Práctica 2. Resolución numérica de problemas de contorno no lineales 1D .

1. El objetivo de esta epígrafe es realizar un programa en python que aproxime la solución del siguiente problema de contorno no lineal:

$$\begin{aligned} -\nu u''(x) - u^2(x) &= f(x),\ x \in (a,b) \\ u(a) &= u_a \\ u(b) &= u_b, \end{aligned}$$

   siendo $a = 0,\ b = \pi,\ \nu = 1.0,\ f(x) = cos(x)\,(1 - cos(x)),\ u_a = 1,\ u_b = -1.$ Para este problema se sabe que la solución exacta es $u(x) = cos(x)$.

   Para aproximar la solución del problema anterior vamos a usar el siguiente algoritmo iterativo:

$$AU^{l+1} = F + (U^2)^l,\ l = 0, 1, \ldots$$

   donde A es la matriz que resulta de aplicar el mtodo de diferencias finitas al operador $-u''(x)$, es decir

$$A = \frac{1}{\Delta x^2} M$$

   siendo $M(i,i) = 2,\ M(i-1,i) = -1,\ M(i,i+1) = -1$; U^{l+1} es el vector de incógnitas (suponemos que u_0^{l+1} y u_N^{l+1} son valores desconocidos)

$$U^{l+1} = \left(u_0^{l+1}, \cdots, u_N^{l+1}\right)^T,$$

   F es el vector que resulta de evaluar $f(x)$ en los nodos del mallado

$$F = (f(x_0), \cdots, f(x_N))^T$$

   y $(U^2)^l$ es el vector que resulta de elevar al cuadrado cada una de las componentes de la aproximación correspondiente a la l-ésima iteración

$$(U^2)^l = \left((u_0^2)^l, \cdots, (u_N^2)^l\right).$$

   El algoritmo iterativo finalizará cuando el error $\|U^{l+1} - U^l\|_\infty < \epsilon$ ($\epsilon \approx 10^{-7}$) o cuando se alcance un número máximo de iteraciones (digamos que 500).

   Realizar un programa en python que aproxime el problema anterior con el algoritmo iterativo descrito y realizar varias ejecuciones para diferentes particiones del itervalo $[0, \pi]$, digamos 50, 100, 200 y 400.

   Observación: en la descripción del algoritmo anterior no se han tenido en cuenta la imposición de las condiciones de contorno. Se recomienda utilizar la técnica de penalización vista en la práctica anterior. Para inicializar el algoritmo iterativo anterior tomaremos $U^0 = 0$.

2. Consideremos ahora el siguiente algoritmo iterativo para aproximar el problema anterior:

$$A(U^l)U^{l+1} = F,\ l = 0, 1, \ldots$$

donde $A(U^l)$ es la matriz

$$A(U^l) = \frac{1}{\Delta x^2} M - D(U^l)$$

siendo $D(U^l)$ la matriz diagonal definida por $D(U^l)(i,i) = u_i^l$.

Hacer un programa en python con la implementación de este algoritmo iterativo para el problema anterior y comparar los resultados obtenidos con los que has obtenido en el epígrafe anterior.

3. Usando lo aprendido en los epígrafes anteriores diseñar un algoritmo en python para aproximar la solución del siguiente problema de evolución usando el método de Euler implícito para la discretización en tiempo.

$$\begin{aligned} \frac{\partial u}{\partial t} - \nu \frac{\partial^2 u}{\partial x^2} &= f(x) + u^2, \; x \in (a,b) \\ u(t,a) &= u_a \\ u(t,b) &= u_b, \\ u(0,x) &= u_0(x) \end{aligned}$$

siendo $a = 0$, $b = \pi$, $\nu = 1.0$, $f(x) = cos(x)\,(1 - cos(x))$, $u_a = 1$, $u_b = -1$, $u_0(x) = 1 - \frac{2x}{\pi}$.