



CÁLCULO NUMÉRICO 1 VARIABLE CURSO 2017-2018  
2 de octubre de 2017.

PRÁCTICA 2

En esta práctica vamos a

- Manipular “Funciones”.
- Utilizar “Paneles” donde dibujar gráficas de funciones.
- Importar una clase `Complex` para manejar “números complejos”.

**Ejercicio 0** Utiliza la opción *Renombrar/Refactor* sobre el paquete *practica1* para renombrarlo como *cn1vPractica1*. Crea en tu proyecto el paquete *cn1vPractica2* donde ir poniendo los ficheros de esta práctica.

Baja de la zona de recursos el fichero comprimido *ParaPractica2.zip*, descomprímelo en el escritorio y espera a los siguientes ejercicios para ir incorporando ficheros a tu proyecto.

**Ejercicio 1 (Uso de “interfaces”)**

1. Crea un paquete auxiliar en tu proyecto.

Baja los ficheros *Funcion.java* y *MetodosFunciones.java* de los recursos de la asignatura en el Aula Virtual e inclúyelos en el paquete auxiliar. Explora el código que contienen y observa su comportamiento con los apartados que siguen:

2. Crea una clase principal (main) *Ejercicio1* en el paquete *cn1vPractica2*. Crea una clase *F1* dentro de *Ejercicio1* que implemente la interface *Funcion* (utiliza el correspondiente *import*) para definir la función

$$f(x) = \begin{cases} 1 + \cos(x^2 e^x) & \text{si } x \geq 0 \\ e^{-0.1x} + \cos(x^2 e^{-x}) & \text{si } x < 0 \end{cases}.$$

Usa el método *tablaGrafica* de la clase *MetodosFunciones.java* para calcular las coordenadas de 11 puntos de la gráfica de la función anterior con abscisas equidistribuidas entre -5 y +5.

Imprime las coordenadas de los puntos anteriores, y como esta tarea la podemos repetir en otras situaciones, implementa métodos *estático toString(double[] lista)* y *toString(double[][] array)* en una nueva clase *MetodosListas* del paquete auxiliar, que devuelvan cadenas *String* con los datos de la lista o del array doble)

3. Crea una clase *F2* dentro de *Ejercicio1* que implemente la función

$$g(x) = \pi + \sqrt{1 + x^2}.$$

Utiliza esta clase y los métodos de la clase *MetodosFunciones.java* para imprimir las coordenadas de 11 puntos de la gráfica de la función *f + g* con abscisas equidistribuidas entre -5 y +5.

4. Implementa en *MetodosFunciones.java* las siguientes clases

- (a) una clase *diferencia* que defina la función diferencia de dos funciones.
- (b) una clase *producto* que defina la función producto de dos funciones.
- (c) una clase *composicion* que defina la función compuesta de dos funciones.

Comprueba su funcionamiento con las funciones *f* y *g* de los apartados anteriores.

5. Crea una clase *F3* también en el *Ejercicio1* que permita implementar funciones  $f_n(x) = x^n e^x$  pasando como argumento el exponente *n* al constructor de los objetos de la clase.

Comprueba su funcionamiento construyendo una tabla con los valores de las funciones  $x, x^2$  y  $x^3$  en 11 abscisas equidistribuidas entre -5 y +5.

## Ejercicio 2 (Dibujando Gráficas)

Descarga el fichero `PanelDibujo.java` de la zona de recursos del Aula Virtual, e inclúyelo en el paquete `auxiliar`.

Descarga el fichero `TestGraficas.java` y pónlo en el paquete `cnlvPractica2`.

Compila los ficheros anteriores y ejecuta el programa `TestGraficas`, comprobando que abre una ventana y va dibujando en ella algunas gráficas.

Procura comprender el funcionamiento de los distintos métodos y objetos que intervienen en el programa leyendo los comentarios incluidos en los correspondientes ficheros.

**Ejercicio 3** Baja del Aula Virtual de la UMU el fichero `ParaPanelDibujo.txt` e incluye las variables y métodos que contiene en la clase `PanelDibujo.java`. Con estos métodos podemos incluir puntos y etiquetas en los paneles de dibujo.

Crea aplicaciones que dibujen en distintos paneles las gráficas de las funciones del Ejercicio 1 para  $x \in [-5, 5]$ :

1. representa la función  $f$  del apartado 2 junto con 11 puntos de las mismas equidistribuidos en ese intervalo.
2. representa simultáneamente las funciones  $f$ ,  $g$  y  $f + g$  utilizando distintos colores y añadiendo etiquetas en esos colores con los nombres de cada función.
3. representa simultáneamente las funciones  $f_n$  del apartado 5 para los exponentes  $n = 1, 2, \dots, 10$  utilizando distintos colores

**Ejercicio 4 (Uso de números complejos)** El propósito de esta práctica es incorporar a nuestros proyectos una clase que nos permita trabajar con números complejos

1. Baja del Aula Virtual, el fichero comprimido `JavaComplex.zip`. Descomprímelo en una carpeta en el escritorio de tu ordenador y copia la carpeta `ORG` obtenida dentro del directorio de ficheros fuente “scr” de tu proyecto.
2. Abre la página `ORG/netlib/math/complex/Complex.html` donde encontrarás la descripción de los distintos métodos que incorpora.

La clase `Complex` incorporada proviene de un “repositorio” de paquetes de distintos lenguajes de programación en los que se pueden encontrar paquetes que podremos usar siguiendo las condiciones de la licencia establecida. Esta licencia aparece descrita en la página que hemos abierto. Debemos ser respetuosos con el autor y su trabajo, respetando íntegramente las condiciones de uso que plantee.

3. Comprueba el uso de los números complejos instados como objetos de la clase `Complex` realizando las siguientes operaciones dentro del programa `Ejercicio2`:

(a)  $z = (1 + 3i) + (2 - 6i)$

(b)  $x = \arg(1 + 3i)$ ,  $y = |1 + 3i|$

(c)  $x = \arg(-2 - 3i)$ ,  $y = |-2 - 3i|$

(d)  $z = \sqrt{1 - i}$ ,  $w = \sqrt{1 + i}$ ,  $z * w$

(e) determina las cinco raíces quintas de  $-1 - 2i$ .