# CSC 372 report1

Yifan Li (yifanli@Email.arizona.edu), Alex Melnychuck(amelnychuck@email.arizona.edu)

April 7, 2020

## 1 Abstract

## 2 Introduction

## 3 Why Matlab?

## 4 History

Matlab is a language invited by Cleve Moler, at first it was not a programming language, it was just a interactive matrix calculator. At 1970S, the author, Cleve Moler wants his student can use 'LINPACK' and 'EISPACK' to write program more conveniently, instead of using Fortran. Then he uses Fortran to write the first version of Matlab. And with the development of Matlab until today, it becomes a very important and popular language to scientist or scholars,

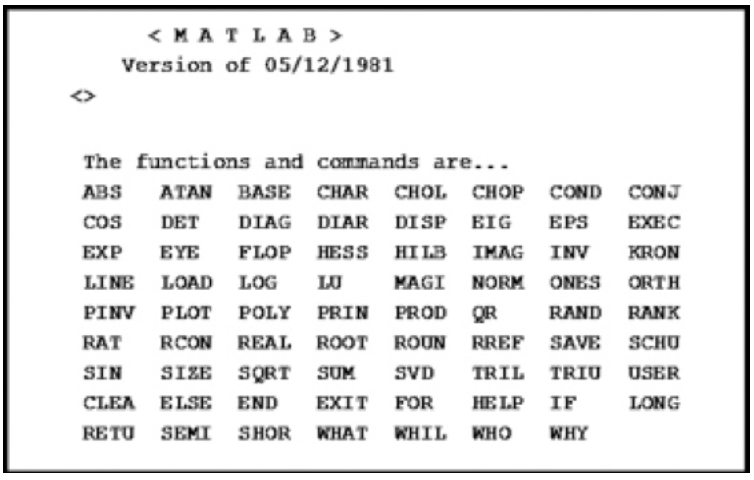


Figure 1: Early Version of Matlab

# 5 Control Structures

Matlab contains many typical control structures that are found in other programming languages such as C or Java. For a studied programmer first learning to program with Matlab, many control structures will be familiar in concept, even if differences in syntax can take getting used to. To someone who is new to programming, the control structures in Matlab may be viewed as simple yet powerful—allowing one to demonstrate a variety of functionality with a relatively small set of structures.

## 5.1 If, Else, and Elseif

A concept that is found in most programming languages, if, else, and elseif control statements carry over into Matlab much like they function in traditionally taught languages.

Observe the following code:

```
grade = 90;
if grade >= 90
    disp('You re a good student!');
else
    disp('Try harder!');
end
```

The if statement will execute the block of code when the condition specified after the if keyword is true. In this case, the message "You're a good student!" will be displayed in the console if the grade variable is set to 90 or greater.

An else statement will execute when the initial if condition is not met. In the example above, a grade that is less than 90 will result in the message "Try harder!" displaying in the console.

The elseif statement can be used to add additional cases to basic if else statements.

Using both elseif and else blocks are not required—that is to say that an if statement can be used on it's own. Conditions are checked from top to bottom, and the first condition that evaluates to true will execute it's corresponding statement.

## 5.2 Switch

For control structures that involve many different cases, or conditions as they were referred to above, a Switch structure may be more appropriate.

Observe the following code:

```
k = 520;
switch k
case 520
area = "Tucson";
case 480
```

```
area = "Mesa"
end
disp(area);
```

The switch keyword takes the value of the variable that follows it—in this case k—and runs a comparison check against each of the subsequent case conditions below. Like the if else block, the first case condition that the variable matches will be the case that executes it's following statement. In the example above, the variable's value is 520 and so the message "Tucson" will be displayed to the user after the switch structure has been run.

## 5.3 For Loops

For Loops are a repetition control structure that can be used to iterate over a collection or sequence in a fashion similar to other programming languages. Although the syntax may vary from C or Java, the basic concept is the same.

Observe the following code:

```
for i = 1:5:25
    disp("I Love Csc372!");
end
```

In this example, the message "I Love Csc372!" is printed to the console a total of 5 times. The first number is the starting point for i, and the third number is the stopping point for i. The middle value represents the value by which i should be incremented (or decremented in the case of negative values.)

## 5.4 While Loops

Similar in nature to the for loop, a while loop can be used to repeat a block of code until a particular condition is no longer met. While loops can be commonly used in recursive functions and in iterative use cases.

Observe the following code:

```
year = 2000;
while year < 2020
    disp("It's not 2020! It's" + year)
    year = year + 1;
end
disp("Wow it's 2020")
```

In the example above, the year variable is initially set to 2000. When the program execution reaches the while block, it checks that the value is less than 2020. Since it is in it's first pass through the block, it continues to display, "It's not 2020! It's 2000". It will continue to increment it's year, and repeat the check to see if the year is still less than 2020. Once the year is actually 2020, it will exit the while block and continue to display "Wow it's 2020.")

### 5.5 Try Catch

Intro paragraph/sentence

Observe the following code:

```
m = rand(3,4);
n = magic(5);
try
      a = m*n;
      disp(a)
catch
      disp(size(m))
      disp(size(n))
end
disp("Since they have different size so can't do matrix
    multiplication")
```

In the example code above, it defines two matrix in different sizes, and try to multiple them, we know that the matrix multiplication is only legal when the size of matrix fit each others, so since m and n's sizes are not fit with each others, the content in try will not run, produces an exception and then be caught. The content in the catch will run. which display the size of matrices.

## 6 Data Types

## 7 Subprograms

## 8 Summary

## 9 References

## 10 Fun Fact

These are example citations such as [1] and [2] and [3] and [4] so that they show up in the References section below.

## References


[1] "Matlab," Apr 2020. [Online]. Available: https://en.wikipedia.org/wiki/MATLAB

[2] "Matlab." [Online]. Available: https://www.mathworks.com/help/matlab/

[3] [Online]. Available: https://www.mathworks.com/help/releases/R2014b/pdf_doc/matlab/getstart.pdf

[4] “Newest ’matlab’ questions.” [Online]. Available: https://stackoverflow.com/questions/tagged/matlab