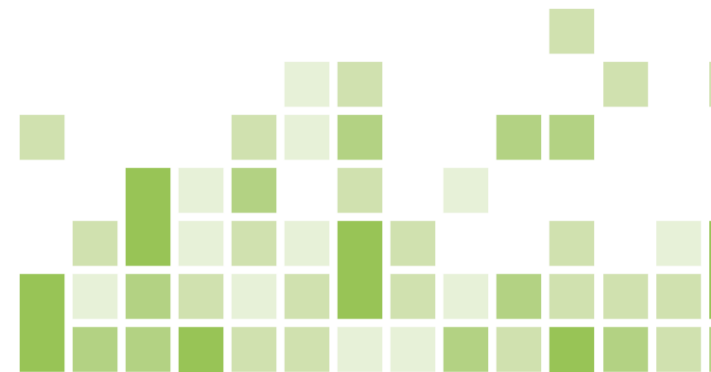




ТОМСКИЙ
ПОЛИТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ



МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ФИЗИЧЕСКИХ
ПРОЦЕССОВ
ЛЕКЦИЯ №1
«Введение в курс математического моделирования
физических процессов»

Отделение ядерно-топливного цикла

Лектор:
Зав. каф. - руководитель ОЯТЦ ИЯТШ
Горюнов А.Г.

2020

План лекции

1.1 Введение в курс математического моделирования физических процессов.

1.2 Введение в Matlab, интерфейс программы, редактор Matlab. Язык Matlab: константы и переменные;

1.3 Типы данных: скалярные, символьные, дата, время;

1.4 Операторы цикла и условного перехода;

1.5 Вектора, матрицы и массивы. Индексация в векторах. Создание векторов.

Литература по курсу:

<https://portal.tpu.ru/SHARED/a/ALEX1479/study/Matmod/Tab>

1.1 Введение в курс математического моделирования физических процессов

Целью данного курса является формирование у обучающихся знаний, умений и приобретение опыта в области математического моделирования процессов и объектов атомной отрасли с использованием стандартных методов и компьютерных кодов для проектирования и анализа.

Цель достигается за счет выполнения следующих **задач**:

- лекции (32 часа), получение информации о теоретическом разделе дисциплины, 16 баллов;
- практические занятия (32 часа), 2 контрольные работы 24 балла;
- лабораторные работы (24 часа), 6 лабораторных работ на 40 баллов;
- Экзамен, 20 баллов.

В дисциплине предусмотрен курсовой проект, диф. зачет, самостоятельная работа 64 часа.

1.1.1 Основные понятия

Моделирование – это научный прием, метод изучения, познания реального окружающего мира.

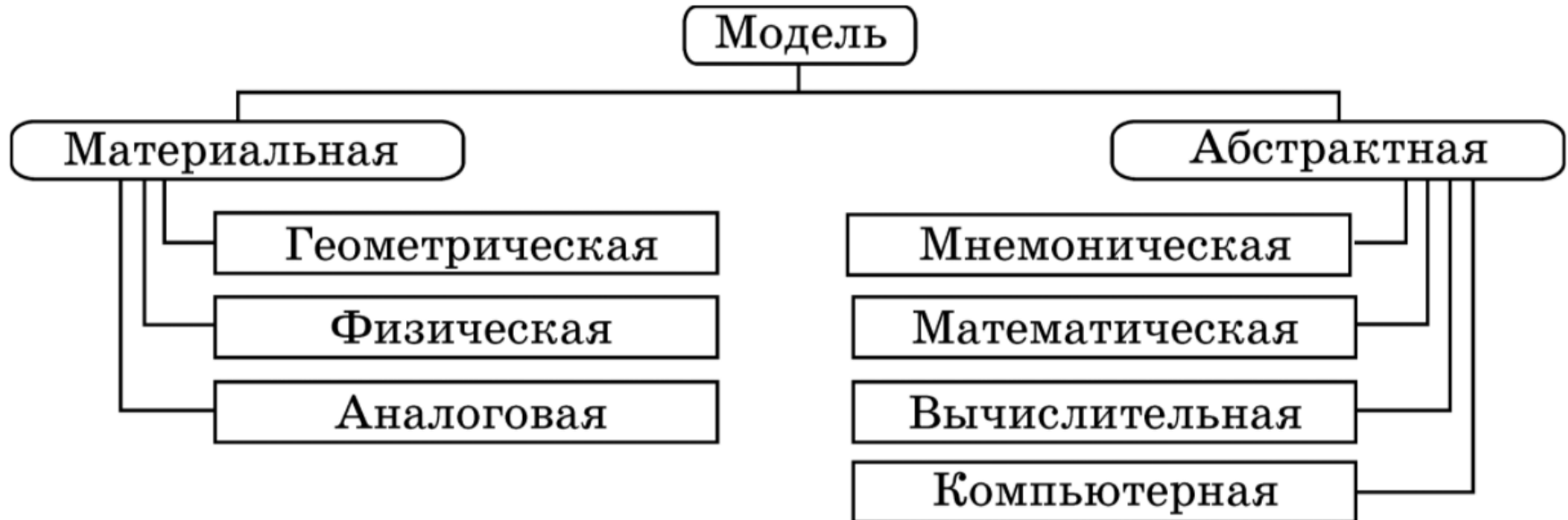
Процесс моделирования включает следующие задачи:

- ✓ реальный изучаемый объект (физическая система, процесс, явление), называемый **оригиналом**, замещается его **моделью** (физическим или абстрактным объектом);
- ✓ модель воспроизводит (имитирует) те свойства и характеристики оригинала, которые существенны для достижения поставленной цели моделирования (для решения конкретной задачи);
- ✓ над моделью проводятся эксперименты и исследования, на основе которых делаются выводы о свойствах объекта – оригинала.

