

## Trabajo práctico

### Representación de números reales en formato punto flotante

#### Ejercicio 1

Analice y compile el archivo ex\_01.c.

1. Inspeccione el código y determine el objetivo del programa.
2. Analice los valores de las variables f1, f2, d1, acum\_1 y acum\_2.
3. ¿Qué conclusión puede obtener a partir de estos valores?

#### Ejercicio 2. Modos de redondeo

Analice y compile el archivo ex\_02.c.

1. ¿Para qué sirven las funciones fegetround() y fesetround()?
2. ¿Cuál es el modo de redondeo por defecto con el que arranca el programa?
3. Ejecutar el programa para los modos de redondeo FE\_DOWNWARD, FE\_UPWARD y FE\_TOWARDZERO, y compárelos con el modo FE\_TONEAREST.
4. ¿Observa diferencias? ¿Estas diferencias son consistentes con los modos de redondeos?

#### Ejercicio 3. Valores especiales

Investigue cómo inicializar en C variables en punto flotante con los valores especiales NaN, Inf e -Inf. Además, escriba un programa en C que produzca como resultado estos valores especiales.

#### Ejercicio 4. Excepciones

Analice y compile el archivo ex\_04.c.

1. Analice los resultados impresos por consola ¿Son los resultados consistentes con las operaciones ejecutadas?
2. Explique qué hacen las funciones feclearexcept(), feraiseexcept() y fetestexcept().

#### Ejercicio 5. Excepciones

Con el código visto en el ejercicio 4, ex\_04.c. genere los 4 tipo de excepciones estudiadas al ejecutar diferentes operaciones matemáticas, en lugar de utilizar la función feraiseexcept().

#### Ejercicio 6. Manejo de excepciones

Analice y compile el archivo ex\_05.c.

3. Analice los resultados impresos por consola ¿Son los resultados consistentes con las operaciones ejecutadas?
4. Descomentar las líneas 44 y 45 y volver a compilar.
5. ¿Qué observa por consola? ¿Cuál es la función de feenableexcept()?
6. Descomentar la línea 47 y volver a compilar.
7. ¿Qué observa por consola? ¿Cuál es el objetivo de la función signal(SIGFPE, fpe\_handler)?