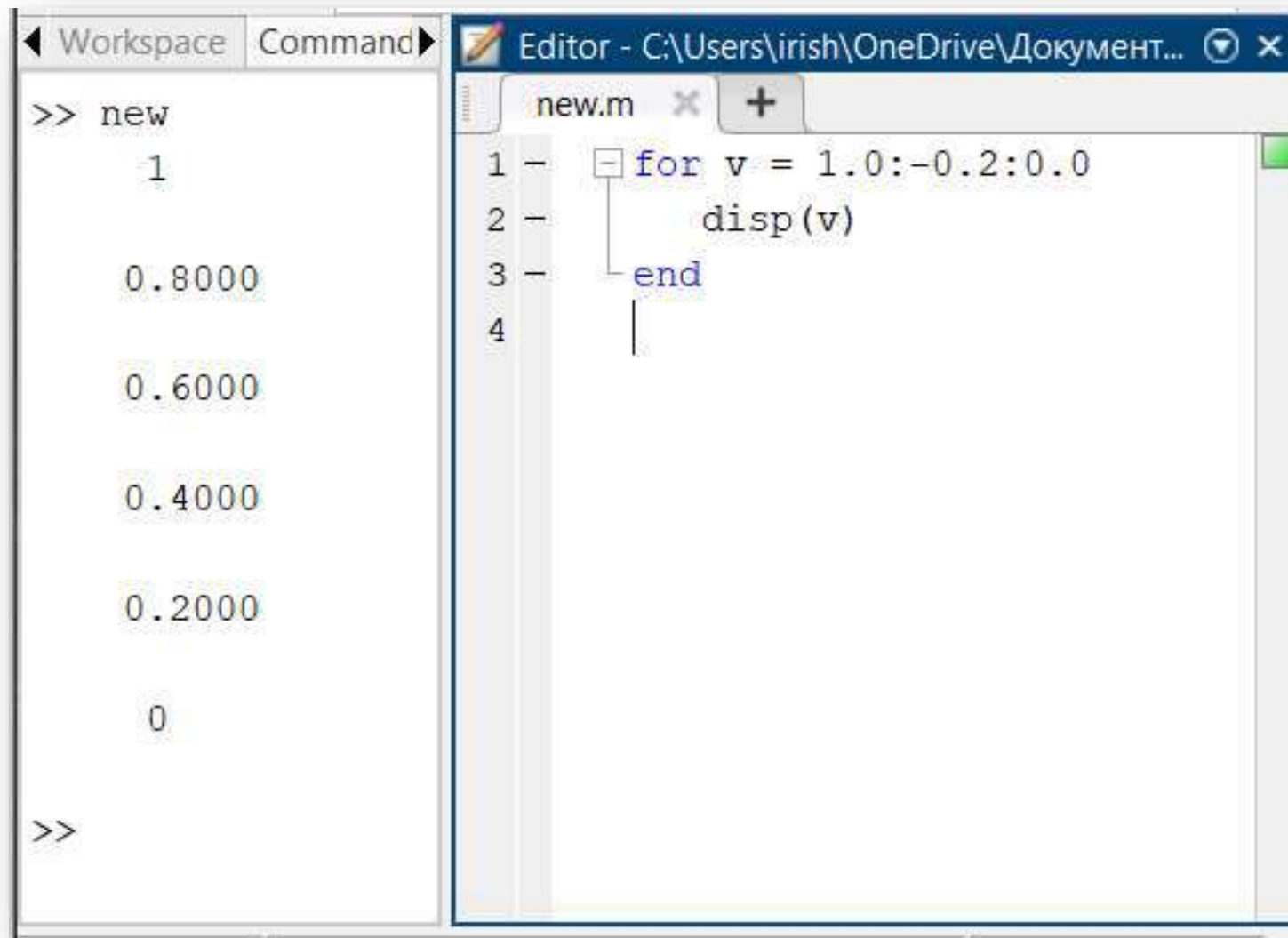


The image displays the MATLAB environment. On the left, the Command Window shows the execution of a script named 'new'. The prompt '>>' is followed by 'new', and the result 'n! = 3628800' is displayed. On the right, the Editor window shows the source code for 'new.m'. The code consists of seven lines: line 1 sets 'n = 10'; line 2 sets 'f = n'; line 3 starts a 'while' loop with the condition 'n > 1'; line 4 indents 'n = n-1;'; line 5 indents 'f = f*n;'; line 6 ends the loop with 'end'; and line 7 uses 'disp' to print the final value of 'n!'. The 'while' keyword on line 3 is highlighted with a blue box.

```
Workspace Command
>> new
n! = 3628800
>>

Editor - C:\Users\irish\OneDrive\Документ...
new.m