Математическое моделирование предполагает описание исследуемых явлений, процессов, систем различной физической природы языком математических соотношений.

1.1.2 Классификация моделей

Классификация моделей по степени их абстрагирования от оригинала:



Голубева, Н. В. Математическое моделирование систем и процессов : учебное пособие / Н. В. Голубева. — 2-е изд., стер. — Санкт-Петербург : Лань, 2016. — 192 с.— Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/76825> (дата обращения: 03.03.2020). — Режим доступа: для авториз. Пользователей.

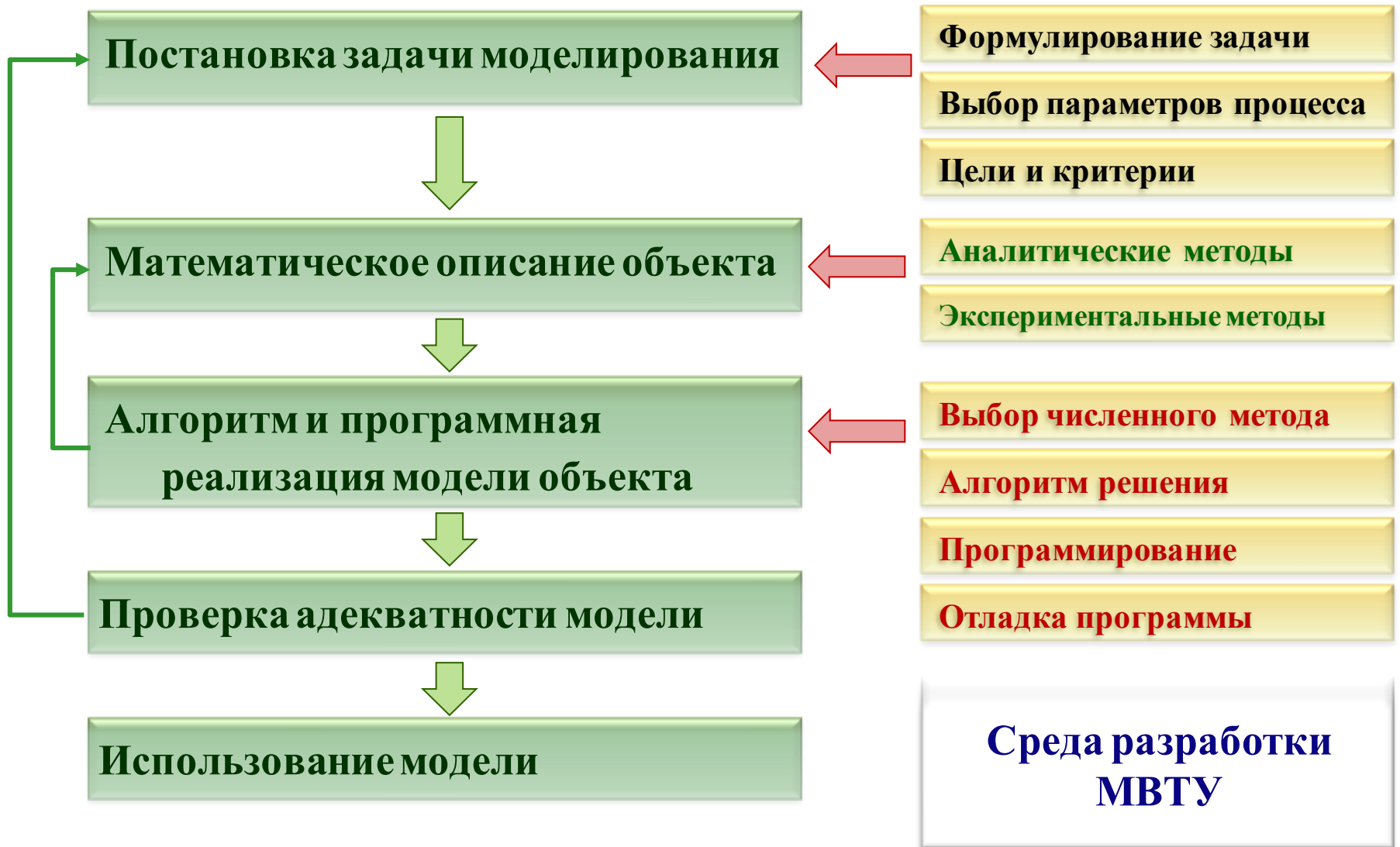
1.1.3 Классификация математических моделей

Признак классификации	Математические модели
Характер отображаемых свойств объекта	Структурные; функциональные
Принадлежность к иерархическому уровню	Микроуровня; макроуровня; метауровня
Степень детализации описания внутри одного уровня	Полные; макромодели
Способ представления свойств объекта	Аналитические, алгоритмические, имитационные
Способ получения модели	Теоретические, эмпирические

Аналитическая модель – математическая **модель**, представляющая собой совокупность **аналитических** выражений и зависимостей, позволяющих оценивать те или иные свойства моделируемого объекта.

