B2 • Discussion #5 • 2021-02-26

Remind us ~10:38 am EST to stop Discussion for the quiz! (15 min + 5 upload)

Announcements

- Homework #2: Due this Monday, March 1st at 10 am (EST)
  - → done individually
  - → don't press the "Run" button... Read the directions carefully to understand what a live script is and the tasks!
- Monday, March 1st is the last day to drop without a "W"
- Class attendance: 0 will be given for students who are not participating in the GPP (muted, no video)

↳ REMINDER: VIDEO ON FOR DISCUSSION FOR FULL ATTENDANCE GRADE CREDIT

Review of Lecture Material

- User-defined functions that return more than one value
- User-defined functions that don't return anything but accomplish a task such as printing or plotting
- Program organization
- Persistent variables
- Function stubs
  - → consists of header, end, and not much in between
  - → just something that mimics what the function will do eventually
  - → tests something to make sure that input and output arguments are the correct type

FROM CLASS, EXAMPLE

Write a persistent function that calculates and returns a running sum of values that are passed to it.

```
1.  function outsum = persistsum(x)
2.  persistent runsum
3.  if isempty(runsum)
4.      runsum = 0;
5.  end
6.  runsum = runsum + x;
7.  outsum = runsum;
8.  end
```

x = 1:3 → 6 ; y = 2:4 → 9

runsum = runsum + sum(x)
0 + 6 => 6
runsum = runsum + sum(y)
6 + 9 => 15

note: the output argument of the function and the persistent local variable must have different names.

Question: What if we wanted to pass a vector? A Matrix?

6. runsum = runsum + sum(x)

6. sum(sum(x))

Given the following func header

```
function doit(a,b)
```

Which calls are valid?

```
>> fprintf('The result is %.1f\n', doit(4,11))
```

INVALID — no output arg, nothing returned so nothing to print

```
>> doit(5, 2, 11.11)
```

INVALID — too many input arg

```
>> x = 11;
>> y = 3.3;
>> doit(x,y)
```

VALID — everything checks out!
- 2 input arg
- func name ok

Function Stubs

What the hell is it?

```
function out = funcname(a)
% comment block
out = 12;
end
```

What's the point?

```
program.m
% comment block

func1

func2

  :

funcn
```

Function that only plots

Write a function that receives an x vector, a minimum value, and a maximum value, and plots sin(x) from the specified minimum to the specified maximum.

plotfunc.m

```
function plotfunc(x, xmin, xmax)
% plots sin(x) from specified min to max
% does not return anything
x = linspace(xmin, xmax);
plot(x, sin(x))
xlabel('x')
ylabel('sin(x)')
end
```