

Práctica 2. Iteración y recursión II (recursión)

En esta práctica resolveremos el mismo problema que en la anterior pero **recursivamente** en vez de iterativamente.

Como en la práctica anterior, supondremos que se quiere hallar una de las raíces de la ecuación $f(x) = 0$ donde f es una función continua de la que se sabe que en los extremos del intervalo $[a, b]$ tiene signos distintos, es decir, que el signo de $f(a)$ es distinto que el de $f(b)$.

El algoritmo es esencialmente el mismo, pero la forma de implementarlo es diferente: **una función debe llamarse a sí misma cambiando alguno de sus parámetros** hasta encontrar la solución.

EJERCICIO:

1. Construya una función para resolver una ecuación mediante el método de bisecciones sucesivas **recursivamente**, con tres parámetros de punto flotante de simple precisión:
 - `$f12`: extremo del intervalo donde la función es negativa.
 - `$f13`: extremo del intervalo donde la función es positiva.
 - `$f14`: tolerancia del error en la solución.

La función devolverá en `$f0` la raíz de la ecuación $f(x) = 0$ donde $f(x)$ es una función definida mediante la etiqueta `Funcion` cuya variable independiente, x , vendrá dada en `$f16` y retornará el valor $y = f(x)$ en el registro `$f0`. La solución devuelta será el punto medio del intervalo de ancho inferior al contenido del parámetro `$f14`.

2. Escriba un programa que, empleando las funciones del ejercicio anterior, pida por teclado los tres parámetros de la función e imprima en la consola la solución de la ecuación $f(x) = 0$. En caso de que en los extremos del intervalo suministrado la función tenga el mismo signo deberá mostrarse un mensaje de error.

EVALUACIÓN Y ENTREGA DE LA PRÁCTICA

La práctica será evaluada con la herramienta Tablon, disponible en <http://tablon-aoc.infor.uva.es/>.

Para poder evaluar la práctica, el fichero `.asm` enviado al Tablon debe cumplir los siguientes requisitos:

- Debe contener únicamente la función implementada por el grupo. Es decir, no debe contener ni el código correspondiente al `main` ni el procedimiento `Funcion` implementado.
- La función debe llamarse `Bisec`.
- $f(x)$ debe invocarse mediante una **llamada a un procedimiento** denominado `Funcion`.

Una vez enviado el código, Tablon realizará una serie de pruebas, testeando diferentes valores de entrada para `Bisec`. Una vez finalizada la batería de pruebas, se podrá consultar el Tablon el número de pruebas superadas y el porcentaje de pruebas completado, además de otra información relevante, como el número de instrucciones ejecutadas, el número de líneas de código de la función `Bisec` y el número de excepciones en tiempo de ejecución. Estas métricas sirven para elaborar el ranking y desempatar en caso de empate en el número de casos de prueba superados. Sin embargo, **la parte de la calificación de la práctica dependiente del Tablon dependerá exclusivamente del número de pruebas superadas**. El resto de métricas y el ranking no se tendrá en cuenta para la evaluación.

Finalmente, se debe entregar en el Aula Virtual el fichero `.asm` que obtuviera el mejor resultado en el ranking de la herramienta Tablon. El nombre del fichero para el Aula Virtual será el número de *request* del Tablon con 5 cifras y la extensión `.asm`.

NOTA: La calificación no depende exclusivamente del resultado del Tablon. El código también se revisará manualmente para evaluar su calidad y proporcionar *feedback* a los alumnos en las siguientes prácticas.

Fecha límite de entrega en el Aula Virtual: 10 de octubre a las 23:55 h.

No se evaluarán las entregas del Tablon que no se hayan depositado dentro del plazo en el Aula Virtual.