1 - n = 10;
2 - f = n;
3 - while n > 1
4 -     n = n-1;
5 -     f = f*n;
6 - end
7 - disp(['n! = ' num2str(f) ])
```

Цикл for



The image shows the MATLAB interface. On the left, the 'Command' window displays the output of a script named 'new'. The output consists of a vertical list of values: 1, 0.8000, 0.6000, 0.4000, 0.2000, and 0. On the right, the 'Editor' window shows the source code for 'new.m'. The code is a for loop that iterates from 1.0 to 0.0 in increments of -0.2, displaying each value.

```
<< Workspace Command >>
>> new
    1
    0.8000
    0.6000
    0.4000
    0.2000
    0
>>

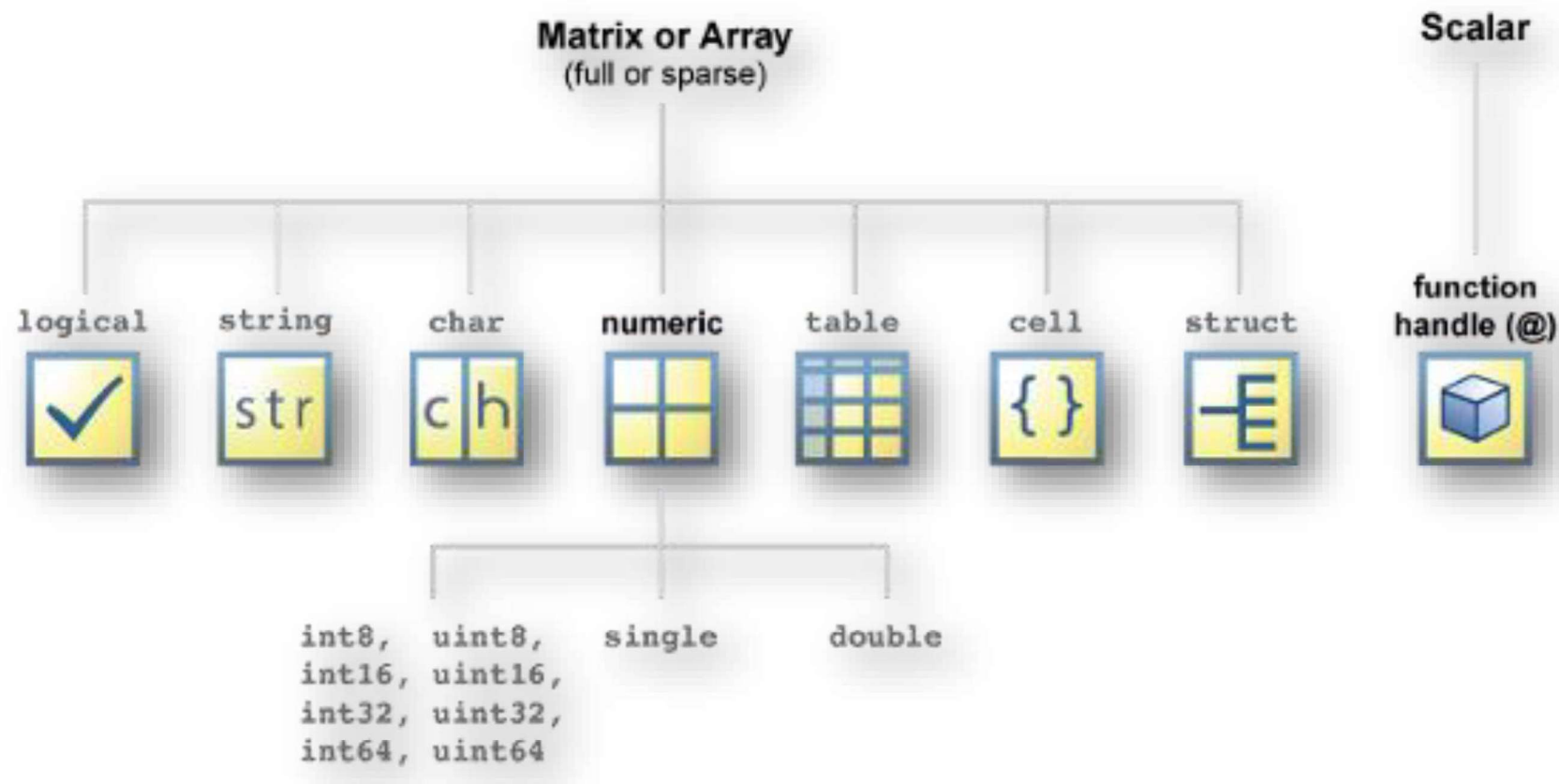
Editor - C:\Users\irish\OneDrive\Документ...
new.m
1 - for v = 1.0:-0.2:0.0
2 -     disp(v)
3 - end
4 -
```


