

Práctica 4 – Obtención de raíces por aproximación

Ejercicio 1

Calcular la raíz de las funciones de trabajo $f_1(x)=x^2-e^x$ y $f_2(x)=2x^3-e^x$ mediante el método de Newton-Raphson. Para ello se pide:

1. Dibujar con GNUPLOT estas dos funciones de trabajo y seleccionar un punto de arranque próximo a la raíz buscada en cada una de ellas.
2. Construir un programa (principal) Fortran en un fichero que:
 - A) Pida por pantalla: a) el punto de arranque próximo a la raíz para cada función, b) la precisión de cero (en y) deseada, y c) el número máximo de iteraciones que se permite al método.
 - B) Llame a una subrutina para que aplique el método a $f_1(x)$.
 - C) Llame a la misma subrutina para que aplique el método a $f_2(x)$.
 - D) Proporcione por pantalla el valor de las raíces (números reales) y el número de iteraciones (números enteros) usadas por el método para cada función de trabajo.
 - E) Informe si se ha superado el número máximo de iteraciones (el método no converge), dando el resultado para la última iteración realizada. Además, indique que se debe: a) acercar el punto de arranque a la raíz y/o, b) reducir el error de cero, y/o, d) aumentar el número máximo de iteraciones.
3. Construir un programa (método) Fortran que codifique mediante una subrutina el método de Newton-Raphson. Esta subrutina debe tener como argumento de entrada una función vectorial ("*function*") que calcule el valor de la función de trabajo en el punto x que se desee y su derivada primera.
4. Construir un fichero (trabajo) Fortran que codifique mediante dos funciones vectoriales ("*function*") cada función de trabajo y su derivada primera.

Ejercicio 2

Repetir el ejercicio 1. Ahora aplicando el método de la secante.

Nota 1: se deben seleccionar dos puntos de arranque para cada función de trabajo.

Nota 2: utilizar el mismo fichero de trabajo que en el ejercicio 1.

Ejercicio 3

Comparar conjuntamente los métodos de obtención de raíces realizados en esta práctica número 4 (Newton-Raphson y secante) y en la práctica número 3 (bisección y falsa posición). Para ello se pide:

- A) Construir un programa (principal) Fortran que: a) use a un módulo en el que se encuentran codificados los 4 métodos (dos funciones vectoriales y dos subrutinas), b) pida por pantalla los valores de entrada que necesita cada método, c) aplique cada método a cada una de las dos funciones de trabajo, d) proporcione por pantalla las raíces obtenidas por cada método para cada función de trabajo, así

como el número de iteraciones empleadas, y e) informe si se ha superado el número máximo de iteraciones.

- B) Construir un programa (módulo) Fortran que construya un módulo el cual codifica mediante dos funciones vectoriales el método de la bisección y el método de la falsa posición, y mediante dos subrutinas el método de Newton-Raphson y el método de la secante.
- C) Construir un programa (trabajo) Fortran que codifique mediante funciones escalares cada función de trabajo (las usadas en la práctica 3) y mediante funciones vectoriales cada función de trabajo y su derivada primera (las usadas en esta práctica 4).

Nota: La compilación debe hacerse con el programa módulo en primer lugar, y el resto en cualquier orden:

gfortran modulo.f90 trabajo.f90 principal.f90 -o salida.exe

Ejercicio 4:

Calcular la raíz de la función de trabajo $f_1(x)=x^2-e^x$ mediante el método de Newton-Raphson simultáneamente para varios puntos de arranque. Para ello se pide:

1. Dibujar con GNUPLOT la función de trabajo y seleccionar varios puntos de arranque próximos a la raíz buscada.
2. Construir un programa (principal) Fortran en un fichero que:
 - A) Dimensione dinámicamente un vector que contenga los valores de arranque próximos a la raíz, pida que se le dé por teclado la dimensión de dicho vector, se le asigne esta dimensión, y se construya entrando los valores por teclado.
 - B) Pida por pantalla el número de iteraciones que se aplica al método (el mismo para todos los puntos de arranque).
 - C) Llame (una sola vez) a una subrutina para que aplique el método a la función de trabajo con cada punto de arranque.
 - D) Proporcione por pantalla el valor de la raíz para cada punto de arranque y el error de cero obtenido por el método para cada uno de estos puntos de arranque.
3. Construir un fichero (método) Fortran que codifique mediante una subrutina el método de Newton-Raphson. Esta subrutina debe tener como argumento de entrada un vector construido de forma supuesta con los puntos de arranque. Además, la subrutina debe tener como argumento una función vectorial ("*function*") que calcule el valor de la función de trabajo en cada punto x y su derivada primera.
4. Construir un fichero (trabajo) Fortran que codifique mediante una función vectorial ("*function*") la función de trabajo y su derivada primera.