

Informática Aplicada II

Dr. Alonso Ramirez Manzanares
Depto. de Matemáticas
Univ. de Guanajuato

e-mail: alam@cimat.mx

web: <http://www.cimat.mx/~alam/>

Profesor

- Alonso Ramirez Manzanares, Ciencias de la computación, computo matemático.
 - DEMAT, Universidad de Guanajuato
 - alam@cimat.mx
 - Tel 732006 ext. 1206

- Tendremos un ayudante, el cual está pendiente de asignar, yo les aviso.

Horarios

- Miércoles de 4:00pm a 7:00pm
- Dos descansos de 10 minutos a las 4:50 y a las 5:50
- Nos podemos comunicar por correo entre semana.

Objetivos del curso

- Esta materia esta enfocada para que en cada bachillerato se imparta software afín a la carrera que el alumno desea estudiar en el nivel superior, así como comenzar a aplicar los conocimientos básicos que le serán de utilidad para el transcurso de la vida personal y profesional.
- Esta materia será transversal para las demás materias que lleve en el área terminal y será de mucha importancia aterrizar los conocimientos utilizando el software adecuado para cada una de las áreas terminales.

Competencias

- El alumno aplica software acorde al área terminal como apoyo a la carrera que desea estudiar.
 - Aplica el MatLab para el diseño de algoritmos.
 - Utiliza AutoCad para la realización de planos y diseños diversos.
 - Utiliza programas administrativos para optimizar procesos.

Temario

- Matlab
 - Programación en Matlab
 - variables
 - ciclos
 - funciones (recursividad)
 - Graficación 2D y 3D
 - Algoritmos
 - Manejo de datos (búsquedas)
 - Solución de problemas (optimización)
 - Toolboxes
- AutoCad
- Hojas de Cálculo (Excel)

Referencias

- QA76 .8 M3 T36. TAPIA RODRIGUEZ, MAXIMO, CURSO DE MATLAB / MAXIMO TAPIA RODRIGUEZ, GUANAJUATO, GTO. : CIMAT, 1996
- QA76 .6 P34. PALM III, WILLIAM J. INTRODUCTION TO MATLAB FOR ENGINEERS / WILLIAM J. PALM III. BOSTON ; MCGRAW-HILL. MCGRAW-HILL'S BEST. 1998
- T385 N33. NAKAMURA, SHOICHIRO. ANALISIS NUMERICO Y VERIFICACION GRAFICA CON MATLAB / SHOICHIRO NAKAMURA. MEXICO : PEARSON EDUCATION. 1997

Prerequisitos

- Saber usar la computadoras (e-mail, procesador de textos, chat, navegador)
- Curso básico de C u otro lenguaje de programación
 - (¿Informática Aplicada 1?)
- Funciones de una variable (nivel básico)
- Derivadas (nivel básico)
- Geometría y trigonometría (nivel básico)
- Cálculo de promedios

Modos de Calificación

Modos de Calificación

- **2 Exámenes Parciales 30%**

Modos de Calificación

- **2 Exámenes Parciales 30%**
- **Tareas 35%** Se dejan en la clase y se entregan antes del próximo miércoles (es decir a mas tardar a las 23:59 del martes)

Modos de Calificación

- **2 Exámenes Parciales 30%**
- **Tareas 35%** Se dejan en la clase y se entregan antes del próximo miércoles (es decir a mas tardar a las 23:59 del martes)
- **Proyecto Final 25%**

Modos de Calificación

- **2 Exámenes Parciales 30%**
- **Tareas 35%** Se dejan en la clase y se entregan antes del próximo miércoles (es decir a mas tardar a las 23:59 del martes)
- **Proyecto Final 25%**
- **Asistencia (puntual) 10%** (evaluación con exámenes rápidos)

Pagina WEB del curso

- Aquí encontrarán
 - Diapositivas de la Clase
 - TAREAS
 - FECHAS de EXAMENES
 - Material adicional
 - Ejemplos
- http://www.cimat.mx/~alram/info_apli2/
- ¡Pero es importante que tomen notas!

¿Preguntas?

