

# Práctica 1

## Índice

1. Acceso al ordenador	1
2. Acceso al programa	1
3. Cálculos numéricos básicos	3
4. Funciones	4
5. Ejercicios	6

## 1. Acceso al ordenador

Para acceder a los ordenadores en los laboratorios de docencia se necesita una identificación. Los datos de entrada son:

**Nombre del usuario:** Usuariocentralizado@udc.es (Por ejemplo: marta.lopez@udc.es)

**Contraseña:** la de servicios centrales de la UDC (es la misma que para el correo electrónico.)

Existe además un acceso provisional a los ordenadores, que sirve para el alumnado que no tenga todavía gestionada la matrícula y no disponga de usuario de servicios de la UDC. También se puede utilizar en caso de que algún día no funcione en la red el acceso a servicios centrales.

**Nombre de usuario:** nombredelamaquina\eup (Por ejemplo: CAD07\eup)

**Contraseña:** local

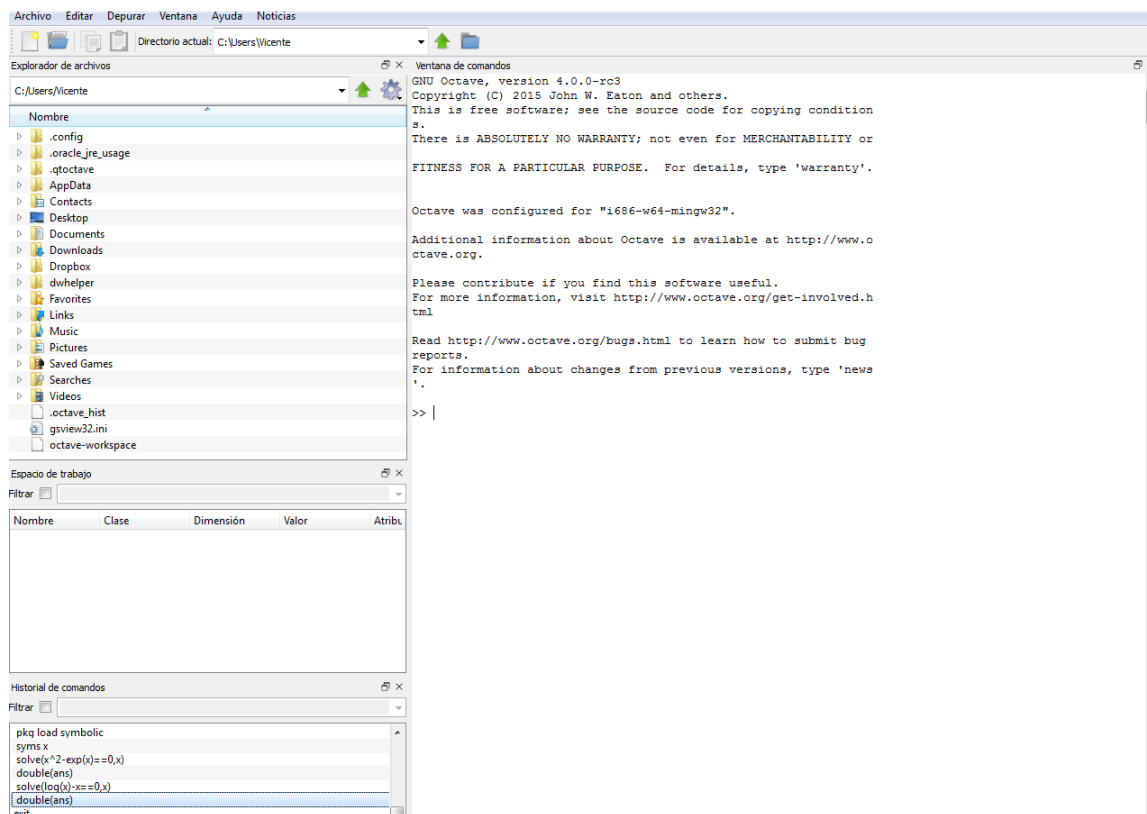
El `nombredelamaquina` de cada ordenador de los laboratorios figura en una etiqueta visible con el nombre de la máquina.

Cuando un usuario se identifica por primera vez en un ordenador de la EUP, se recoge y graba el perfil por defecto de la máquina local del laboratorio, conservándose posteriormente en el servidor de Servicios Centrales de la UDC.

