Федеральное агентство связи

Ордена трудового Красного Знамени федеральное государственное бюджетное

образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

Кафедра Математической кибернетики и информационных технологий

Отчет по лабораторной работе № 1

по дисциплине «Введение в профессию»

Выполнил: студент группы БВТ 1905

Колышев Николай Игоревич

Проверил: Мосева Марина Сергеевна

Москва, 2019

Лабораторная работа № 1

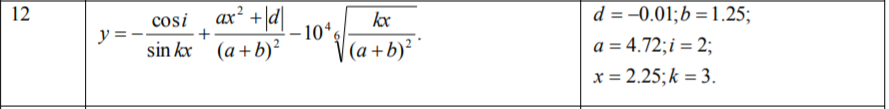
**Название:** Простейшие вычисления в MatLab

**Цель:** Изучение интерфейса пользователя системы MATLAB и основ работы с системой в режиме прямых вычислений.

**Индивидуальное задание**:

* **Часть 1**

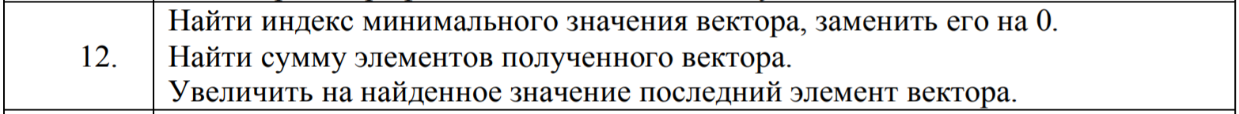
1. В командном окне задать значения переменных, согласно варианту задания, представленному в таблице.



1. Записать выражение на языке MATLAB.

* **Часть 2**

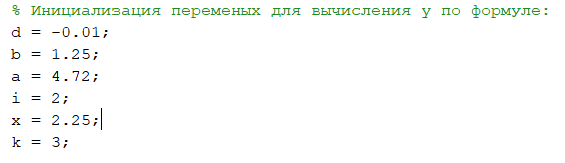
1. Сформировать массив из значений переменных и вычисленного значения выражения, используя свой вариант.
2. Произвести операции с полученным массивом согласно варианту задания.



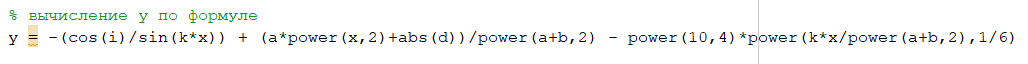
**Ход работы.**

***Часть 1.***

Итак, код начинается с инициализации переменных, необходимых для вычисления значения y по заданной формуле. Мы создаём переменные d, b, a, i, x, k, тут же присваивая им соответствующие значения, указанные в таблице.



Следующим этапом нашей работы будет вычисление значения y по формуле, позаимствованной из всё той же таблицы. Здесь мы используем такие операторы, как унарный знак минус (-), а также бинарное суммирование (+), вычитание (-), умножение (\*) и деление (/). Кроме того, в формуле использованы функции cos() и sin() для вычисление соответственно косинуса и синуса. Встречается и функция abs(), возвращающая модуль числа, и, наконец, применение функции power() позволяет возводить аргумент в произвольную степень.



***Часть 2.***

Переходя ко второй части работы, необходимо создать вектор Mas, элементами которого станут инициализированные до этого переменные d, b, a, i, x, k.

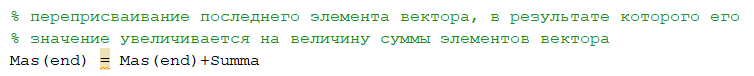


Теперь найдём индекс минимального элемента вектора, используя такие функции, как min(), которая возвращает значение минимального элемента в матрице и find(), которая находит элемент по заданному условию (в данном случае проверяется, является ли он минимальным) и возвращает индекс этого элемента.