Ciencias de la computación en Matlab

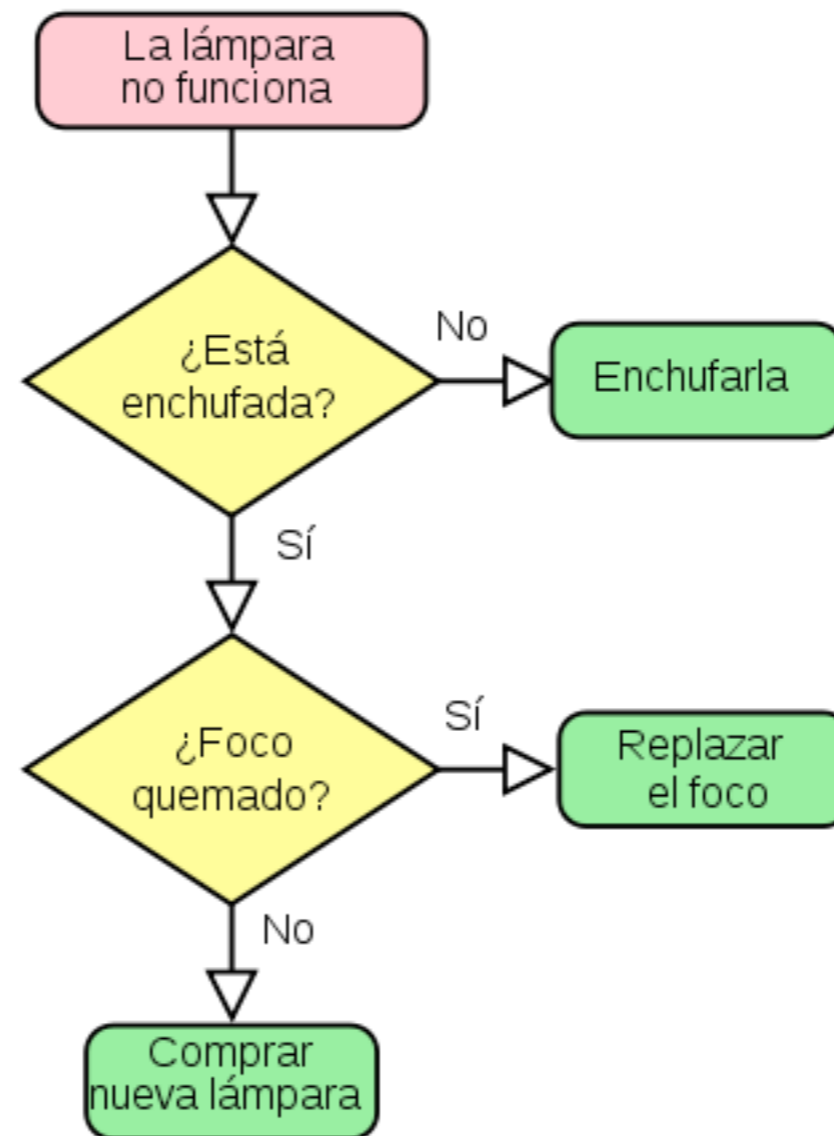
Algoritmos

Algoritmos

- Concepto similar a *receta, proceso, método, técnica, procedimiento o rutina*. Es un conjunto **finito** de reglas que da una secuencia de operaciones para resolver un problema específico.

Diagrama de flujo de algoritmo

(Algoritmo para cambiar una lámpara)



Matlab, la interfase

The screenshot displays the MATLAB 7.8.0 (R2009a) environment. The main window is the Editor, showing a script named `main_polinomios.m` with the following code:

```
1 function main_polinomios
2
3     clc
4
5     % grado del polinomio A
6     nA = 2;
7
8     % definir donde vamos a guardar el polinomio
9     coeficientesA = zeros(1,nA+1);
10
11
12     % definir los coeficientes del polinomio p(x) = x^2 - 1
13     coeficientesA(1) = -1; % -1
14     coeficientesA(2) = 0; % no hay termino para x
15     coeficientesA(3) = 1; % x^2
```

The Command Window below the editor shows the output of the script:

```
25.7289
25.7289
4
-1.0000    0    1.5000   -4.0000    2.0000
```

Red circles highlight the File menu icon, the current directory path in the Editor toolbar, the Run button, and the Command Window output area.

Matlab, la interfase

- Se pueden teclear los comandos en la **consola** >>
- Poner el directorio actual, donde estan los archivos fuente de trabajo
- Guardar los comando en un archivo nuevo **.m**
- Ejecutar los programas con la flecha verde “de play”

Primeros comandos

```
>> x = 1:10
```

```
x =
```

```
     1     2     3     4     5     6     7     8     9    10
```

```
>> y = 3*x
```

```
y =
```

```
     3     6     9    12    15    18    21    24    27    30
```

```
>> plot(x,y)
```

```
fx >>
```