Имитационная модель – универсальное средство исследования сложных систем, представляющее собой логико-алгоритмическое описание поведения отдельных элементов системы и правил их взаимодействия, отображающих последовательность событий, возникающих в моделируемой системе.

1.1.4 Подход к математическому моделированию процессов

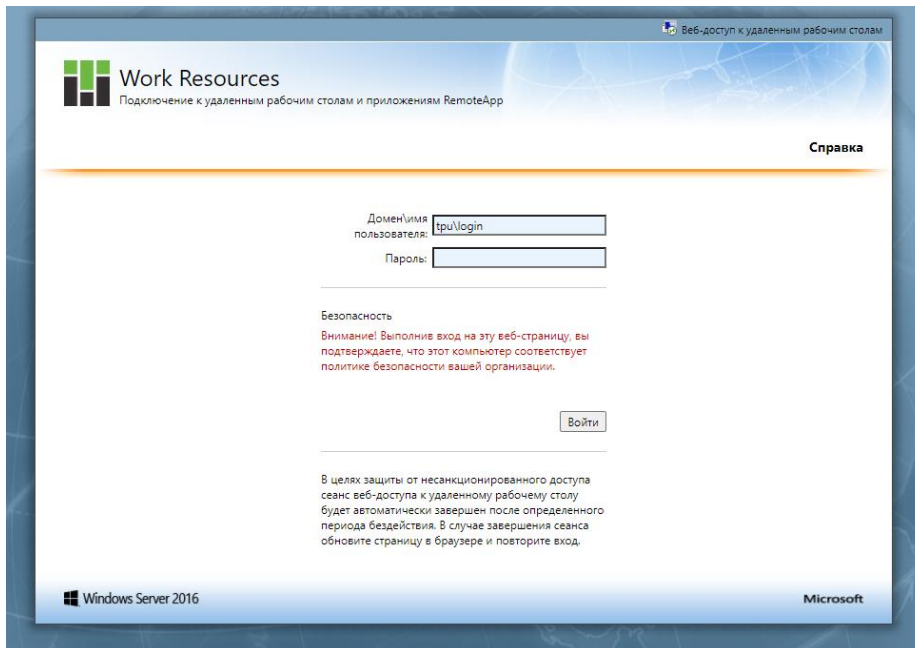


1.2 Введение в Matlab, интерфейс программы, редактор Matlab. Язык Matlab: константы и переменные

1.2.1 Работа с Matlab в ТПУ

Сотрудники ТПУ и студенты имеют возможность работать с Matlab на серверах ТПУ. Для этого необходим компьютер с доступом в интернет. Для запуска Matlab в любом браузере необходимо набрать адрес:

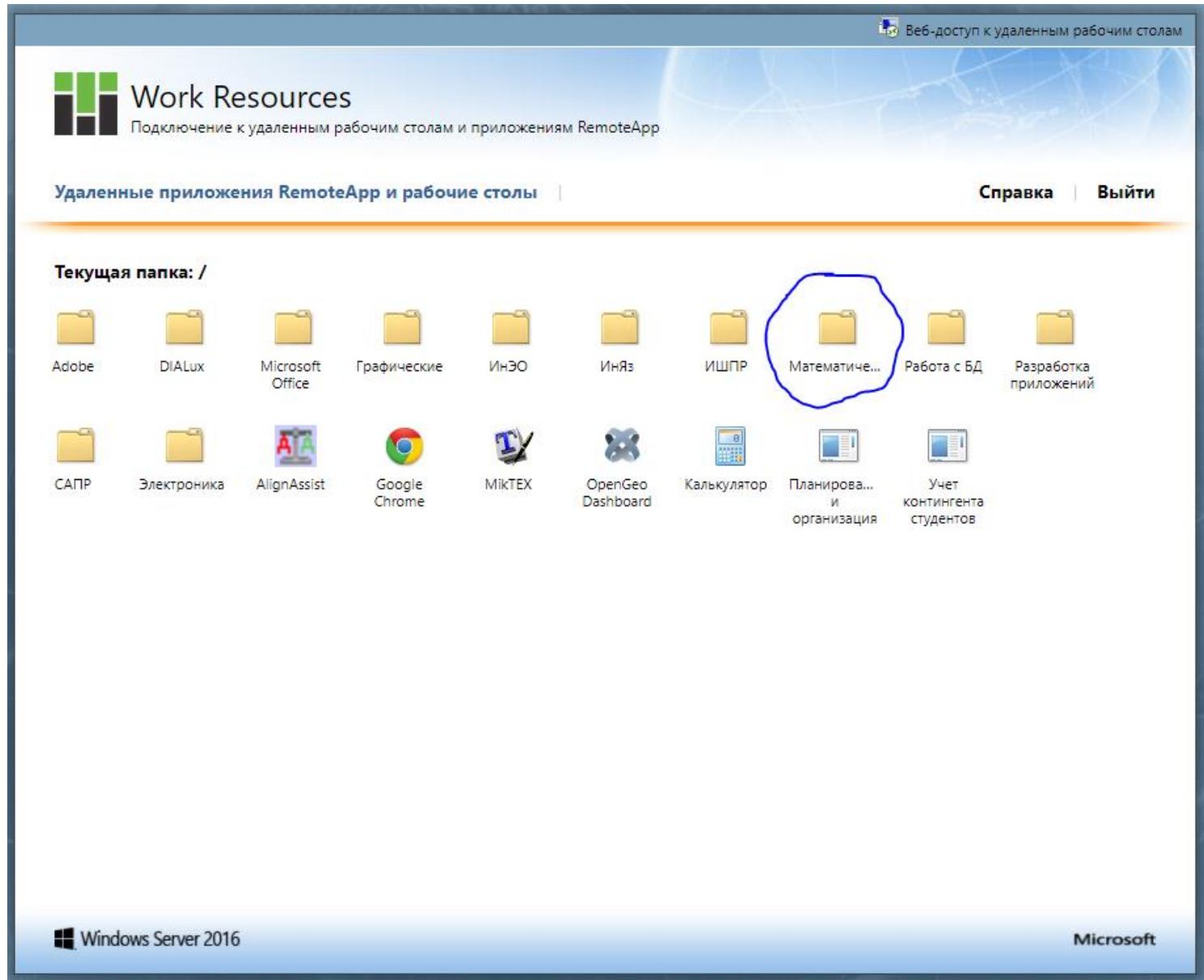
<http://www.vap.tpu.ru/>



В поле «имя пользователя» необходимо ввести **TPU\login**, где login – Ваш корпоративный логин (логин для входа в почту, личный кабинет и т.д.)

1.2.2 Запуск Matlab

Необходимо перейти в папку «Математические»



Веб-доступ к удаленным рабочим столам

Work Resources
Подключение к удаленным рабочим столам и приложениям RemoteApp

Удаленные приложения RemoteApp и рабочие столы | Справка | Выйти

Текущая папка: /

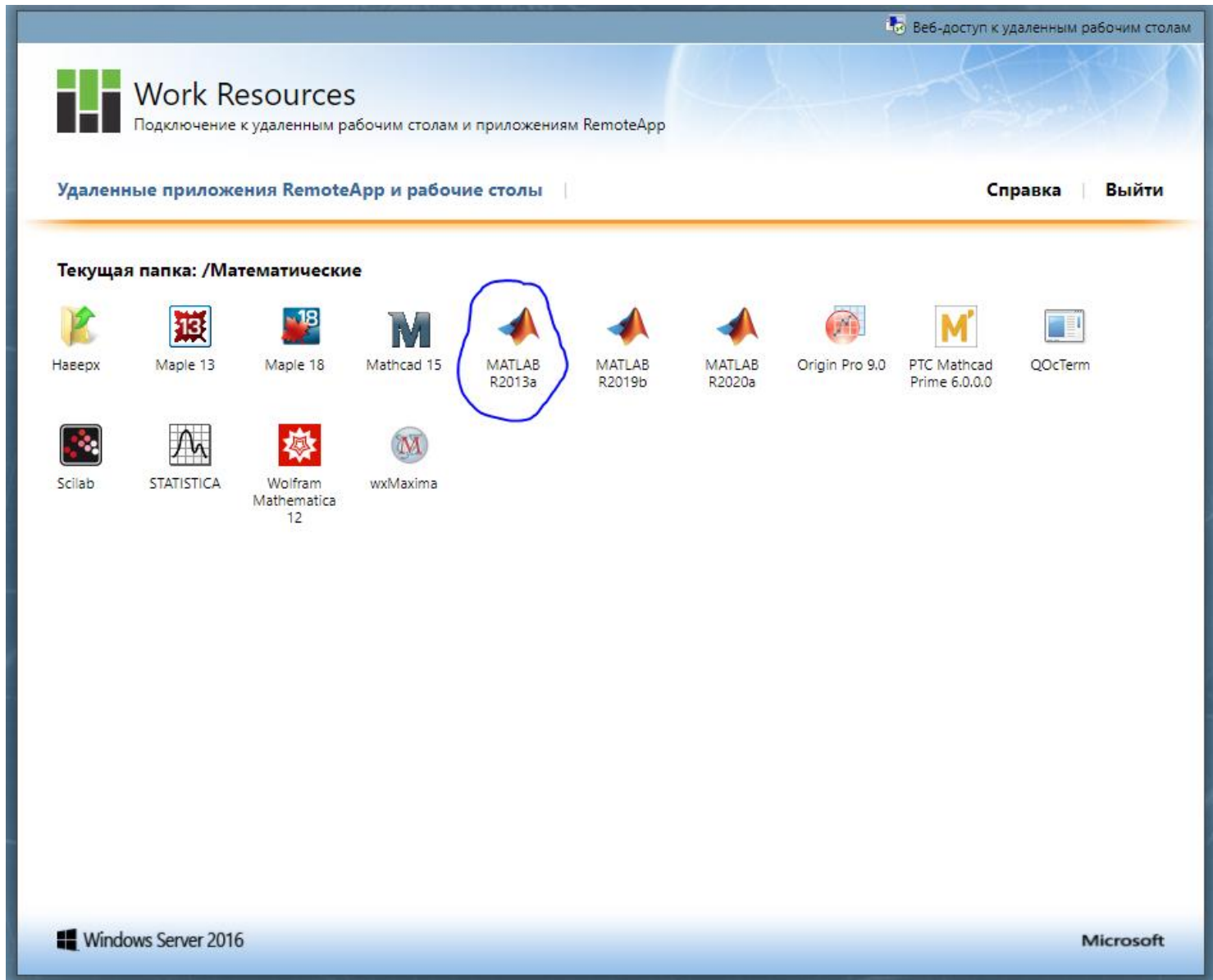
Adobe, DIALux, Microsoft Office, Графические, ИНЭО, ИнЯз, ИШПР, **Математиче...**, Работа с БД, Разработка приложений