Сохранение данных в файл .mat

```
A = rand(5);  
B = magic(10);  
save('myFile.mat', 'A', 'B', '-v7.3', '-  
nocompression')
```

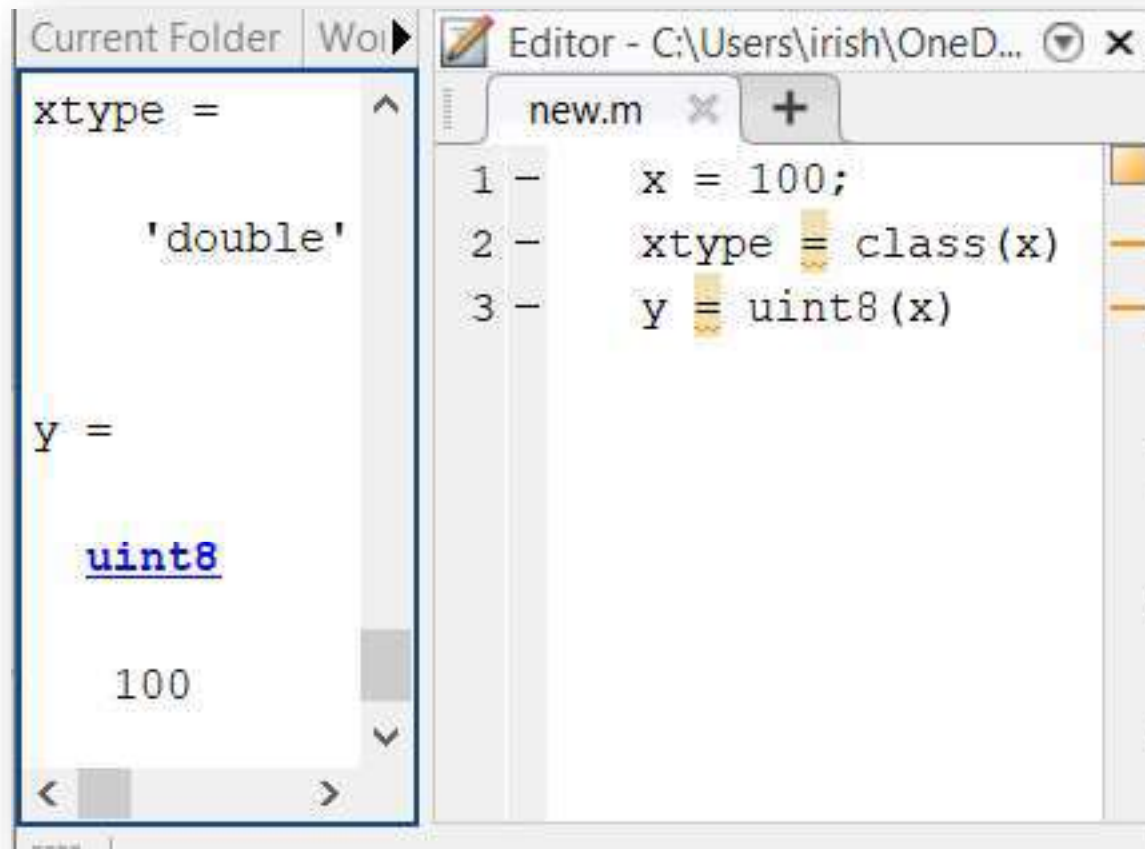
Типы данных