Primeros comandos

```
>> x = 1:10
```

```
x =
```

```
1 2 3 4 5
```

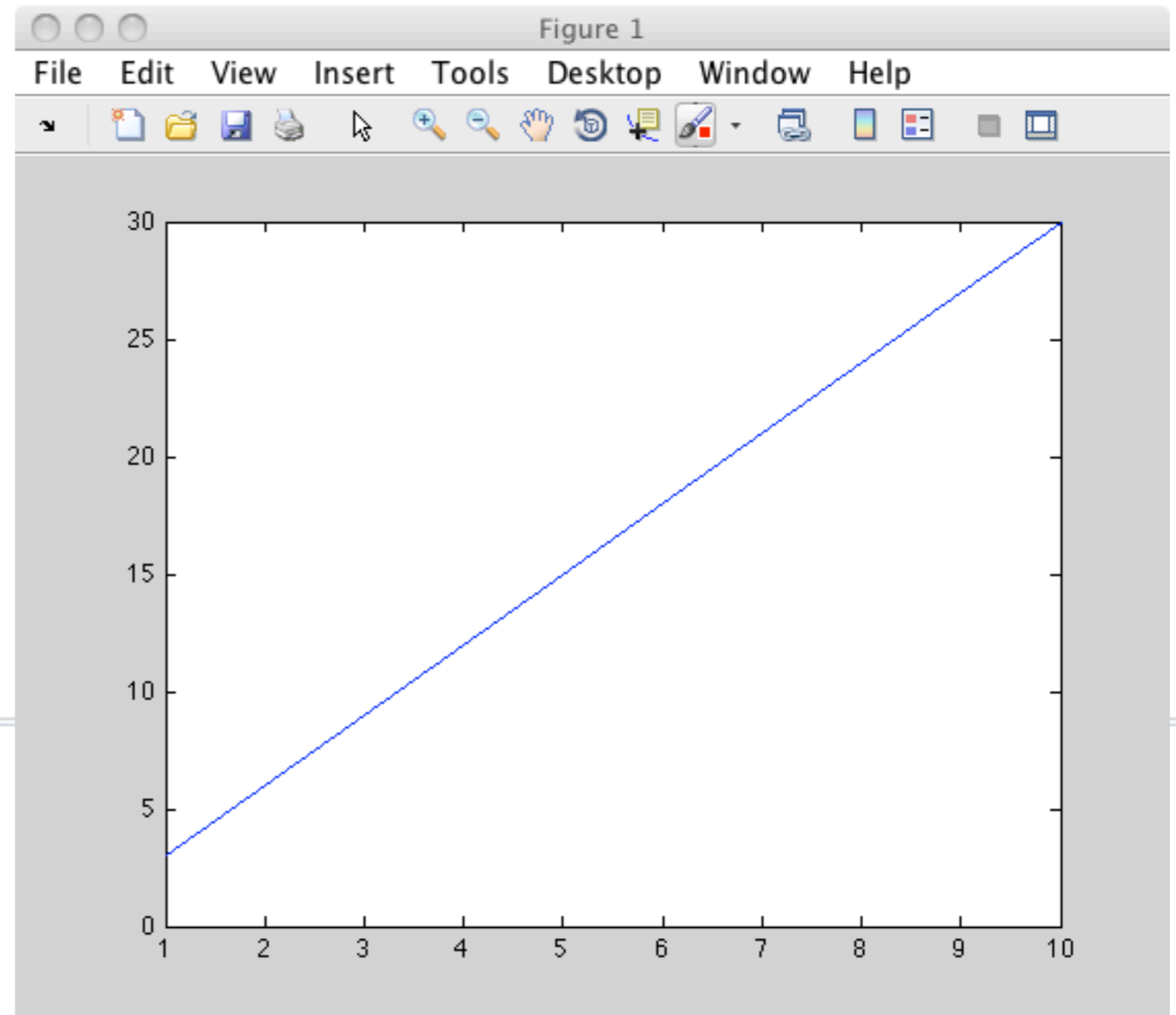
```
>> y = 3*x
```

```
y =
```

```
3 6 9 12 15
```

```
>> plot(x,y)
```

```
fx >>
```



Primeros comandos

- Calculadora

```
>> 2 + 2  
ans =  
    4
```

- Los valores de las variables se quedan en memoria

```
>> a = 4;  
>> a + 2;  
ans =  
    6
```

- Hay muchos mas comandos

```
>> sin(1.4)  
ans = 0.98545  
  
>> sqrt(4)  
ans = 2
```


Primeros comandos

- Los operadores matemáticos básicos se expresan en Matlab mediante los siguientes símbolos:
- Suma: +
- Resta: - (El signo menos también sirve como prefijo para expresar que un número es negativo)
- Multiplicación: *
- División: /
- Potencia: ^

