

# Funciones y Control de Flujo I

## Solución de Problemas con Programación (TC1017)

---

M.C. Xavier Sánchez Díaz  
mail@tec.mx



# Outline

- 1 ¿Qué es una función?
- 2 Funciones en MATLAB
- 3 Consideraciones adicionales

# Definición formal

¿Qué es una función?

## Definition 1

Una **función unitaria** de un conjunto  $A$  en un conjunto  $B$  es cualquier relación binaria  $R$  de  $A$  a  $B$  que satisfaga la condición de que *para todo*  $a \in A$  existe *exactamente un*  $b \in B$  tal que  $(a, b) \in R$ .

Podemos describir una función  $f$  de  $A$  en  $B$  como  $f : A \rightarrow B$ .

## Ejemplo

La relación *sucesor* es una **función** de los naturales en los naturales

$f : \mathbb{N} \rightarrow \mathbb{N}$

$$\text{suc}(n) = \{(1, 2), (2, 3), (3, 4), (4, 5), \dots\}$$

# Definición formal

¿Qué es una función?

## Definition 1

Una **función unitaria** de un conjunto  $A$  en un conjunto  $B$  es cualquier relación binaria  $R$  de  $A$  a  $B$  que satisfaga la condición de que *para todo*  $a \in A$  existe *exactamente un*  $b \in B$  tal que  $(a, b) \in R$ .

Podemos describir una función  $f$  de  $A$  en  $B$  como  $f : A \rightarrow B$ .

## Ejemplo

La relación *sucesor* es una **función** de los naturales en los naturales  
 $f : \mathbb{N} \rightarrow \mathbb{N}$

$$\text{suc}(n) = \{(1, 2), (2, 3), (3, 4), (4, 5), \dots\}$$

# Definición formal

¿Qué es una función?

## Definition 1

Una **función** *unitaria* de un conjunto  $A$  en un conjunto  $B$  es cualquier relación binaria  $R$  de  $A$  a  $B$  que satisfaga la condición de que *para todo*  $a \in A$  existe *exactamente un*  $b \in B$  tal que  $(a, b) \in R$ .

Podemos describir una función  $f$  de  $A$  en  $B$  como  $f : A \rightarrow B$ .

## Ejemplo

La relación *sucesor* es una **función** de los naturales en los naturales

$$f : \mathbb{N} \rightarrow \mathbb{N}$$

$$\text{suc}(n) = \{(1, 2), (2, 3), (3, 4), (4, 5), \dots\}$$

# ..¿Qué?

Qué es una función

## Definition 2

Una caja mágica que al introducirle *ingredientes* devuelve *resultados*.

... con las siguientes condiciones:

- La caja mágica **siempre devuelve un resultado**.
- **Por cada ingrediente**, la caja mágica generará **solamente un resultado**.
  - ▶ Sin embargo, ingredientes distintos pueden generar el mismo resultado.
- **Por cada ingrediente**, la caja mágica generará **siempre el mismo resultado**.

A los ingredientes los llamamos **parámetros** y a los resultados **valores de retorno**.

# ..¿Qué?

Qué es una función

## Definition 2

Una caja mágica que al introducirle *ingredientes* devuelve *resultados*.

... con las siguientes condiciones:

- La caja mágica **siempre devuelve un resultado**.
- **Por cada ingrediente**, la caja mágica generará **solamente un resultado**.
  - ▶ Sin embargo, ingredientes distintos pueden generar el mismo resultado.
- **Por cada ingrediente**, la caja mágica generará **siempre el mismo resultado**.

A los ingredientes los llamamos **parámetros** y a los resultados **valores de retorno**.

# ..¿Qué?

Qué es una función

## Definition 2

Una caja mágica que al introducirle *ingredientes* devuelve *resultados*.

... con las siguientes condiciones:

- La caja mágica **siempre devuelve un resultado**.
- **Por cada ingrediente**, la caja mágica generará **solamente un resultado**.
  - ▶ Sin embargo, ingredientes distintos pueden generar el mismo resultado.
- **Por cada ingrediente**, la caja mágica generará **siempre el mismo resultado**.