САПР, Электроника, AlignAssist, Google Chrome, MikTEX, OpenGeo Dashboard, Калькулятор, Планирова... и организация, Учет контингента студентов

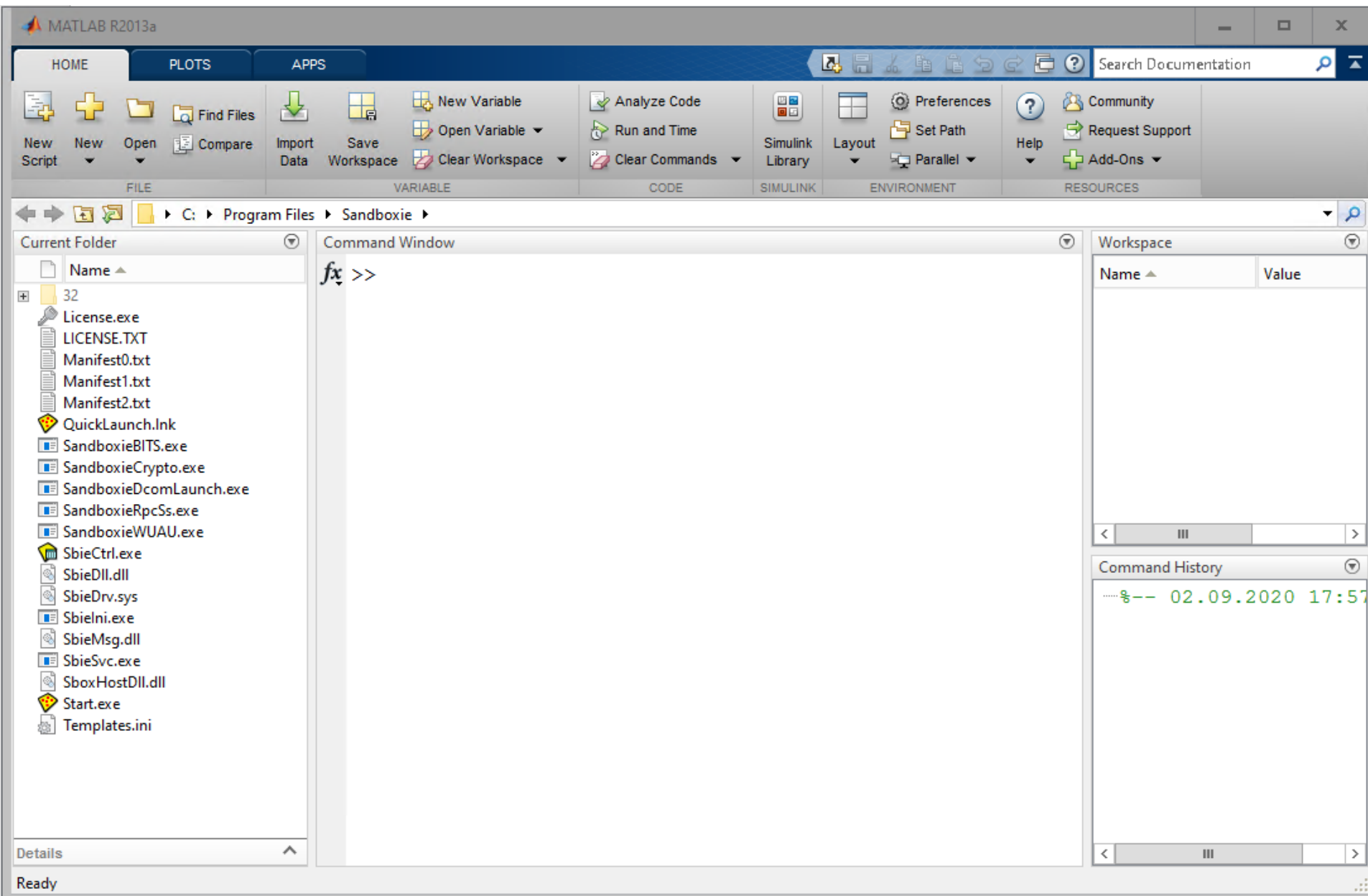
Windows Server 2016 Microsoft

1.2.2 Запуск Matlab

Запустить Matlab используя ярлык в пвпке.