Ayuda

- Si no sabemos como usar algo podemos usar la ayuda

```
>> help sin
SIN      Sine of argument in radians.
        SIN(X) is the sine of the elements of X.

        See also asin, sind.

        Overloaded methods:
            codistributed/sin

        Reference page in Help browser
            doc sin
```

Vectores

- Contenedores de muchos escalares

```
>> v = [11,12,13,14,15,16,17,18,19]
v =
    11    12    13    14    15    16    17 ...
```

- Obtener elementos

```
>> v(2)
ans = 12
```

```
>> v(2:4)
ans =
    12    13    14
```

```
>> v([2,4,6,7])
ans =
    12    14    16    17
```

Operaciones con Vectores

- Operaciones con escalares

```
>> v = [1,2,3,4];  
>> 3+v  
ans =  
  
    4    5    6    7  
>> 3.*v  
ans =  
  
    3    6    9   12
```

- Operaciones vectoriales (con vectores del mismo tamaño)

Operaciones con Vectores

- Operaciones con escalares

```
>> v = [1,2,3,4];
>> 3+v
ans =

     4     5     6     7
>> 3.*v
ans =

     3     6     9    12
```

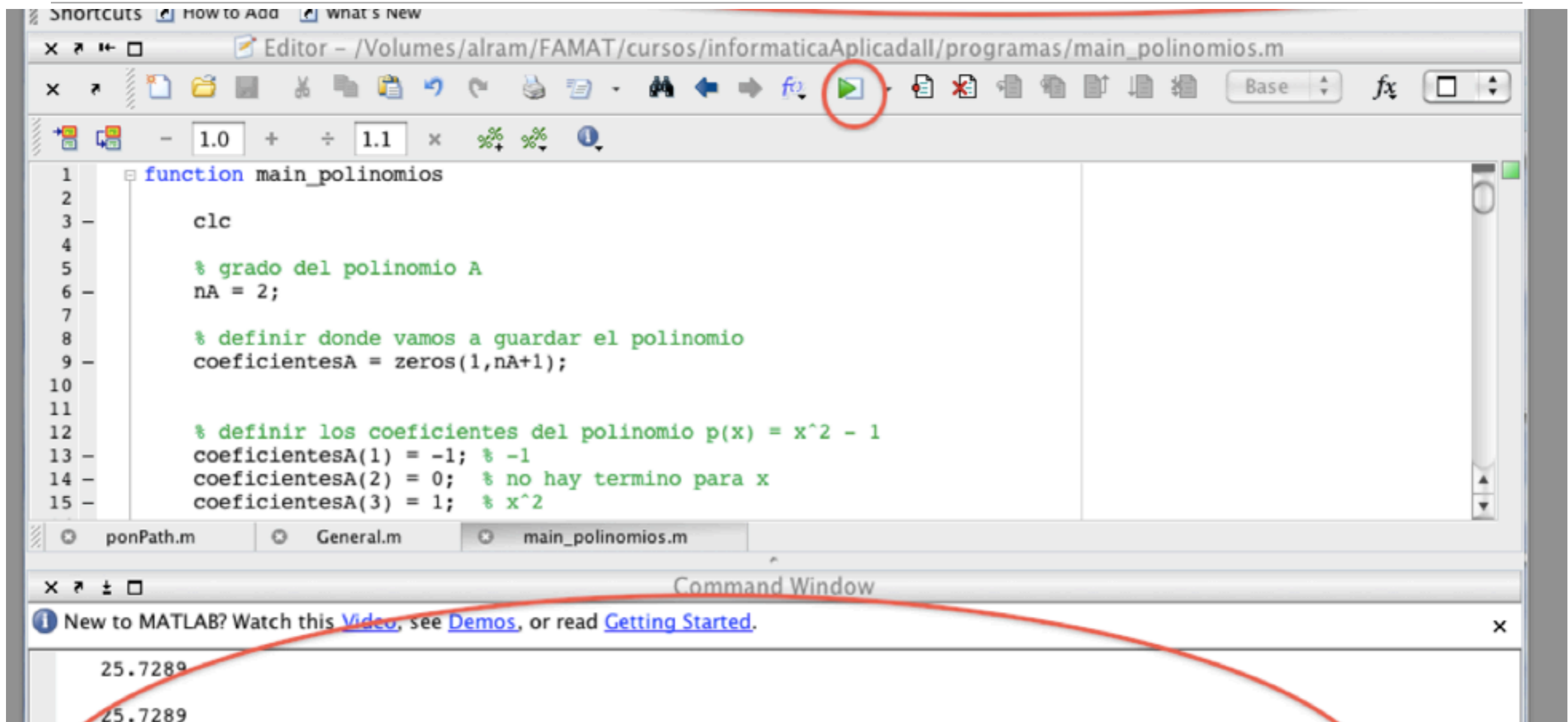
- Operaciones vectoriales (con vectores del mismo tamaño)

```
>> w = [8,7,6,5];
>> v+w
ans =

     9     9     9     9
>> v.*w
ans =

     8    14    18    20
```