A los ingredientes los llamamos **parámetros** y a los resultados **valores de retorno**.



# ..¿Qué?

Qué es una función

## Definition 2

Una caja mágica que al introducirle *ingredientes* devuelve *resultados*.

... con las siguientes condiciones:

- La caja mágica **siempre devuelve un resultado**.
- **Por cada ingrediente**, la caja mágica generará **solamente un resultado**.
  - ▶ Sin embargo, ingredientes distintos pueden generar el mismo resultado.
- **Por cada ingrediente**, la caja mágica generará **siempre el mismo resultado**.

A los ingredientes los llamamos **parámetros** y a los resultados **valores de retorno**.

# ..¿Qué?

Qué es una función

## Definition 2

Una caja mágica que al introducirle *ingredientes* devuelve *resultados*.

... con las siguientes condiciones:

- La caja mágica **siempre devuelve un resultado**.
- **Por cada ingrediente**, la caja mágica generará **solamente un resultado**.
  - ▶ Sin embargo, ingredientes distintos pueden generar el mismo resultado.
- **Por cada ingrediente**, la caja mágica generará **siempre el mismo resultado**.

A los ingredientes los llamamos **parámetros** y a los resultados **valores de retorno**.

# ..¿Qué?

Qué es una función

## Definition 2

Una caja mágica que al introducirle *ingredientes* devuelve *resultados*.

... con las siguientes condiciones:

- La caja mágica **siempre devuelve un resultado**.
- **Por cada ingrediente**, la caja mágica generará **solamente un resultado**.
  - ▶ Sin embargo, ingredientes distintos pueden generar el mismo resultado.
- **Por cada ingrediente**, la caja mágica generará **siempre el mismo resultado**.

A los ingredientes los llamamos **parámetros** y a los resultados **valores de retorno**.

# Ejemplos

¿Qué es una función?

La función  $f(x) = 2x - 3$  es una función porque, **usando cualquier número real...**

- Siempre devuelve un resultado
- Nos da solamente un resultado
- Nos da siempre el mismo resultado

# Ejemplos

¿Qué es una función?

La función  $f(x) = 2x - 3$  es una función porque, **usando cualquier número real...**

- Siempre devuelve un resultado
- Nos da solamente un resultado
- Nos da siempre el mismo resultado

# Ejemplos

¿Qué es una función?

La función  $f(x) = 2x - 3$  es una función porque, **usando cualquier número real...**

- Siempre devuelve un resultado
- Nos da solamente un resultado
- Nos da siempre el mismo resultado

# Ejemplos

¿Qué es una función?

La función  $f(x) = 2x - 3$  es una función porque, **usando cualquier número real...**

- Siempre devuelve un resultado
- Nos da solamente un resultado
- Nos da siempre el mismo resultado

# Ejemplos

¿Qué es una función?

La función  $\sin(x)$  es una función porque, usando cualquier número real. . .

- Siempre devuelve un resultado
- Nos da solamente un resultado
- Nos da siempre el mismo resultado



# Ejemplos

¿Qué es una función?

La función  $\sqrt{x}$  es una función porque, usando cualquier número real...

- Siempre devuelve un resultado
- Nos da solamente un resultado
- Nos da siempre el mismo resultado

# Elementos de una función

¿Qué es una función

Cuando la **definimos** . . .

$$f(x) = 7x^2 + 17x - 3$$

- Nombre de la función
- Parámetro
- Encabezado de la función
- Cuerpo de la función

# Elementos de una función

¿Qué es una función

Cuando la **definimos** . . .

$$f(x) = 7x^2 + 17x - 3$$

- Nombre de la función
- Parámetro
- Encabezado de la función
- Cuerpo de la función

# Elementos de una función

¿Qué es una función

Cuando la **definimos** . . .