1.2.3 Интерфейс Matlab



1.2.3 Интерфейс Matlab

Необходимо выбрать Ваш диск и директорию для сохранения файлов

The screenshot displays the MATLAB R2013a software interface. The top menu bar includes HOME, PLOTS, and APPS. Below it is a ribbon with various toolbars: FILE, VARIABLE, CODE, SIMULINK, ENVIRONMENT, and RESOURCES. The FILE toolbar is circled in blue, highlighting the 'U:' drive and 'MDL' folder in the breadcrumb navigation. The 'Current Folder' pane on the left shows a file tree with 'Matlab' and 'Multisym' folders. The 'Command Window' in the center shows the prompt 'fx >> |'. The 'Workspace' pane on the right is empty. The 'Command History' pane at the bottom right shows a timestamp '02.09.2020 18:04'. The status bar at the bottom left indicates 'Ready'.

MATLAB R2013a

HOME PLOTS APPS

New Script New Open Find Files Compare Import Data Save Workspace New Variable Open Variable Clear Workspace Analyze Code Run and Time Clear Commands Simulink Library Layout Set Path Parallel Preferences Help Community Request Support Add-Ons

U: MDL

Current Folder

Name

Matlab

Multisym

Command Window

fx >> |

Workspace

Name Value

Command History

02.09.2020 18:04

Ready

1.2.4 Константы в Matlab

Константа – это предварительно определенное числовое или символьное значение, представленное уникальным именем. Числа (например 1, -2 и 1.23) являются безымянными **числовыми константами**.

Другие виды констант в MATLAB принято назвать **системными переменными**, поскольку, с одной стороны, они задаются системой при ее загрузке, а с другой – могут переопределяться. Основные системные переменные, применяемые в системе MATLAB, указаны ниже:

i или **j** – мнимая единица (корень квадратный из -1);

pi – число π – 3.1415926...;

eps – погрешность операций над числами с плавающей точкой (2^{-52});

realmin – наименьшее число с плавающей точкой (2^{-1022});

realmax – наибольшее число с плавающей точкой (2^{1023});

inf – значение машинной бесконечности;

ans – переменная, хранящая результат последней операции и обычно вызывающая его отображение на экране дисплея;

NaN – указание на нечисловой характер данных (**Not-a-Number**).

1.2.4 Константы в Matlab

The screenshot displays the MATLAB R2013a environment. The Command Window shows the following commands and their outputs:

```
>> 2*pi  
ans =  
  
6.2832  
  
>> eps  
ans =  
  
2.2204e-16  
  
>> realmax  
ans =  
  
1.7977e+308  
  
>> 1/0  
ans =  
  
Inf  
  
>> Inf  
ans =  
  
Inf  
  
>> 0/0  
ans =  
  
NaN
```

The Workspace window shows the following variables and their values:

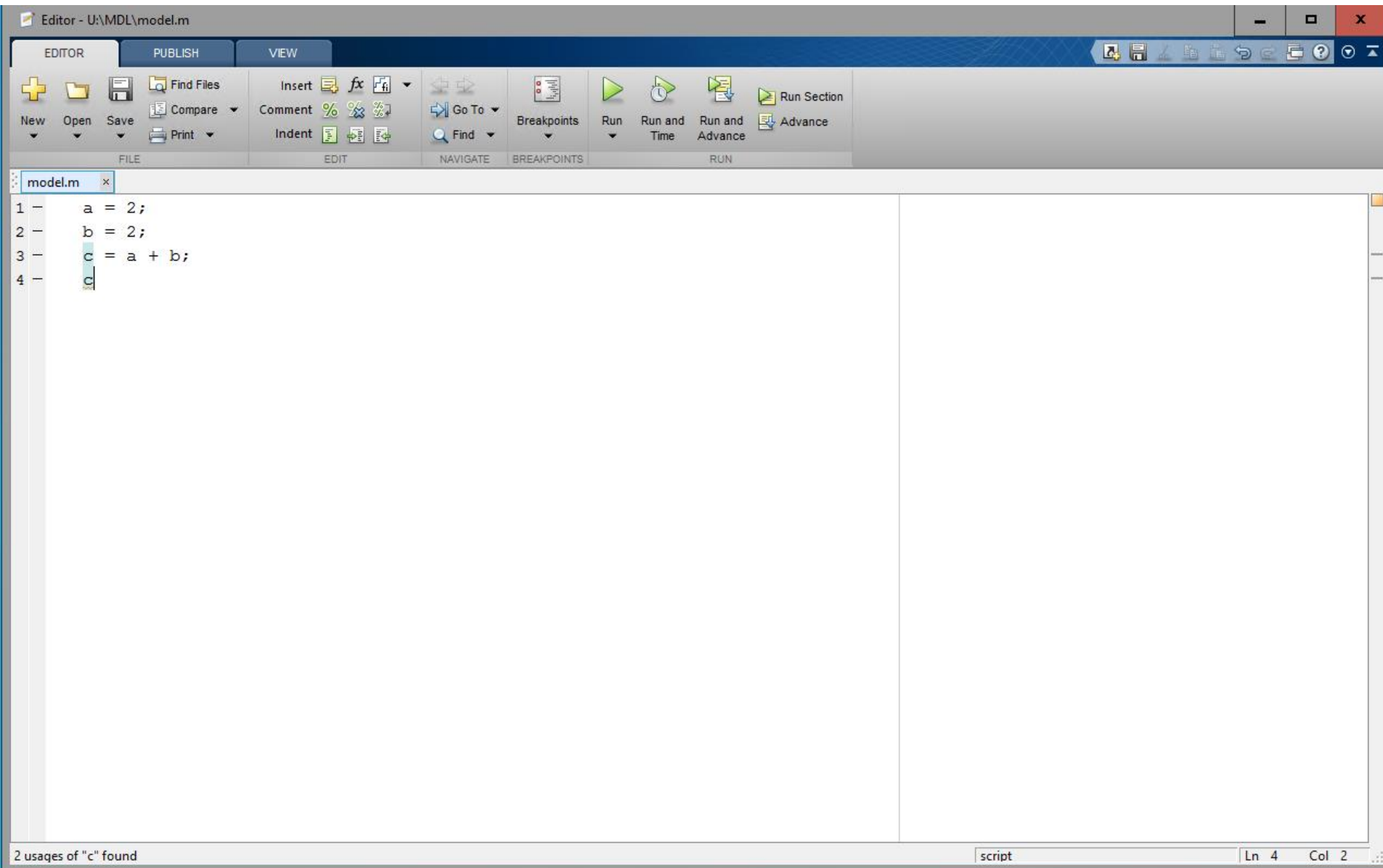
Name	Value
a	2
ans	NaN
b	2
c	4

The Command History window shows the following commands:

```
My fist model  
Untitled2  
model  
2*pi  
eps  
realmax  
1/0  
Inf  
0/0
```

1.2.5 Редактор языка Matlab.

Переменные в Matlab



1.3 Типы данных в Matlab

1.3.1 Основные типы данных

Основные типы данных

Основные типы данных: **числовые**, **logical** , **char** , **cell** , **struct** , **table** и **function_handle** .

```
Command Window
>> a = 1.14;
>> b = [1 2 3];
>> c = '123';
>> whos
  Name      Size      Bytes  Class  Attributes
  a         1x1         8  double
  ans       1x1         8  double
  b         1x3        24  double
  c         1x3         6   char
fx >>
```

```
Command Window
>> a = 1.1479; % По умолчанию двойная точность
b = single(a); % В переменной b получим одинарную точность
>> whos
  Name      Size      Bytes  Class  Attributes
  a         1x1         8  double
  b         1x1         4  single
```


1.3.1 Основные типы данных

```
Command Window
>> a = int32(10); % целая переменная 32 разряда
b = int8(127); % целая переменная 8 bit
>> whos
  Name      Size      Bytes  Class  Attributes

  a         1x1         4   int32
  b         1x1         1   int8

>> c = logical(1); % логическая переменная true / false (1 или 0)
>> whos
  Name      Size      Bytes  Class  Attributes

  a         1x1         4   int32
  b         1x1         1   int8
  c         1x1         1   logical
```

1.3.2 Символьные переменные

```

Command Window
>> x = sym('x'); % создаем символьную переменную
>> f1 = (x^100 - 2*x + 1)/(x^10 + 2*x + 1); % создали функцию
>> pretty(f1) % вывести функцию "красиво"

      100
x      - 2 x + 1
-----
      10
x      + 2 x + 1

```

1.3.3 Форматы представления даты

Dateform(номер)	Dateform (строка)	Пример
0	'dd-rmiM-yyyy HH:MM:SS'	11-Mar-1995 03:45
1	'dd-mmM-yyyy'	01-Mar-1995
2	'mm/dd/yy'	03/01/95
3	' mmm '	Mar
4	'm'	M
5	'mm'	3
6	'mm/dd'	03/01
7	'dd'	1
8	'ddd'	Wed
9	'd'	W
10	'yyyy'	1995
11	'yy'	95
12	' mmm yy '	Mar95
13	'HH:MM:SS'	15:45:17

1.3.3 Форматы представления даты

```
Command Window
>> n = datevec('09/03/20') % перевод из типа "дата" в числовой массив

n =

Columns 1 through 5

    2020         9         3         0         0
```

1.3.4 Форматы представления времени

```
Command Window
>> t = clock; % clock — возвращает вектор из 6 элементов, содержащий текущую дату и время
>> % в десятичной форме [год месяц день час минуты секунды].
>> % Первые пять элементов этого вектора — целые числа. Шестой элемент имеет
>> % несколько десятичных знаков после запятой.
>> t

t =

    1.0e+03 *
    2.0200    0.0090    0.0020    0.0190    0.0380    0.0287

>> time = fix(clock); % функция fix(clock) округляет число секунд до целого значения.
>> time

time =

    2020         9         2         19         40         41
```

1.4 Операторы цикла и условного перехода

1.4.1 Условный оператор if

```
if <выражение>  
<операторы1> % выполняются, если истинно  
условие  
else  
<операторы2> % выполняются, если условие  
ложно  
end
```

```
if <выражение1>  
<операторы1> % выполняются, если истинно  
выражение1  
elseif <выражение2>  
<операторы2> % выполняются, если истинно  
выражение2  
...  
elseif <выражениеN>  
<операторыN> % выполняются, если истинно  
выражениеN  
end
```

1.4.1 Условный оператор if

Command Window

```
>> x = 5;
if x > 0
disp(1); % выполняется, если x > 0
elseif x < 0
disp(-1); % выполняется, если x < 0
else
disp(0); % выполняется, если x = 0
end
    1
```