Trabajar en el editor, en archivos



Iteradores

- El famoso *for*

```
for i = 1:5  
    disp(i)  
end
```

```
1  
2  
3  
4  
5
```

Iteradores

- El famoso *for*

```
for i = 1:5  
    disp(i)  
end
```

```
1  
2  
3  
4  
5
```

```
for i = [1,4,3,2,9]  
    disp(i)  
end
```

```
1  
4  
3  
2  
9
```


Condicionales

- El condicional If (si)

```
a = zeros(9,1);  
for i = 1:9  
    if i>3  
        a(i) = 2;  
    else  
        a(i) = 0;  
    end  
end  
  
disp(a)
```

Condicionales

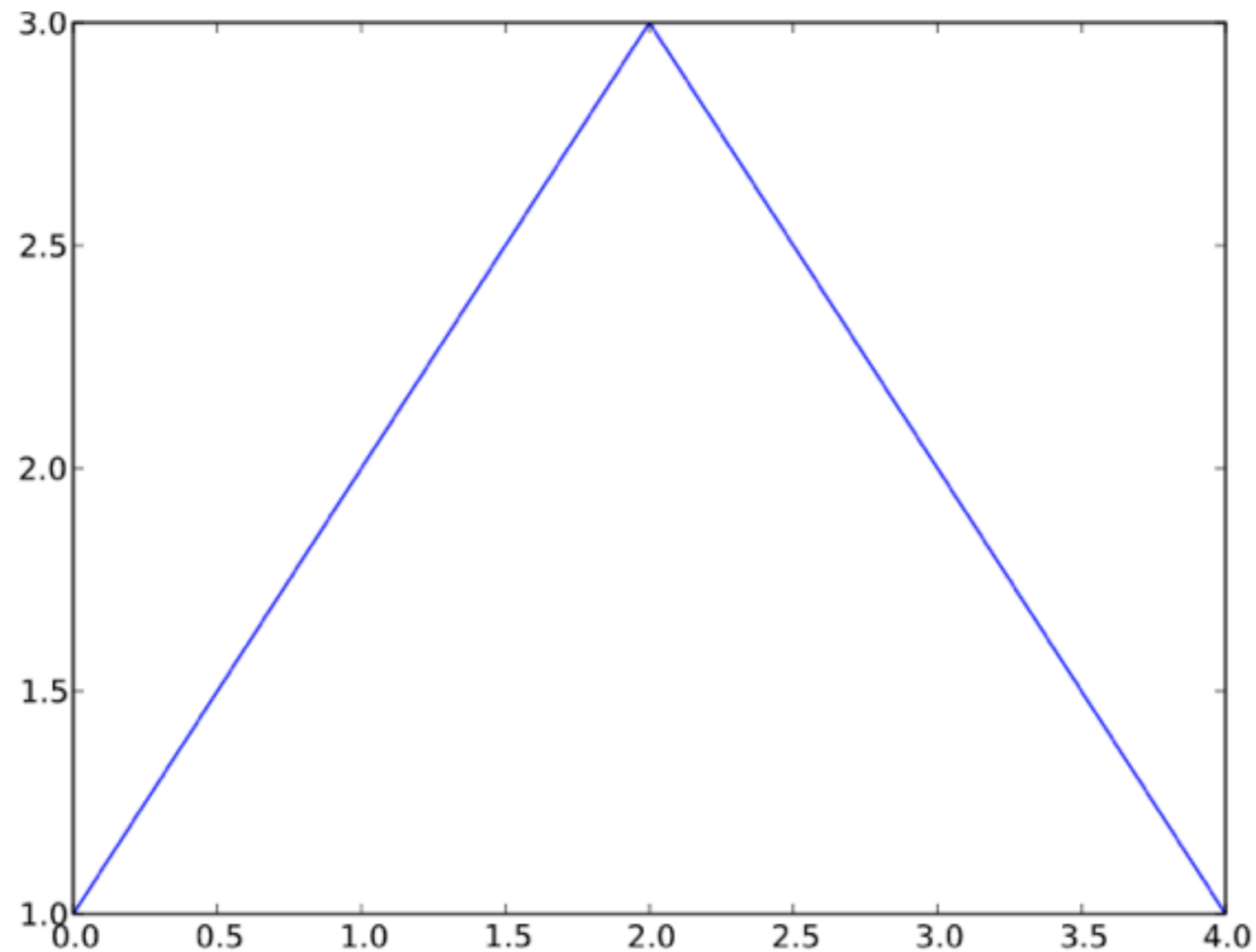
- El condicional If (si)

```
a = zeros(9,1);  
for i = 1:9  
    if i>3  
        a(i) = 2;  
    else  
        a(i) = 0;  
    end  
end  
  
disp(a)
```

```
a = zeros(9,1);  
for i = 1:9  
    if i>3 && i<7  
        a(i) = 2;  
    else  
        a(i) = 0;  
    end  
end  
  
disp(a)
```