$$f(\textcolor{red}{x}) = 7x^2 + 17x - 3$$

- Nombre de la función
- **Parámetro**
- Encabezado de la función
- Cuerpo de la función

# Elementos de una función

¿Qué es una función

Cuando la **definimos** . . .

$$f(x) = 7x^2 + 17x - 3$$

- Nombre de la función
- Parámetro
- Encabezado de la función
- Cuerpo de la función

# Elementos de una función

¿Qué es una función

Cuando la **definimos** . . .

$$f(x) = 7x^2 + 17x - 3$$

- Nombre de la función
- Parámetro
- Encabezado de la función
- **Cuerpo de la función**

# Elementos de una función

¿Qué es una función

Cuando la **evaluamos** por ejemplo, con  $x = 2 \dots$

$$f(2) = 7(2)^2 + 17(2) - 3 = 59$$

- Nombre de la función
- Argumento
- Llamada o evaluación de la función
- Valor de retorno

# Elementos de una función

¿Qué es una función

Cuando la **evaluamos** por ejemplo, con  $x = 2 \dots$

$$f(2) = 7(2)^2 + 17(2) - 3 = 59$$

- Nombre de la función
- Argumento
- Llamada o evaluación de la función
- Valor de retorno



# Elementos de una función

¿Qué es una función

Cuando la **evaluamos** por ejemplo, con  $x = 2 \dots$

$$f(2) = 7(2)^2 + 17(2) - 3 = 59$$

- Nombre de la función
- **Argumento**
- Llamada o evaluación de la función
- Valor de retorno

# Elementos de una función

¿Qué es una función

Cuando la **evaluamos** por ejemplo, con  $x = 2 \dots$

$$f(2) = 7(2)^2 + 17(2) - 3 = 59$$

- Nombre de la función
- Argumento
- Llamada o evaluación de la función
- Valor de retorno

# Elementos de una función

¿Qué es una función

Cuando la **evaluamos** por ejemplo, con  $x = 2 \dots$

$$f(2) = 7(2)^2 + 17(2) - 3 = 59$$

- Nombre de la función
- Argumento
- Llamada o evaluación de la función
- Valor de retorno

# Manos a la obra

## Funciones en MATLAB

Todos los comandos en MATLAB están dados de alta internamente en el software como **funciones**.

- `sin` necesita un parámetro para operar: `sin 67` significa `sin(67)`
- `sqrt` necesita un parámetro para operar: `sqrt 2` significa  $\sqrt{2}$
- Tanto `sin` como `sqrt` cumplen con las reglas de siempre devolver resultados, sólo devolver un resultado, y devolver siempre el mismo.

# Manos a la obra

## Funciones en MATLAB

Todos los comandos en MATLAB están dados de alta internamente en el software como **funciones**.

- `sin` necesita un parámetro para operar: `sin 67` significa `sin(67)`
- `sqrt` necesita un parámetro para operar: `sqrt 2` significa  $\sqrt{2}$
- Tanto `sin` como `sqrt` cumplen con las reglas de siempre devolver resultados, sólo devolver un resultado, y devolver siempre el mismo.

# Manos a la obra

## Funciones en MATLAB

Todos los comandos en MATLAB están dados de alta internamente en el software como **funciones**.

- `sin` necesita un parámetro para operar: `sin 67` significa `sin(67)`
- `sqrt` necesita un parámetro para operar: `squert 2` significa  $\sqrt{2}$
- Tanto `sin` como `squert` cumplen con las reglas de siempre devolver resultados, sólo devolver un resultado, y devolver siempre el mismo.

# Manos a la obra

## Funciones en MATLAB

Todos los comandos en MATLAB están dados de alta internamente en el software como **funciones**.

- `sin` necesita un parámetro para operar: `sin 67` significa `sin(67)`
- `sqrt` necesita un parámetro para operar: `sqrt 2` significa  $\sqrt{2}$
- Tanto `sin` como `sqrt` cumplen con las reglas de siempre devolver resultados, sólo devolver un resultado, y devolver siempre el mismo.

