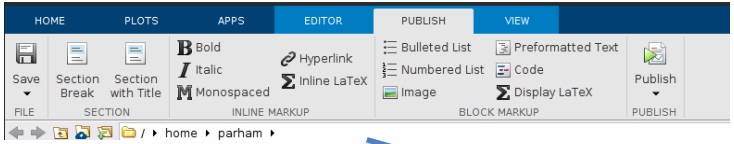
Lab Session 02
09/06/2024

ENME 303 Computational Methods for Engineers

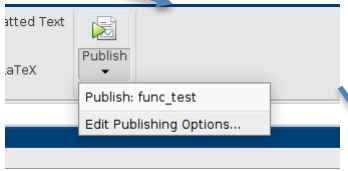
Slides adapted from Parham Oveissi (2023)

Publishing MATLAB Codes

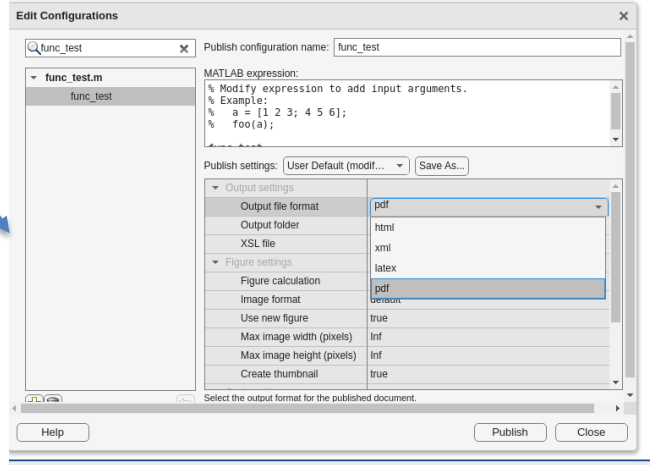


Go to the PUBLISH tab

Click in Edit Publishing Options



Change the output format to pdf and then click on publish

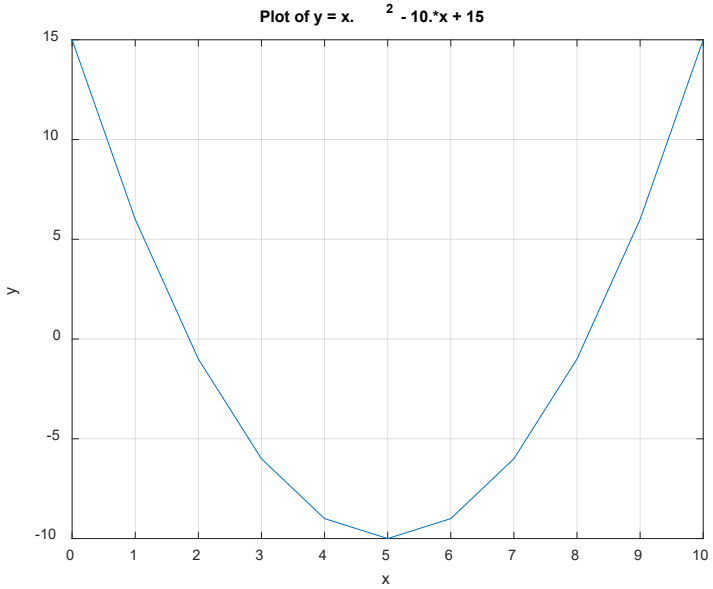


Plot

- plot (x, y, 'Color Linestyle Marker', 'linewidth' ,3)
  - plots the vector y with respect to x.

Color		Marker Style		Line Style	
y	yellow	.	point	-	solid
m	magenta	o	circle	:	dotted
c	cyan	x	x-mark	-.	dash-dot
r	red	+	plus	–	dashed
g	green	*	star	<none>	no lines
b	blue	s	square		
w	white	d	diamond		
k	black	v	triangle (down)		
		^	triangle (up)		
		<	triangle (left)		
		>	triangle (right)		
		p	pentagram		
		h	hexagram		
		<none>	no marker		