Graficación básica

```
>> plot([1,2,3,2,1]);
```

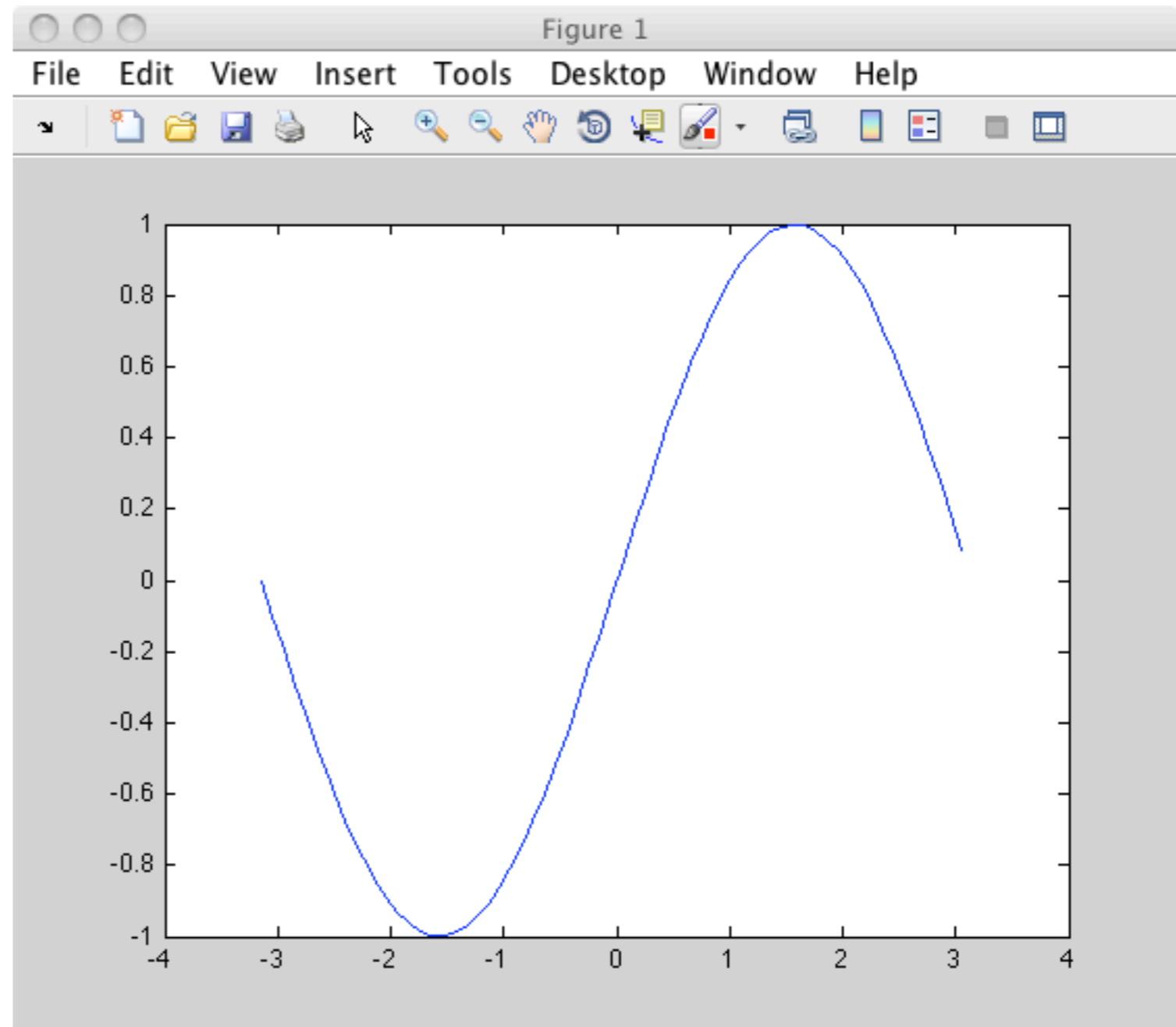


Graficación básica

```
x = -pi:0.1:pi;  