# La palabra reservada `function`

## Funciones en MATLAB

Para implementar funciones en MATLAB es necesario utilizar la palabra reservada `function`.

En el editor:

---

```
1 function <return_value> = <function_name>(<parameter>)  
2   <body of the function>  
3 end
```

---



# La palabra reservada `function`

## Funciones en MATLAB

Editor:

---

```
1 function result = successor(x)
2     result = x + 1;
3 end
```

---

Command Window:

---

```
>> successor(9)
```

```
ans =
```

```
    10
```

---

# La palabra reservada `function`

## Funciones en MATLAB

Editor:

---

```
1 function output = donothing(x)
2     output = x;
3 end
```

---

Command Window:

---

```
>> donothing(5)
```

```
ans =
```

```
5
```

---

# La palabra reservada `function`

## Funciones en MATLAB

Editor:

---

```
1 function y = fancyname(x)
2     y = 2 * x ^ 2 + 5 * x - 3;
3 end
```

---

Command Window:

---

```
>> fancyname(2)
```

```
ans =
```

```
15
```

---

# Funciones de más de un parámetro

## Consideraciones adicionales

Hasta ahora hemos visto funciones *unitarias*:

$$f(x) = 2x$$

Sin embargo, también puede haber funciones con una *aridad* mayor a uno:

$$f(x, y) = 7x + 3y$$

¿Se te ocurre alguna función matemática *con nombre* que reciba más de un parámetro?

# Funciones de más de un parámetro

## Consideraciones adicionales

Hasta ahora hemos visto funciones *unitarias*:

$$f(x) = 2x$$

Sin embargo, también puede haber funciones con una **aridad** mayor a uno:

$$f(x, y) = 7x + 3y$$

¿Se te ocurre alguna función matemática *con nombre* que reciba más de un parámetro?

# Funciones de más de un parámetro

## Consideraciones adicionales

Hasta ahora hemos visto funciones *unitarias*:

$$f(x) = 2x$$

Sin embargo, también puede haber funciones con una **aridad** mayor a uno:

$$f(x, y) = 7x + 3y$$

¿Se te ocurre alguna función matemática *con nombre* que reciba más de un parámetro?

# Más preguntas interesantes con funciones

## Consideraciones adicionales

Si podemos tener más de un parámetro, ¿podemos tener más de un valor de retorno?

*Sí, usualmente ordenados y agrupados. Esto lo veremos más adelante, en estructuras de datos básicas.*

¿Puedo hacer una función que no me dé resultado alguno?

*Sí, de hecho son muy frecuentes. Sin embargo, si no devuelve nada, ya no es propiamente una función. A estas funciones sin valor de retorno les llamamos **procedimientos** (procedures en inglés).*

¿Puedo hacer una función que no reciba un parámetro?

*Sí, sí puedes. Sin embargo no tiene mucho sentido. Lo recomendable es simplemente hacer una secuencia de instrucciones. A esta serie de instrucciones les llamamos **scripts**.*

# Más preguntas interesantes con funciones

## Consideraciones adicionales

Si podemos tener más de un parámetro, ¿podemos tener más de un valor de retorno?

*Sí, usualmente ordenados y agrupados. Esto lo veremos más adelante, en estructuras de datos básicas.*

¿Puedo hacer una función que no me dé resultado alguno?

*Sí, de hecho son muy frecuentes. Sin embargo, si no devuelve nada, ya no es propiamente una función. A estas funciones sin valor de retorno les llamamos **procedimientos** (procedures en inglés).*

¿Puedo hacer una función que no reciba un parámetro?

*Sí, sí puedes. Sin embargo no tiene mucho sentido. Lo recomendable es simplemente hacer una secuencia de instrucciones. A esta serie de instrucciones les llamamos **scripts**.*



# Más preguntas interesantes con funciones

## Consideraciones adicionales

Si podemos tener más de un parámetro, ¿podemos tener más de un valor de retorno?

