Guía de usuario (o eso me gustaría)

Los siguientes programas, escritos en el lenguaje de programación Python , describen métodos numéricos utilizados en búsquedas, ecuaciones lineales e interpolación.

He intentado (fallidamente) implementar estos en un sitio web bajo el framework Django, haciendo uso de formularios con el fin de mejorar la accesibilidad a los mismos.

Los diferentes códigos pueden fallar en dado caso de no utilizar los tipos de variable indicados.

**Métodos de búsquedas:**

Bisección:

El método de la bisección se caracteriza por ser un método cerrado o por intervalos porque para su ejecución se requiere que en el intervalo la función cambie de signo en los extremos y que sea continua, es decir, que se garantice la existencia de un intervalo que contenga al menos una raíz.

Éste método recibirtá una función en formato de string, los límites inferior y superior de un intervalo en formato de float, una tolerancia en formato de float y una cantidad de iteraciones en formato de int por parte del usuario.

Regla falsa:

El método de la regla falsa parte de un intervalo que cumple las condiciones de cambio de signo y continuidad. Además, se utiliza para encontrar una aproximación a una raíz de una ecuación determinada, mediante la estrategia de reducir dicho intervalo en cada iteración. Para ello utiliza un punto medio en cada etapa, que a su vez es la aproximación a la raíz.   
La peculiaridad de éste, es que combina dos métodos: el método de bisección y el de la secante.

Éste método recibirá como parámetros una función como string, un x0 y un x1 como float, una tolerancia como float y una cantidad de iteraciones como int por parte del usuario.

Punto fijo:

El método del punto fijo re-formula la ecuación f(x)=0 y genera una ecuación de la forma x=g(x) que permite encontrar un valor de x que al reemplazarlo en g su resultado sea el mismo, es decir que x sea invariable para g, y adicionalmente que la f(x) converja a cero.

Éste método recibirá como parámetros una función como string, un valor de aproximación inicial como float, una tolerancia deseada como float y una cantidad de iteraciones como int por parte del usuario.

Búsquedas incrementales:

Si se tiene un intervalo cerrado en el cual existe un cambio de signo en el valor de la función, f(x1)\*f(x2)<0 ,es posible tomar valores pequeños y consecutivos dentro de este intervalo para verificar entre cuáles de ellos existe el cambio de signo. Al identificarlo, tendremos un nuevo intervalo pequeño que contiene a la raíz.

En este método recibiremos una función como string, los límites inferior y superior de un intervalo como float, un delta como float dado por los límites del intervalo y una cantidad de iteraciones como int.

Raíces múltiples:

El método de raíces múltiples facilita la solución de problemas en donde una misma raíz Xn puede repetirse en la ecuación, ya que para los métodos de Newton y secante puede darse el caso de una convergencia lineal si hay presencia de raíces múltiples por lo que el método fallaría y no se resolvería la ecuación, es ahí donde es importante implementar un proceso diferente.

Este método recibirá como parámetros un primer punto p1 dado como float, una tolerancia dada como float y una cantidad máxima de iteraciones dada como int.

Newton:

El método de Newton es uno de los más rápidos, este puede considerarse como un tipo de método de punto fijo, Si se observara el método de Newton gráficamente podría observarse que la recta tangente que pasa por el punto (Xn, f(Xn)) interceptará al eje X en el punto Xn+1, de esta manera es como el método converge a la raíz de la ecuación, mediante rectas tangentes que terminarán por pasar por la raíz de la función.

Este método recibirá como parámetros una función y su derivada como dos strings, un valor inicial como float, una tolerancia como float y una cantidad máxima de iteraciones como int.

Secante:

Como método de la secante no siempre converge, es recomendable buscar un par de buenos valores iniciales, y en lo posible, que definan un intervalo que contenga una raíz.  Es un método abierto porque parte de dos valores iniciales. Es decir, el método calcula valores mediante los dos últimos y desecha siempre el más viejo de todos.

Este método recibirá como entradas una función como string, un x0 como float, un x1 como float, una tolerancia como float y una cantidad máxima de iteraciones como int.

**Ecuaciones lineales:**

Gauss Simple:

En este método, primero realizaremos operaciones elementales de filas hasta llegar a una matriz triangular inferior, luego realizaremos despejes para encontrar los valores de nuestras variables.  
Recibirá como parámetros un int n que dará las dimensiones de la matriz nxn, los valores de las posiciones de la matriz como float y los valores del vector respuesta como float.

Gauss pivoteo parcial:

Muy similar a Gauss Simple, pero se hará un intercambio de fila con el fin de reducir el error de propagación.

Recibirá como parámetros un int n que dará las dimensiones de la matriz nxn, los valores de las posiciones de la matriz como float y los valores del vector respuesta como float.

Gauss LU:

Éste método factoriza la matriz A en el producto de dos matrices (L y U), para luego mediante sustitución progresiva y regresiva hallar la solución del sistema.

Recibirá como parámetros un int n que dará las dimensiones de la matriz nxn, los valores de las posiciones de la matriz como float y los valores del vector respuesta como float.

Dolittle:

El método de Doolittle es una variación de Crout que obtiene las matrices de factorización LU fila a fila o columna a columna. Resulta útil para matrices de grandes dimensiones de las cuales solo se guardan fila a fila o columna a columna los elementos distintos de cero.

Este método recibirá como parámetros un int n que dará las dimensiones de la matriz nxn, los valores de las posiciones de la matriz como float y los valores del vector respuesta como float.

Gauss Seidel:

El método de Gauss Seidel es basicamente igual al método de Jacobi, la principal diferencia es que cada valor calculado es Xk es usado para recalcular el valor de Xk+1 por ende converge más rapido a la solución que el método de Jacobi.

Este método recibirá como parámetros un int n que dará las dimensiones de la matriz nxn, los valores de las posiciones de la matriz como float y los valores del vector respuesta como float.

**Interpolación:**

Vandermonde:

Una matriz de Vandermonde es aquella que presenta una progresión geométrica en cada fila. En el primer elemento de cada fila hay solamente unos (al ser la potencia de cero) y en el segundo elemento hay una serie de números arbitrarios. En el tercero se encuentran esos mismos números elevados al cuadrado. En el cuarto están esos mismos números elevados al cubo y en las siguientes columnas elevados a la potencia inmediatamente superior de manera que en el elemento n de cada fila esos números estén elevados a la potencia n-1.