Day 3

- (Concept Question)
- Comments
- Suppressing "ans= "
- Fancy fprintf
- (Exercises)

Comments

Comments make the script more readable

- Any line that starts with a % will be ignored
- Anything *in* a line *after* the % is ignored

Comments

Good commenting – use from now on!

- Identifier block at beginning
- Break your code into sections when appropriate
- Include units in comments

Examples on next slides

Comments

Identifier block at beginning

```
%*********************************************************
% Day3_Exercise1.m
%
%    PROGRAM DESCRIPTION
%    This program finds the instantaneous x and  y position
%    of a rocket launched at a particular angle and speed.
%
%    Input :   All input is in the Data section
%    Output :  Prints results to a file named Day3_Ex1.txt
%
%    Written by Lorraine Olson
%               11/25/2013
%*********************************************************
```

Comments

If you have many input variables to define,
create a Data Section to group them together

```
%
%    Data
%
vlaunch=80           % launch velocity in m/s
g=9.81               % acceleration due to gravity in m/s^2
```

Units!!!!

Comments

Group calculations together and add
comments to clarify what is being computed

```
%
%    Calculations
%
u0=vlaunch*cosd(theta)       % initial x-velocity
v0=vlaunch*sind(theta)       % initial y-velocity
```

Try to group outputs together

```
%
%    Output
%
file_number=fopen('Day2_Ex4.txt','w')
fprintf(file_number,'With a launch veloci ...
fclose(file_number)
```

Suppressing "ans= "

Putting a semicolon ; at the end of a line suppresses the "ans= " output to the command window

- Works in command window or script
- Cleans up the command window
- Still allows fprintf statements to do their job:

```
>> a=3;
>> fprintf('The answer is %f \n',a);
The answer is 3.000000
```

semicolon

Fancy fprintf

fprintf allows us to control the number of digits used in printing variables

```
>> fprintf('The answer is %5.2f \n',a);
```

one space is blank
because you typed a blank

2 digits after the
decimal point

```
The answer is  3.00
```

5 total
places used,

Fancy fprintf

If you give an invalid format, Matlab invents a way to print it.

```
>> fprintf('The answer is %0.2f \n', a);
The answer is 3.00
```

(This one is invalid because you can't use 0 total places to print the number.)

Fancy fprintf

The 'f' in '%f' stands for 'fixed point'. We will use this format frequently.

%e for exponential is also quite useful

```
>> fprintf('The answer is %8.2e \n', a);
The answer is 3.00e+00
```

Others are available. Try 'help fprintf' or 'doc fprintf' for more details.

Fancy fprintf

fprintf is very picky. Common mistakes:

- forgetting the f or the %

```
>> fprintf('a is %4.2f and b is %4.2 \n',a,b);
a is 3.00 and b is >> |

>> fprintf('a is 4.2f and b is %4.2f \n',a,b);
a is 4.2f and b is 3.00
a is 4.2f and b is 2.00
```

- forgetting the \n which makes everything run together on one line

```
>> fprintf('a is %4.2f and b is %4.2f',a,b);
a is 3.00 and b is 2.00>>
```