*Sí, usualmente ordenados y agrupados. Esto lo veremos más adelante, en estructuras de datos básicas.*

¿Puedo hacer una función que no me dé resultado alguno?

*Sí, de hecho son muy frecuentes. Sin embargo, si no devuelve nada, ya no es propiamente una función. A estas funciones sin valor de retorno les llamamos **procedimientos** (procedures en inglés).*

¿Puedo hacer una función que no reciba un parámetro?

*Sí, sí puedes. Sin embargo no tiene mucho sentido. Lo recomendable es simplemente hacer una secuencia de instrucciones. A esta serie de instrucciones les llamamos **scripts**.*

# Más preguntas interesantes con funciones

## Consideraciones adicionales

Si podemos tener más de un parámetro, ¿podemos tener más de un valor de retorno?

*Sí, usualmente ordenados y agrupados. Esto lo veremos más adelante, en estructuras de datos básicas.*

¿Puedo hacer una función que no me dé resultado alguno?

*Sí, de hecho son muy frecuentes. Sin embargo, si no devuelve nada, ya no es propiamente una función. A estas funciones sin valor de retorno les llamamos **procedimientos** (procedures en inglés).*

¿Puedo hacer una función que no reciba un parámetro?

*Sí, sí puedes. Sin embargo no tiene mucho sentido. Lo recomendable es simplemente hacer una secuencia de instrucciones. A esta serie de instrucciones les llamamos **scripts**.*

# Más preguntas interesantes con funciones

## Consideraciones adicionales

Si podemos tener más de un parámetro, ¿podemos tener más de un valor de retorno?

*Sí, usualmente ordenados y agrupados. Esto lo veremos más adelante, en estructuras de datos básicas.*

¿Puedo hacer una función que no me dé resultado alguno?

*Sí, de hecho son muy frecuentes. Sin embargo, si no devuelve nada, ya no es propiamente una función. A estas funciones sin valor de retorno les llamamos **procedimientos** (procedures en inglés).*

¿Puedo hacer una función que no reciba un parámetro?

*Sí, sí puedes. Sin embargo no tiene mucho sentido. Lo recomendable es simplemente hacer una secuencia de instrucciones. A esta serie de instrucciones les llamamos **scripts**.*

# Más preguntas interesantes con funciones

## Consideraciones adicionales

Si podemos tener más de un parámetro, ¿podemos tener más de un valor de retorno?

*Sí, usualmente ordenados y agrupados. Esto lo veremos más adelante, en estructuras de datos básicas.*

¿Puedo hacer una función que no me dé resultado alguno?

*Sí, de hecho son muy frecuentes. Sin embargo, si no devuelve nada, ya no es propiamente una función. A estas funciones sin valor de retorno les llamamos **procedimientos** (procedures en inglés).*

¿Puedo hacer una función que no reciba un parámetro?

*Sí, sí puedes. Sin embargo no tiene mucho sentido. Lo recomendable es simplemente hacer una secuencia de instrucciones. A esta serie de instrucciones les llamamos **scripts**.*

# En cómputo, todo es una función

## Consideraciones adicionales

Específicamente en MATLAB, tenemos algunas redundancias para facilitar la escritura:

- `plus (1, 2)` es lo mismo que `1 + 2`
- `a == 5` es lo mismo que `eq(a, 5)`
- `mtimes(5, 2)` es otra manera de decir `5 * 2`

¿Qué otros operadores conocemos?

# En cómputo, todo es una función

## Consideraciones adicionales

Específicamente en MATLAB, tenemos algunas redundancias para facilitar la escritura:

- `plus (1, 2)` es lo mismo que `1 + 2`
- `a == 5` es lo mismo que `eq(a, 5)`
- `mtimes(5, 2)` es otra manera de decir `5 * 2`

¿Qué otros operadores conocemos?