Command Window

```
>> x = -5;
if x > 0
disp(1); % выполняется, если x > 0
elseif x < 0
disp(-1); % выполняется, если x < 0
else
disp(0); % выполняется, если x = 0
end
    -1
```

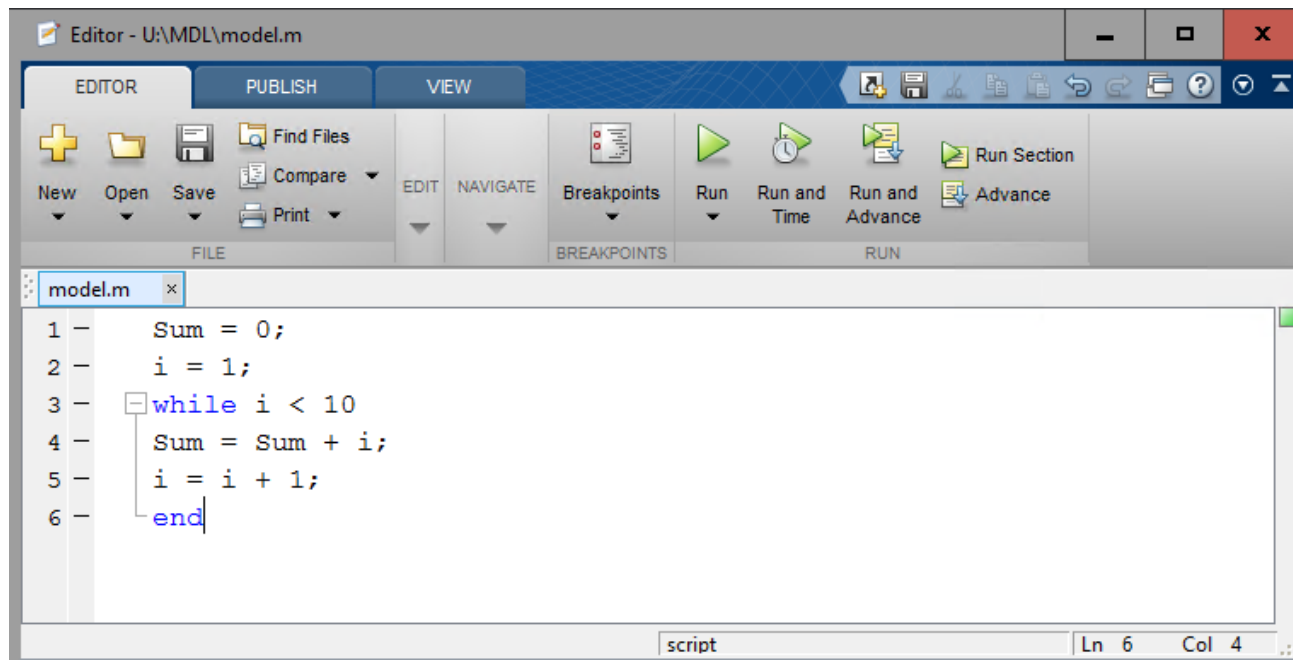
Command Window

```
>> x = 0;
if x > 0
disp(1); % выполняется, если x > 0
elseif x < 0
disp(-1); % выполняется, если x < 0
else
disp(0); % выполняется, если x = 0
end
    0
```

1.4.2 Оператор цикла while

```
while <условие>  
<операторы>  
end
```

```
Command Window  
  
>> Sum = 0;  
>> i = 1;  
>> while i < 10  
Sum = Sum + i;  
i = i + 1;  
end  
>> Sum  
  
Sum =  
  
45
```



1.5 Вектора, матрицы и массивы. Индексация в векторах. Создание векторов.

1.5.1 Вектора в Matlab

Command Window

```
>> a = [1 2 3 4];           % вектор-строка
>> disp( a(1) );          % отображение значения 1-го элемента вектора
disp( a(2) );             % отображение значения 2-го элемента вектора
disp( a(3) );             % отображение значения 3-го элемента вектора
disp( a(4) );             % отображение значения 4-го элемента вектора
1
2
3
4
```

Command Window

```
>> a = [10; 20; 30; 40];   % вектор-столбец
>> c = a(1) + a(4)

c =

    50
```

1.5.2 Матрицы в Matlab

Command Window

```
>> E = [1 0 0; 0 1 0; 0 0 1];           % единичная матрица 3x3
```

```
>> E
```

```
E =
```

```
    1    0    0
    0    1    0
    0    0    1
```

```
>> A = [1 0 0
        0 1 0
        0 0 1] % единичная матрица 3x3
```

```
A =
```

```
    1    0    0
    0    1    0
    0    0    1
```


1.5.3 Индексация в векторах и матрицах

Command Window

```
>> A = [1 2 3  
        4 5 6  
        7 8 9];
```

```
>> A(1,1)
```

```
ans =
```

```
1
```

```
>> c = A(1,1) + A(1,2) + A(3,3)
```

```
c =
```

```
12
```

```
>> A(1,2)
```

```
ans =
```

```
2
```

```
>> A(3,3)
```

```
ans =
```

```
9
```