plot(x, sin(x))
```

Graficación básica

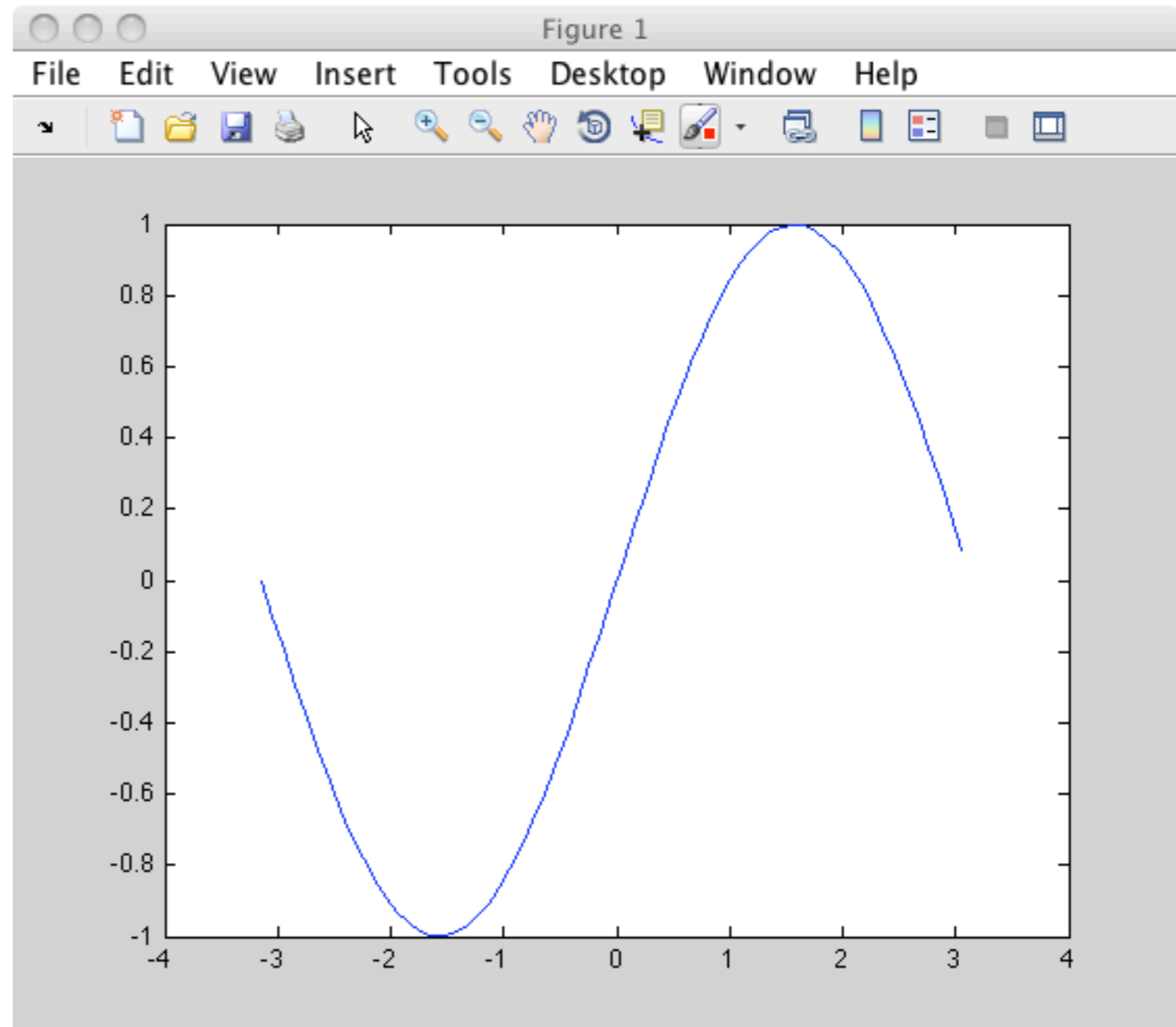
```
x = -pi:0.1:pi;  
plot(x, sin(x))
```



