

Programación: Método del punto fijo (método de iteración simple)

Objetivos. Programar el método del punto fijo.

Requisitos. Entender bien la idea del ciclo `while`. Tener preparados varios ejemplos “buenos” y “malos” que muestren la convergencia y divergencia del método. Se recomienda usar las funciones g_1, g_2, g_3, g_4, g_5 de la clase anterior (“Programación: ejemplos para el método del punto fijo”).

1. Idea del método. Buscamos el punto fijo de una función dada g . A partir de un punto dado x_0 construimos x_1, x_2, \dots mediante la siguiente fórmula recursiva:

$$x_p = g(x_{p-1}).$$

Se garantiza la convergencia de este proceso, si la función g es contractiva.

2. Guardar solamente el punto actual y el anterior. En el programa no es necesario guardar todos los puntos x_0, x_1, x_2, \dots . En cada momento vamos a guardar solamente el punto actual x_p denotándolo por x y el punto anterior x_{p-1} que denotamos por x_{prev} .

3. Condición de terminación y condición de continuación. Denotemos por d a la distancia entre el punto actual x y el punto anterior x_{prev} . El método de punto fijo debe pararse cuando d es pequeño (menor que un número dado $xtol$) o cuando el número de los pasos hechos es mayor o igual al número de pasos máximo permitido. En otras palabras, la condición de terminación es la siguiente:

$$d < xtol \quad \vee \quad s \geq smax. \quad (1)$$

La condición de continuación será la negación de la condición de paro (1). Recuerde la ley de De Morgan sobre la negación de la disyunción:

$$\overline{a \vee b} =$$

Ahora escribamos la condición de continuación en el método de punto fijo. Hay que continuar, mientras se cumpla la siguiente condición:

$$\underbrace{\hspace{2cm}} \quad \underbrace{?}_{\text{?}} \quad \underbrace{\hspace{2cm}}$$

4. Algoritmo FixPoint.

Escriba una función con argumentos g , x_0 , $xtol$, $smax$ que aplique el método de iteración de punto fijo a la función g con el punto inicial x_0 . La función debe regresar la última aproximación al punto fijo y el número de las iteraciones hechas. Se recomienda seguir el siguiente esquema (sustituir los signos ??? por expresiones correctas).

```
función FixPoint(g, x0, xtol, smax),
  variables locales: ???;
  s := ???;
  d := 2 * xtol;
  x := ???;
  mientras ???:
    xprev := x;
    x := ???;
    d := abs(x - xprev);
    s := s + 1;
  devolver x, s.
```

5. Pruebas. Pruebe la función FixedPoint con varios ejemplos “buenos” y “malos”; se recomienda usar las funciones g_1, g_2, g_3, g_4, g_5 de la clase anterior. Ponga $smax = 30$, $xtol = 10^{-8}$. Como la aproximación inicial x_0 puede usar el extremo izquierdo a_k (escriba este número con el punto decimal).

- $g_1(x) = \sqrt[3]{x+1}$, $[a_1, b_1] = [1, 2]$, $x_0 = 1.0$.
- $g_2(x) = \log(2-x)$, $[a_2, b_2] = [0, 0.8]$, $x_0 = 0.0$.
- $g_3(x) = x - \frac{x^3 - x - 1}{3x^2 - 1}$, $[a_3, b_3] = [1.2, 2]$, $x_0 = 1.2$.
- $g_4(x) = x^3 - 1$, $[a_4, b_4] = [1, 2]$, $x_0 = 1.0$.
- $g_5(x) = 2 - e^x$, $[a_5, b_5] = [0, 0.8]$, $x_0 = 0.0$.

Por ejemplo, en Wolfram Mathematica puede escribir el programa de la siguiente manera:

```
FixPoint[g_, x0_, xtol_, smax_] := ...
g1[x_] := (x + 1) ^ (1/3);
FixPoint[g1, 1.0, 10^(-8), 30]
```

6. Problema adicional FixedPointRecur. Escriba una función recursiva con argumentos g , x_0 , $xtol$, $smax$ que realice el método de iteración de punto fijo. La función tiene que regresar el punto fijo y el número de las iteraciones hechas.