По умолчанию используется тип `double` (число с двойной точностью). Однако, если необходимо экономить память ЭВМ, то можно указывать самостоятельно желаемый тип.



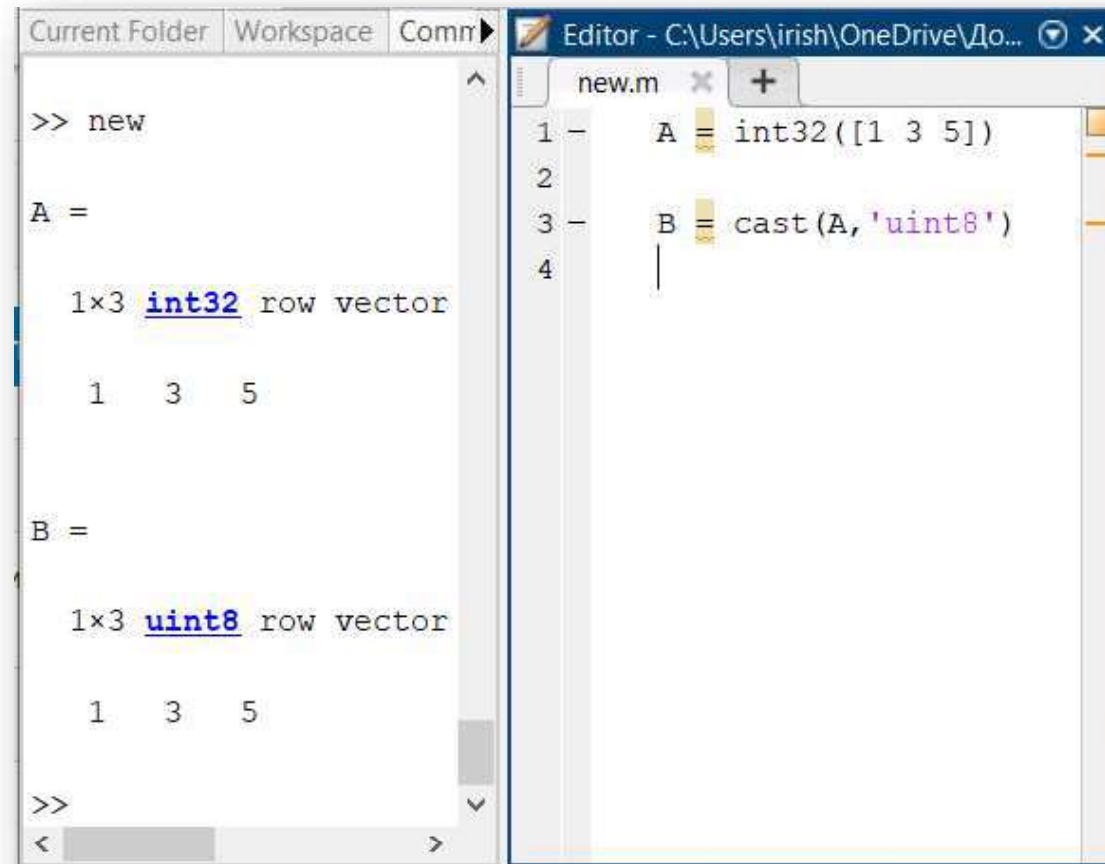
Приведения типов uint8()

