MATLAB/GNU OCTAVE

Dongzhe Chen
University of Arizona
dongzhechen@email.arizona.edu

Peilin Feng
University of Arizona
peilinfeng@email.arizona.edu

April 21, 2020

ABSTRACT

(WAIT FOR COMPLETE)

1 Introduction

(WAIT FOR COMPLETE)

2 History

2.1 why was the language designed

Octave helps in solving linear and nonlinear problems numerically, and for performing other numerical experiments using a language that is mostly compatible with MATLAB.

2.2 who designed it

Developer: John W. Eaton and many others

2.3 what is its current status

It is a good language for now and still being used by many programmers.

3 Control Structures

There are many control structures inside Matlab. Just like many other basic language Java, C, python. Matlab has same control structures like for loop, if and else, while loop and so on. There are some similarities and differences.

The similarities between them is the structure looks the same, for loop, if else, and while loop are all having the same structure like Java or C. The unique part of Matlab is there is an extra element 'end' after each control structure. GNU Octave Beginner's Guide [4] has showed an example of if and else structure.

3.1 if and else

if condition 1

do something (body)

elseif condition 2

do something else (body)

else

do something else if if condition not met (body)

endif

The example above shows exactly how an if and else control structure looks in Matlab. The only unique part is the endif, by using the 'endif' toward the end of the structure declares that the if and else statement ends here. This unique 'end' not only helps to tell the structure ends here, but also can be used in many different structure. There are also 'endwhile', 'endfor' and so on. It will help to tell programmer when does the specific structure ends. There are some more examples shows while loop and for loop.

3.2 for loop

for i = 1 : 20

fprintf(i);

endfor

3.3 while loop

i = 1;

while i < 11

fprintf(i);

i++;

endwhile

The above shows for loop and while loop control structure. The unique shows 'endfor' and 'endwhile' to delcare these two structures end.

4 Data Types

4.1 Single

Single is just like the integer type in java or c. Compare to double, it requires less storage space but has a smaller range. The range is between $3 * 10^{38}$.

An example code :

N = 1;

fprintf('Here shows single type, N = %d', N);

end

The output will be "Here shows single type, N = 1" Just like any other structure, there is and 'end' to show the function ends here.

4.2 Double

Double is just like the double type in java or c. Compare to single, it requires more storage space but has a larger range. The range is between $3 * 10^{300}$.

An example code:

D = $3 * 10^{300}$;

fprintf('Here shows double type, D = %f', D);

end

Just like single, the double is also to declare numeric numbers but with a much bigger size.

4.3 Logical

Just like any other computer science language, matlab's logical uses 1 and 0 to represent true and false. And example :

```
T = 5 > 3 ;
F = 5 < 3 ;
end
```

It will give an output of true. Sine 5 > 3 does right and 5 < 3 does wrong.

4.4 Char

'Hello' is an prefect example of Char in matlab. Just like string in other language.

4.5 Cell array

cell array can be represented as 2D array. An example could be :

```
M = [[zeros(2,5)] ,[ones(2,1)]; 1:5,1;0:2:8,1;[8,7,2,5,9,1]];
end;
```

output will be :

```
0 0 0 0 0 1
0 0 0 0 0 1
1 2 3 4 5 1
0 2 4 6 8 1
8 7 2 5 9 1
```

4.6 Structure

each structure having named fields capable of storing an array of a different dimension and data type. An example will be :

```
S(1).name = 'Mike';
S(1).class = 'CSC 372';
S(1).finalGrade = 'A';
S(2).name = 'Bob';
S(2).class = 'CSC 372';
S(2).finalGrade = 'B';
end
```

The above shows how a structure can contain different data data type.

5 Subprograms

5.1 Class

Just like other programing languages, matlab has class as well. The unique part of matlab is that it starts with classdef title in the beginning. For each class, it mainly has two parts. An example will be :

```
classdef someclass
properties
endproperties
methods
```

```
endmethods
endclassdef
```

properties declare the property of this class, and methods declare its methods. Inside methods, it can include different functions. It brings another unique element inside matlab. Octave provides each function with an automatic variable called nargin. Whenever a function is called, the nargin will tell the number of arguments that has been passed into the function. So nargin could be used as a checking condition to see whether if there is valid arugment passed into the function.

Once the class is created, can create a main file that declares the class. For example, if one class named BasicClass has been created, then in main, can declare x = BasicClass(a). Then it will follow the class that you create. The example will be :

```
classdef BasicClass
properties
Value
end
methods
function obj = BasicClass(val)
if nargin == 1
obj.Value = val;
end
end
function a = get.Value(obj)
a = obj.Value;
end
end
end
```

Inside main, it can declare x = BasicClass(1:5)

the output will be 1 2 3 4 5

6 Summary

(WAIT FOR COMPLETE)

References

[1] Gnu octave: Simple examples.

[2] Matlab - gnu octave tutorial - tutorialspoint.

[3] John Wesley Eaton, David Bateman, and Søren Hauberg. *Gnu octave*. Network thoery London, 1997.

[4] Jesper Schmidt Hansen. *GNU Octave: Beginner's Guide: Become a Proficient Octave User by Learning this High-level Scientific Numerical Tool from the Ground Up*. Packt Publishing Ltd, 2011.

[4] [3] [2] [1]