## 2. Acceso al programa

Al ejecutar el comando `QtOctave` desde un terminal, o *clicando* dos veces sobre el icono situado en el escritorio, aparece en la pantalla el entorno gráfico de trabajo de OCTAVE (ver figura 1).

En la pantalla inicial de `Octave` aparecen cuatro ventanas:



**Figura 1:** Pantalla inicial

- **Ventana de comandos:** en ella se pueden escribir los comandos que se desea ejecutar. Para evaluar una expresión se debe escribir ésta tras los símbolos `>>` y pulsar **Enter**. Los resultados aparecerán escritos a continuación y serán guardados en una variable, llamada `ans`.
- **Editor:** en esta ventana se pueden introducir varias líneas de comandos y compilarlas conjuntamente. Para eso se deben seleccionar todas las líneas que se deseen compilar y pulsar la tecla **F9**. El resultado de la ejecución aparecerá en la ventana de comandos.
- **Explorador de archivos:** aparecen en ella todos los archivos usados.
- **Espacio de trabajo:** aparecen en ella todas las variables definidas.
- **Historial de comandos:** proporciona un listado de los comandos utilizados en la sesión.

A lo largo de este curso usaremos, además de las funciones estándar ya instaladas en `OCTAVE`, algunas que aparecen incluidas en el paquete `SYMBOLIC`. Para poder trabajar con estas funciones se debe teclear en la ventana de comandos:

```
>> pkg load symbolic
```

### 3. Cálculos numéricos básicos

Las operaciones aritméticas habituales están definidas en `OCTAVE`. Estas son:

operación	símbolo
suma	+
resta	-
producto	*
división	/
potenciación	^

Sus prioridades son 1º: ^, 2º: \* o /, y 3º: + o -. Estas prioridades se pueden cambiar con el uso de paréntesis.

Cada vez que introduzcamos una expresión (por ejemplo si quisiéramos sumar  $2 + 3$ ) aparecerá en la ventana de comandos el símbolo de entrada

```
>> 2+3
```

mientras que cada vez que `OCTAVE` calcule un resultado, mostrará como salida

```
ans = 5
```

Para evaluar las expresiones en una sesión de trabajo debemos pulsar la tecla `Enter`. Como podemos comprobar, a las expresiones de salida se les asigna por defecto la variable `ans`, que cada vez que se calcula una nueva expresión va cambiando su valor. En general, cada vez que escribamos una expresión matemática en `OCTAVE` podemos finalizarla usando un punto y coma “;” para evitar que aparezca el resultado de los cálculos en la pantalla.

Al contrario de lo que ocurre en otros lenguajes de programación, las variables no es necesario declararlas con un tipo específico, simplemente hay que asignarles un valor. Podemos asignarle nombres a las variables a las que nos referiremos posteriormente. El nombre de una variable debe ser una secuencia de letras, dígitos o guiones bajos, pero no puede comenzar con un dígito. A lo largo de este curso trabajaremos con variables numéricas y simbólicas. Por ejemplo, si introducimos las siguientes expresiones desde la línea de comandos:

#### Editor

```
a=0.3333  
b=sym(1/3)  
syms x
```

que nos informa que `a` es una variable numérica con un valor determinado, `b` es una variable simbólica con un valor determinado y `x` es una variable simbólica que se usará para definir funciones.

Veremos en el siguiente ejemplo que `OCTAVE` siempre calcula los resultados exactos en el caso de las variables simbólicas (por muy larga que sea la expresión) mientras que cuando los cálculos involucran variables numéricas el resultado de la evaluación de las funciones siempre va a ser una aproximación decimal. En el caso de necesitarse el valor numérico de una expresión simbólica con 4 dígitos de precisión, debemos emplear el comando `double`. Por ejemplo:

**Ventana de comandos**

```
>> a=1/3
a = 0.33333
>> b=sym(1/3)
b = (sym) 1/3
>> a+1
ans = 1.3333
>> b+1
ans = (sym) 4/3
>> double(b+1)
ans = 1.3333
```

A lo largo del curso emplearemos habitualmente constantes numéricas que ya están predefinidas en `OCTAVE`: el número  $e$  y la unidad imaginaria  $i$ , y el número  $\pi$ , que se escribirá en la forma `pi`.