Преобразование переменной с двойной точностью в 8-битное беззнаковое целое.



Функции преобразования типов `cast()`, `typecast()`

`cast()` - преобразование переменной в другой тип данных



The image shows a MATLAB interface with two windows. The 'Editor' window on the right contains the following code in a file named 'new.m':

```
1 - A = int32([1 3 5])
2
3 - B = cast(A, 'uint8')
4
```

The 'Command Window' on the left shows the execution results:

```
>> new

A =

    1x3 int32 row vector

     1     3     5

B =

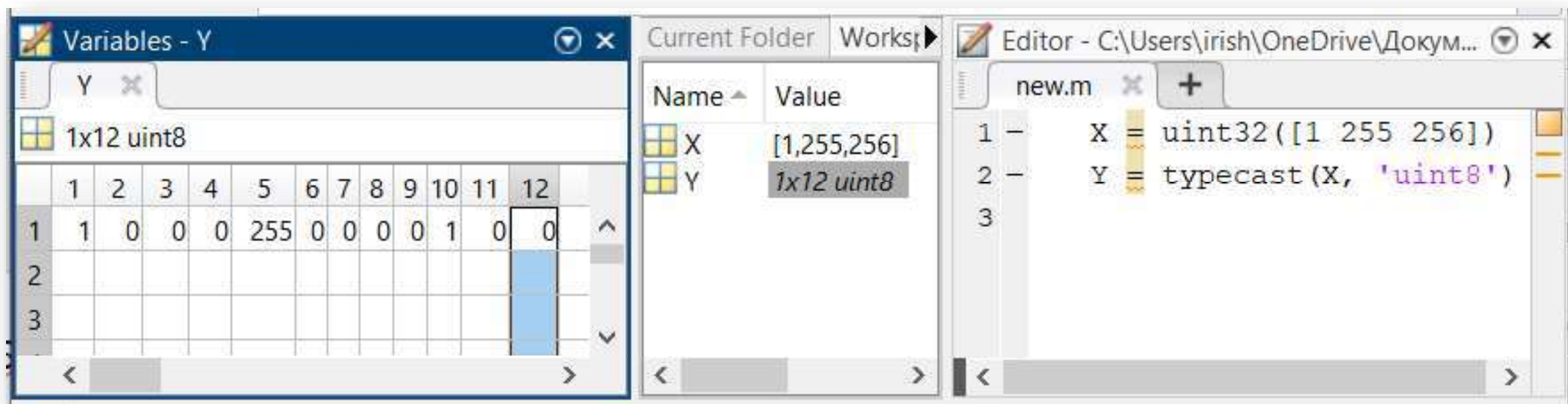
    1x3 uint8 row vector

     1     3     5

>>
```