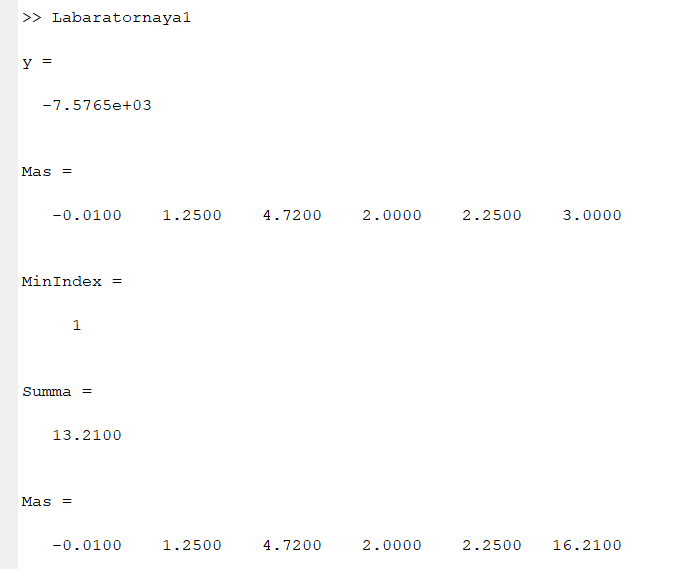
******

Нашим следующим шагом становится нахождение суммы элементов вектора. На помощь приходит функция sum(), она, собственно, и возвращает значение этой суммы, после чего мы присваиваем его переменной Summa.

******

Финальным этапом нашей работы будет присваивание последнему элементу вектора его текущего значения, увеличенного на сумму элементов вектора, найденную на предыдущем шаге.Из новшеств в этой части кода, пожалуй, стоит выделить обращение к элементу вектора. Мы добились этого, прописав имя вектора и указав в круглых скобках служебное слово end, позволяющее обратиться непосредственно к *последнему* элементу в векторе.****

Теперь взглянем на то, как работает наша программа, протестировав её код.

****

Сперва MatLab выводит в окно значение переменной y, что мы от него и ожидали. Затем пользователь видит инициализированный вектор из значений наших переменных. Следующей строкой выводится индекс *1* минимального элемента. Очевидно, что этот элемент равен -0.0100, а поскольку в MatLab`е нумерация элемента начинается с *1*, а не *0*, то и его индекс абсолютно соответствует действительности. После этого в окне появляется значение суммы всех элементов вектора. Если взять калькулятор и вручную просуммировать все наши переменные, то мы и правда получим 13.2100, что в очередной раз доказывает корректность кода. В последней строке снова обнаруживаем наш массив, но теперь его последний элемент увеличен на значение 13.2100, которое мы только что рассматривали. Именно этого мы и хотели от программы.

**Исходный код программы.**

***///начало кода///***

% Инициализация переменных для вычисления y по формуле:

d = -0.01;

b = 1.25;

a = 4.72;

i = 2;

x = 2.25;

k = 3;

% вычисление y по формуле

y = -(cos(i)/sin(k\*x)) + (a\*power(x,2)+abs(d))/power(a+b,2) - power(10,4)\*power(k\*x/power(a+b,2),1/6)

% Создание вектора из 6 элементов

Mas = [d b a i x k]

% поиск индекса минимального элемента в векторе

MinIndex = find(Mas==min(Mas))

% нахождение суммы элементов вектора

Summa = sum(Mas)

% переприсваивание последнего элемента вектора, в результате которого его

% значение увеличивается на величину суммы элементов вектора

Mas(end) = Mas(end)+Summa

***///конец кода (это не скриншот, всё, что расположено выше, полностью написано в программе MS Word)///***

**Заключение.**

В заключение нужно сказать, что мы научились применять MatLab в области достаточно сложных математических операций. Мы узнали о множестве математических функций: cos(), sin(), power(), abs(), а также об унарных и бинарных операторах MatLab. Кроме того, была произведена работа с векторами, в ходе которой были задействованы функции min() для нахождения минимального элемента и find() для поиска его индекса. Не обошлось и без функции sum(), пришедшей к нам на помощь, когда было необходимо просуммировать элементы массива. Теперь начало нашего пути в изучении языка программирования MatLab положено, и уже сейчас мы знаем достаточно, чтобы упростить работу пользователя с рутинными математическими вычислениями и преобразованиями.