Por último, debemos señalar que el comando `clear` nos permite eliminar las variables que tenemos definidas a lo largo de nuestra sesión de trabajo. Por ejemplo, si queremos eliminar la variable  $x$  teclearíamos `clear x`, mientras que si queremos eliminar todas las variables con las que trabajamos en una sesión escribiríamos `clear all`. Cuando queramos realizar sucesivamente varias tareas en `OCTAVE`, es conveniente crear un fichero `.m` desde el menú `Editor` (que se puede lanzar desde el botón de menú de `OCTAVE`). Una vez guardado este fichero de comandos, se pueden volver a ejecutar todos los comandos contenidos en él abriéndolo de nuevo desde `OCTAVE`. Para facilitar la lectura de este fichero se pueden incluir líneas de texto con comentarios (describiendo el código, explicando el uso de cada comando, etc). Todas las líneas de comentarios deben comenzar por `%` o `#`.

## 4. Funciones

`OCTAVE` tiene definidas las funciones habituales: las funciones trigonométricas `sin`, `cos`, `tan`, el logaritmo neperiano `log`, la raíz cuadrada `sqrt`, la función exponencial `exp`, etc. Además de trabajar con expresiones, ya sean variables o constantes, también podemos definir otras funciones, dependientes de una o varias variables.

Existen dos formas de definir funciones: la definición en línea y la simbólica.

- Función definidas en línea.

Se utilizará el comando `@( )`, especificando entre paréntesis las variables de las que depende la función y escribiendo a continuación la ecuación de la función. Por ejemplo, para definir la función  $f(x) = x^2$  se debe escribir

```
>> f = @(x) x^2;
```

Ahora podríamos calcular el valor de la función en cualquier punto:

```
>> f(3)
ans=9
```

- Funciones definidas de forma simbólica.

En primer lugar, debemos definir las variables simbólicas usando el comando `syms`. A continuación se define la función. Por ejemplo, si quisiésemos definir  $f(x) = x^2$  tan solo tendríamos que escribir:

```
>> syms x;
f=x^2
```

Ahora podríamos calcular el valor de la función en cualquier punto:

```
>> subs(f,x,2)
ans= 4
```

Se puede definir ahora una función escalar, por ejemplo  $f(x,y) = x^2 + 2y + 1$ , y evaluarla en el punto  $(2,3)$ .

#### editor

```
f = @(x,y) x^2+2*y+1;
f(2,3)
```

#### Ventana de comandos

```
>> f = @(x,y) x^2+2*y+1;
>> f(2,3)
ans = 11
```

También se puede realizar el cálculo con variables simbólicas.

#### editor

```
syms x y
f=x^2+2*y+1
subs(f,{x,y},{2,3})
```

#### Ventana de comandos

```
>> syms x y
>> f=x^2+2*y+1
f = (sym)
      2
      x  + 2*y + 1
>> subs(f,{x,y},{2,3})
ans = (sym) 11
```

---

## 5. Ejercicios

**Ejercicio 1** Realiza las siguientes operaciones y determina cuál fue el orden de evaluación de las operaciones aritméticas:

$$\begin{aligned} &2.01 * 4 + 3.1416 \\ &-2.98 + 0.23 * 14 + 2 \\ &6 + 4/2 + 3.111 \\ &5.22 * 3.1416/6 - 4 \end{aligned}$$

**Ejercicio 2** Realiza las siguientes operaciones (modifica como sea necesario la precedencia de los operadores con paréntesis) con variables numéricas:

$$\frac{3 + 4^2}{\frac{2}{\sqrt[5]{3}} - \left(\frac{1}{3.1 \times 2}\right)^{\frac{3}{4}}} \qquad \frac{4.1^{\frac{0.2+1}{2}}}{\frac{2}{0.1^{\frac{1}{2}}} - \frac{0.4}{2^{\frac{1}{3}}}}$$

**Ejercicio 3** Sabiendo que  $a = 21/10$ ,  $b = 3$ , calcula los valores numéricos de las siguientes expresiones definiendo las variables  $a$  y  $b$  como simbólicas:  $a + bi$ ,  $\sqrt{2}$ ,  $\sqrt{7a/5}$ ,  $\ln(2a)$  y  $7e^{5/4} + 3.4$ .

**Ejercicio 4** Calcular  $\text{sen}(\pi)$  e interpretar el resultado proporcionado por el programa.