Функции преобразования типов `cast()`, `typecast()`

`typecast()` - преобразование типов данных, не изменяя базовые данные



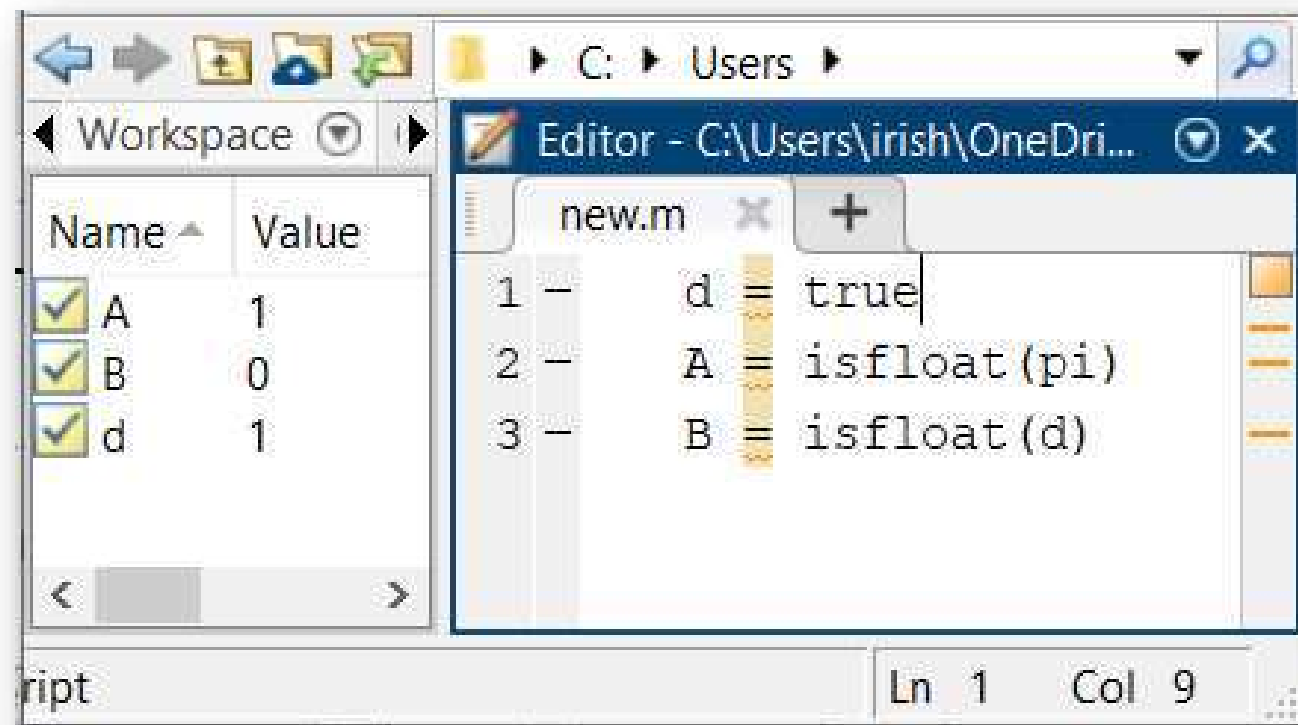
The image displays the MATLAB environment with three windows:

- Variables - Y:** Shows a 1x12 `uint8` array. The first row contains the values [1, 0, 0, 0, 255, 0, 0, 0, 0, 1, 0, 0].
- Workspace:** Lists two variables:
 - `X`: [1,255,256]
 - `Y`: 1x12 `uint8`
- Editor - C:\Users\irish\OneDrive\Докум...:** Contains the following MATLAB code in `new.m`:

```
1 X = uint32([1 255 256])
2 Y = typecast(X, 'uint8')
3
```