# En cómputo, todo es una función

## Consideraciones adicionales

Específicamente en MATLAB, tenemos algunas redundancias para facilitar la escritura:

- `plus (1, 2)` es lo mismo que `1 + 2`
- `a == 5` es lo mismo que `eq(a, 5)`
- `mtimes(5, 2)` es otra manera de decir `5 * 2`

¿Qué otros operadores conocemos?

# En cómputo, todo es una función

## Consideraciones adicionales

Específicamente en MATLAB, tenemos algunas redundancias para facilitar la escritura:

- `plus (1, 2)` es lo mismo que `1 + 2`
- `a == 5` es lo mismo que `eq(a, 5)`
- `mtimes(5, 2)` es otra manera de decir `5 * 2`

¿Qué otros operadores conocemos?



# En cómputo, todo es una función

## Consideraciones adicionales

Específicamente en MATLAB, tenemos algunas redundancias para facilitar la escritura:

- `plus (1, 2)` es lo mismo que `1 + 2`
- `a == 5` es lo mismo que `eq(a, 5)`
- `mtimes(5, 2)` es otra manera de decir `5 * 2`

¿Qué otros operadores conocemos?

# Funciones seccionadas (Piecewise)

## Consideraciones adicionales

Hasta ahora, todas las funciones que hemos visto son hermosas porque son continuas y por tanto diferenciables. Sin embargo, no todas las funciones son así...

¿Cuál es el valor absoluto de 2? ¿Y el valor absoluto de  $-2$ ?

$$|x| = \begin{cases} -x, & \text{si } x < 0 \\ x, & \text{si } x \geq 0 \end{cases}$$

¿Cómo representamos esto en MATLAB?

# Funciones seccionadas (Piecewise)

## Consideraciones adicionales

Hasta ahora, todas las funciones que hemos visto son hermosas porque son continuas y por tanto diferenciables. Sin embargo, no todas las funciones son así...

¿Cuál es el valor absoluto de 2? ¿Y el valor absoluto de  $-2$ ?

$$|x| = \begin{cases} -x, & \text{si } x < 0 \\ x, & \text{si } x \geq 0 \end{cases}$$

¿Cómo representamos esto en MATLAB?

# Funciones seccionadas (Piecewise)

## Consideraciones adicionales

Hasta ahora, todas las funciones que hemos visto son hermosas porque son continuas y por tanto diferenciables. Sin embargo, no todas las funciones son así...

¿Cuál es el valor absoluto de 2? ¿Y el valor absoluto de  $-2$ ?

$$|x| = \begin{cases} -x, & \text{si } x < 0 \\ x, & \text{si } x \geq 0 \end{cases}$$

¿Cómo representamos esto en MATLAB?

# Funciones seccionadas (Piecewise)

## Consideraciones adicionales

Hasta ahora, todas las funciones que hemos visto son hermosas porque son continuas y por tanto diferenciables. Sin embargo, no todas las funciones son así...

¿Cuál es el valor absoluto de 2? ¿Y el valor absoluto de  $-2$ ?

$$|x| = \begin{cases} -x, & \text{si } x < 0 \\ x, & \text{si } x \geq 0 \end{cases}$$

¿Cómo representamos esto en MATLAB?

# Funciones seccionadas (Piecewise)

## Consideraciones adicionales

Hasta ahora, todas las funciones que hemos visto son hermosas porque son continuas y por tanto diferenciables. Sin embargo, no todas las funciones son así. . .

¿Cuál es el valor absoluto de 2? ¿Y el valor absoluto de  $-2$ ?

$$|x| = \begin{cases} -x, & \text{si } x < 0 \\ x, & \text{si } x \geq 0 \end{cases}$$

¿Cómo representamos esto en MATLAB?

# Primer encuentro con el control de flujo

## Consideraciones adicionales

Los **condicionales** son las **estructuras de control** de flujo más comunes.

---

```
1  if <condition>
2      % Si se cumple
3      <do something>
4  else
5      % Si no se cumple
6      <do something else>
7  end
```

---

¿Puedes implementar la función del valor absoluto en MATLAB?