Graficación básica

```
x = -pi:0.1:pi;  
plot(x, sin(x))
```

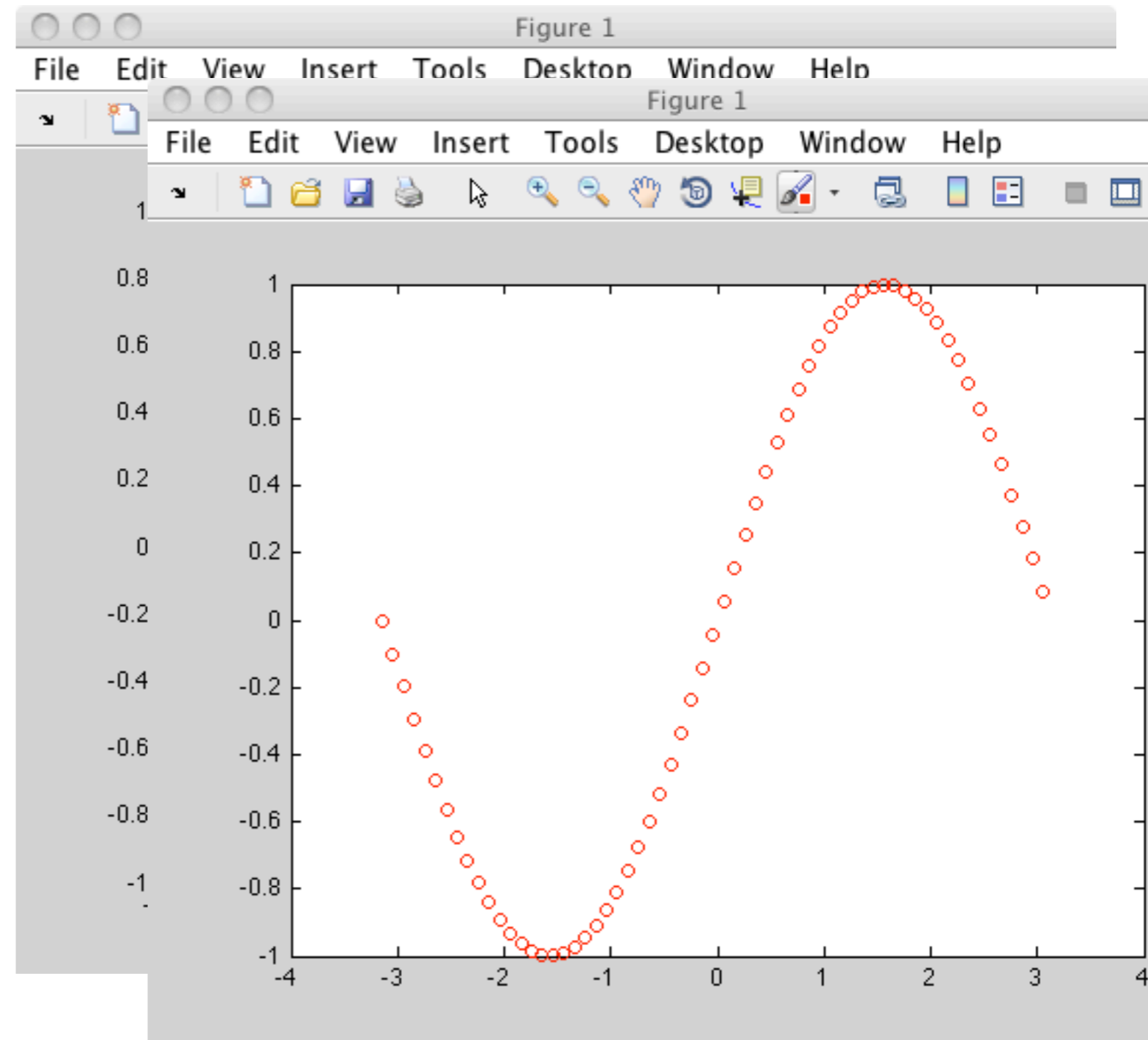
```
plot(x, sin(x), 'ro')
```



Graficación básica

```
x = -pi:0.1:pi;  
plot(x, sin(x))
```

```
plot(x, sin(x), 'ro')
```



Funciones

- Se les pasa parametros (escalares, vectores) y se regresan escalares o vectores

```
function resultado = ordena(x,y)
```

```
    if x < y
```

```
        resultado = [ x y ];
```

```
    else
```

```
        resultado = [ y x ];
```

```
    end
```

```
end
```

```
...
```

```
>> a = ordena(34,4)
```

```
a =
```

```
    4    34
```

Practica I:

Usemos todo lo anterior para trabajar con un polinomio