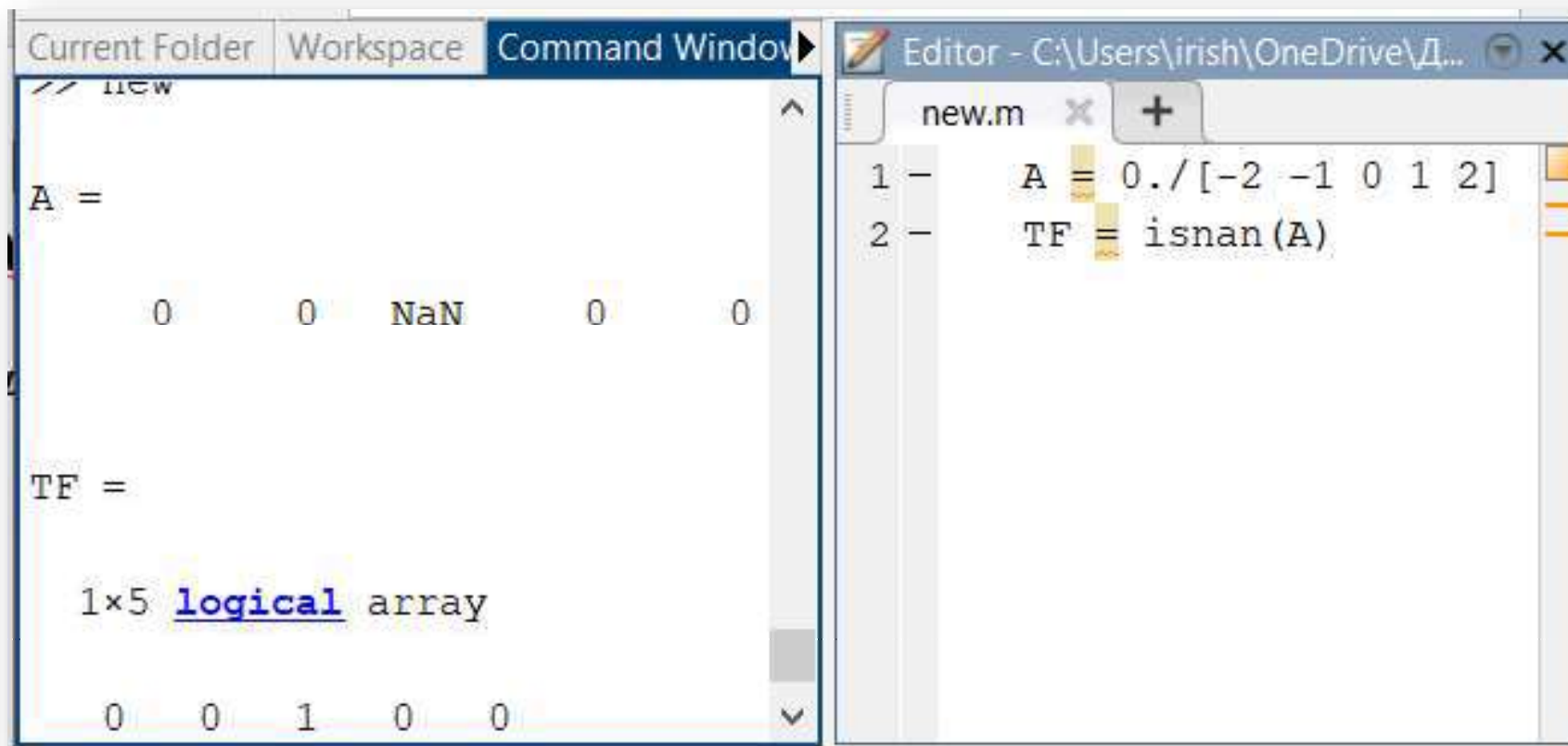
Функции определения типа переменной `isfloat()`, `isnan()`

`isfloat()` - определяет введено ли значение с плавающей точкой



Функции определения типа переменной `isfloat()`, `isnan()`

`isnan()` - определяет элементы массива, которые являются NaN



The image shows a MATLAB interface with two windows. The 'Command Window' on the left displays the results of a script: a 1x5 array 'A' with values 0, 0, NaN, 0, 0, and a logical array 'TF' with values 0, 0, 1, 0, 0. The 'Editor' window on the right shows the script 'new.m' with two lines of code: `A = 0./[-2 -1 0 1 2];` and `TF = isnan(A);`.

```
// new
A =
    0     0   NaN     0     0

TF =

1x5 logical array

    0     0     1     0     0
```

```
new.m
1 - A = 0./[-2 -1 0 1 2]
2 - TF = isnan(A)
```