title, xlabel, ylabel, grid

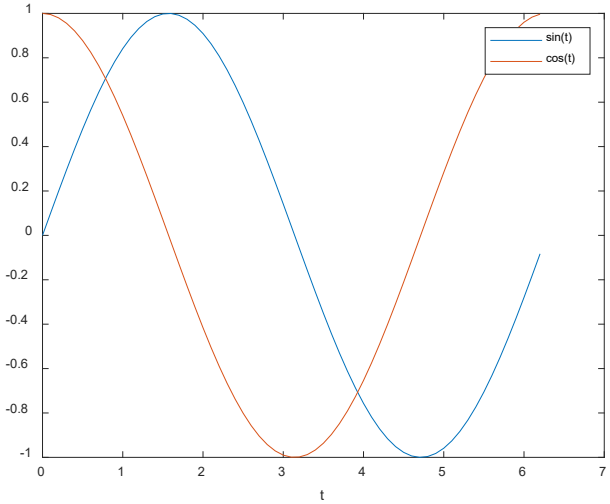



```
x = 0:1:10;
y = x.^2 - 10.*x + 15;
plot(x,y);
title('Plot of y = x.^2 - 10.*x + 15');
xlabel('x');
ylabel('y');
grid on;
```

Multiple Plots

- Making a new figure:

```
>> figure;
>> plot(x1,y1)
>> hold on %Allows plotting multiple traces on top of each other
>> plot(x2,y2)
>> legend ('sin(t)', 'cos(t)')
```



- subplot?!
- semilogx?!
- semilogy?!
- plotyy?!

Use doc <function name> for more information

Loops (for loop)

- The for loop is a loop that executes a block of statements a specified number of times.

```
for   k = i : T : n
      commands f (k)
end
```

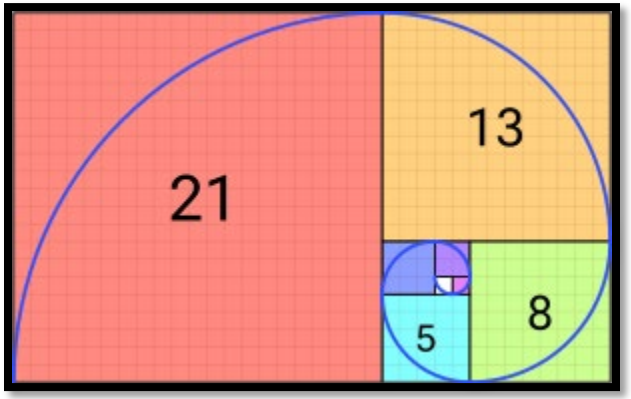
```
for i = 0:10
    disp(i)
end
```

```
for k1 = i1:T1:n1
    for k2 = i2:T2:n2
    .
    .
    commands f (k1 , k2)
    .
    .
    end
end
```

For Loop Example

- Fibonacci Sequence :
  - 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, …
  - $x(1) = 1,\ \ x(2) = 2,\ \ x(n) = x(n-1) + x(n-2)$ for $\mathrm{n} \geq 3$

- Golden Ratio:
  - $\frac{x(n)}{x(n-1)}$



Loops (while loop)

- A while loop is a block of statements that are repeated indefinitely as long as some condition is satisfied.

```
while  dynamic logical expression
    commands
    updating dynamic logical expression
end
```

```
i = 1;
while i<=5
    fprintf('i = %0.0f \n', i)
    i = i + 1;
end
```

if, else, elseif

```
if  logical expression
      commands
end
```

```
N = input('Enter a Number: ');
flag = 0;
if  N > 70
     flag = 1;
end
fprintf('flag is: %0.0f \n', flag);
```

```
if  logical expression
      commands 1
else
      commands 2
end
```

```
N = 10;
if   rem(N,2) == 0
      fprintf('Number is even \n');
else
      fprintf('Number is odd \n');
end
```

```
if  logical expression 1
      commands 1
elseif  logical expression 2
      commands 2
.
.
elseif  logical expression n-1
      commands n-1
else
      commands n
end
```

break and continue

```
for k = i : T : n
    if logical expression
        break  or  continue
    end
    commands f (k)
end
```

```
for i = 1:1:5
    if i == 3
        break  or  continue
    end
    fprintf('i = %0.0f \n', i)
end
```

Functions

- Each ordinary MATLAB function should be placed in a file with the same name (including capitalization) as the function along with the file extension ".m". For example, if a function is named My_fun, that function should be placed in a file named My_fun.m.

Outputs

Function Name

Inputs

```
function [outarg1, outarg2, ...] = fname(inarg1, inarg2, ...)
% H1 comment line
% Other comment lines
...
(Executable code)
...
(return)
(end)
```

Functions

- A function is invoked by naming it in an expression together with a list of actual arguments. A function can be invoked by typing its name directly in the Command Window or by including it in a script file or another function.

```
function [outarg1, outarg2, ...] = fname(inarg1, inarg2, ...)
% H1 comment line
% Other comment lines
...
(Executable code)
...
(return)
(end)
```

```
>> fname (x ,y, …)
```

Thanks!