

Week 2 review

- Matrix manipulations.
- Scripts
- Started Flow control .

- Let's fire up Matlab.

TODAY

More flow control.

Logic.

Exercises.

IF exercise

Ask user for password. If password is less than 6 characters long, return "invalid password"

IF exercise

Algorithm steps:

1. ask for user input (password)
2. check size of password
3. if size is too small, reject password

IF exercise

```
> pwd = input('Please enter password: ');  
pwdsize = length(pwd); %check size of pwdsize  
if (pwdsize < 6)  
    'password invalid'  
else  
    'valid password'  
end
```

SWITCH: a more convenient if

If you know in advance SEVERAL OUTCOMES ARE POSSIBLE, use SWITCH.

For example: ask user to choose a number from 1 to 10, each number prints an associated joke.

SWITCH: syntax

Something like:

```
Choice = input('number from 1 to 10? ')
```

```
switch Choice
```

```
    case 1 ...
```

```
    case 2 ...
```

```
...
```

```
    otherwise ...
```

```
end.
```

Choice is the name of the variable that can take the values listed after the “case” word.

SWITCH: syntax

More generally, if testing n
switch n

case $n1$

...

case $n2$

...

otherwise

...

end

$\{n1, n2, \dots\}$ is the list of
possible values that n
can take}

SWITCH example :

edit MHarmony.m

```
myage='29';
status='single';
avail='immediate';
info=input('what information do you want to know?...
Enter 1 for age, 2 for status, 3 for availability or 4 to quit.')
switch info
    case 1
        txt=['The age is ' num2str(myage)];           %vectors within brackets are
                                                    % concatenated too.

    case 2
        txt=['The status is ' status];
    case 3
        txt=['The availability is ' avail];
otherwise
    txt=['You Quitter!'];
end
txt
```

Switch exercise

"Guess the mood of the user (Happy, Surprised, Sad, Angry, Depressed, Overworked)"

Ok, only first four options.

Switch exercise

Mood.m

```
%here are your comments for help
```

```
name = input('What's your name? ');
```

```
%Note double single quotes...
```

```
%if you don't specify 's', you can still input
```

```
%a string with the ' command
```

```
randmood = floor(rand * 4) + 1; %rand is between 0 and 1
```

```
switch randmood
```

```
    case 1
```

```
        message = [name ' is happy!']
```

```
    case 2
```

```
        message = [name ' is surprised!']
```

```
    case 3
```

```
        message = [name ' is sad.']
```

```
    case 4
```

```
        message = [name ' is angry!']
```

```
end;
```

WHILE

A loop for an a priori unknown number of iterations.

```
WHILE (condition is satisfied)  
    keep doing this.  
End
```

WHILE: syntax

`while (condition)` %if condition is false,
loop is not executed,
“jumps to end.”

...

`if (condition)`

`break`

%exits while

`end`

...

`end`

Exercise

Write a program that asks user for PIN number, until user gets it right.

Exercise

- `pin=0;`
- `While (pin~=1234)`
 `pin=input('What's your pin number? ');`
- `end;`

Exercise

- Modify script to end if user has three bad entries.
- `pin=0;`
- `counter=0;`
- `while (pin~=1234)`
`pin=input('What''s your pin number? ');`
`counter=counter+1;`
`if (counter==3)`
`break`
`end`
- `end;`

Homework

Write a script that asks the users for their name and returns it in scrambled fashion.

Use the commands 'input' and 'randperm'.

Write a script that asks the users for their name and returns it in scrambled fashion.

Use the commands 'input' and 'randperm'.

Algorithm steps

- obtain user name
- Question: do we know the number of steps or repetitions?
- Can we obtain it?
- scramble using for loop.

Scramble.m

```
% notes on program/mer
```

```
name = input('What's your name? ','s');
```

```
name_sz = length(name);
```

```
neworder = randperm(name_sz);
```

```
for i=1:name_sz
```

```
    newname(i) = name(neworder(i));
```

*%Matlab dynamically adds cells to your variable in this way,
%but only when it is being constructed for the first time!*

```
end;
```

```
char(newname)
```

Run Scramble with a LONG name

Run it again with a smaller name.

What happened?

How truly inconvenient!!!

Next week, "functions" will deal with this issue.

Write a script that asks the users for their name and returns it in scrambled fashion.

Use the commands 'input' and 'randperm'.

DO the same WITHOUT a flow control statement!!!

Scramble.m

```
% notes on program/mer
```

```
name = input('What"s your name? ','s');
```

```
name_sz = length(name);
```

```
neworder = randperm(name_sz);
```

```
newname=name(neworder);
```

the "find" command

A very useful command!

It can save you some for loops...

Allows you to efficiently look for a value inside a matrix.

For instance, do you want to find all the RTs in your data file that are below 150 ms?

find

FINDS a specific value in an array (or matrix) and returns the indeces in which it is located

SYNTAX:

indeces = find(expression);

returns indeces for which expression is true.

the "find" command

RT might be a 10000 rows variable with RT values.

Let's imagine:

```
RT = [ 50 100 250 350 85 450 550 650 79 120 200 4000];
```

then

```
badRTIndeces = find(RT <150)
```

and

```
badRTs=RT(find(RT < 150)); % or badRTs=RT(badRTIndeces)
```

How would you change this code to find RTs<150 and RTs>1000?

```
find( (RT < 150) | (RT > 1000))
```

the "find" command

How would you change this code to find RTs<150 and RTs>1000?

- 1 . find((RT < 150) | (RT > 1000))
2. find((RT < 150) & (RT > 1000))

% | is logical OR

% & is logical AND

the "isempty" command

When FIND does not find ANY indices, it returns an empty matrix.

TRY:

```
A=[1 2 3;1 4 8];
```

```
B=find(A==1)
```

```
C=find(A==1000)
```

ISEMPTY(X) returns TRUE if the matrix X is empty. TRY:

```
if isempty(C)
```

```
    'it worked'
```

```
else
```

```
    'im screwed'
```

```
end
```

Exercise/Homework

Write a program that asks a user for a new password which must abide by the following rules:

- 1.Length must be at least 6 characters.
- 2.Must have at least one digit.
- 3.Must have at least one Upper case and one Lower case letters.

If user enters valid password, say "Ok, password valid"

If user enters a password without a digit, say "you forgot to include at least one digit"

If passwords does not have at least one Upper and one lower case letter, say "you forgot to include at least one upper case and one lower case letter"

Exercise/Homework

What flow control statements should we use and for which instructions?

- Since we don't know how many times it will take for the user to come up with a good pin number:

"while"

- To check on the appropriateness of a possible pwd:

"if"

- Since there are three types of feedback:

"switch"

Exercise/Homework

UTpwd.m

```
%This program asks for a new password from the user  
%the password must have at least one digit, one lower caps  
%and one upper caps letter and be at least 6 characters long.  
%Known bugs: program crashes if input is not  
%Programmed by Alejandro Lleras  
%Last updated: September 7, 2006
```

```
invalid=true;  
while (invalid)
```

```
end;  
'Success. Thanks!'
```

Exercise/Homework

UTpwd.m

```
%This program asks for a new password from the user
%the password must have at least one digit, one lower caps
%and one upper caps letter and be at least 6 characters long.
%Known bugs: program crashes if input is not
%Programmed by Alejandro Lleras
%Last updated: September 7, 2006

invalid=true;
while (invalid)
    pwd = input('Please enter an uptight password: ');
    pwd_sz = length(pwd);
    if (pwd_sz > 6) %if it passes the size test then we check for...

        end;

end;

'Success. Thanks!'
```

Exercise/Homework

```
invalid=true;
while (invalid)
    pwd = input('Please enter an uptight password: ');
    pwd_sz = length(pwd);
    if (pwd_sz > 6) %if it passes the size test then we check for...
        if ("pwd has a number") %if it also passes the number test then...

            end;
        end;
    end;

end;
'Success. Thanks!'
```

Exercise/Homework

```
invalid=true;
while (invalid)
    pwd = input('Please enter an uptight password: ');
    pwd_sz = length(pwd);
    if (pwd_sz > 6) %if it passes the size test then we check for...
        if ("pwd has a number") %if it also passes the number test then...
            if ("pwd has a lowercase")
                if ("pwd has an uppercase ")
                    invalid=false; %break would work too
                end;
            end;
        end;
    end;
end;

end;
'Success. Thanks!'
```

For next week...

1. Finish password code.
2. Write MessWord.m

This program takes a word as input and shuffles only the inner letters of the word, leaving the first and last letter undisturbed.

Alejandro --> Arjelandoo

House --> Husoee

MessWord

What are the steps to undertake?

1. Figure out if the word is valid
(no characters or numbers)
2. Figure out if the length is valid
(if length < 3, what should happen?)
3. Use your old homework to do the rest.