Algorithm 1 INCREMENTAL SEARCH

```
Input: $f, x_0, h, n_{max}$
Output: $a, b, iter$
  $x_{ant} \leftarrow x_o$
  $f_{ant} \leftarrow f(x_{ant})$
  $x_{act} \leftarrow x_{ant} + h$
  $f_{act} \leftarrow f(a_{ct})$
  For $i \leftarrow 1, n_{max}$ do
    if $f_{ant} * f_{act} < 0$ Then
      break
    End if
    $x_{ant} \leftarrow x_{act}$
    $f_{ant} \leftarrow f_{act}$
    $x_{act} \leftarrow x_{ant} + h$
    $f_{act} \leftarrow f(x_{act})$
  End For
  $a \leftarrow x_{ant}$
  $b \leftarrow x_{act}$
  $iter \leftarrow i$
  Return $a, b, iter$
```

Algorithm 2 BISECTION

```
Input: $f, a, b, tol, n_{max}$
Output: $x, iter, err$
$fa \leftarrow f(a)$
$f_{pm} \leftarrow (a+b)/2$
$f_{pm} \leftarrow f(pm)$
$E \leftarrow 1000$
$cont \leftarrow 1$
While $(E > tol)$ & $(cont < N_{max})$ do
    if $(f_a * f_{pm} < 0)$ Then
        $b \leftarrow pm$
    else if
        $a \leftarrow pm$
    End if
    $p_0 \leftarrow pm$
    $pm \leftarrow (a+b)/2$
    $f_{pm} \leftarrow f(pm)$
    $E \leftarrow |pm - p_0|$
    $cont \leftarrow cont + 1$
End While
$x \leftarrow pm$
$iter \leftarrow cont$
$err \leftarrow E$
Return $x, iter, err$
```

Algorithm 3 FALSE RULE

```
**Input:** $f, a, b, tol, N_{max}$
**Output:** $x, iter, err$
$f_a \leftarrow f(a)$
$f_b \leftarrow f(b)$
$f_{pm} \leftarrow f(pm)$
$E \leftarrow 1000$
$cont \leftarrow 1$
**While** $(E > tol)$ & $(cont < N_{max})$ **do**
    **if** $(f_a * f_{pm} < 0)$ **Then**
        $b \leftarrow pm$
    **else if**
        $a \leftarrow pm$
    **End if**
    $p_0 \leftarrow pm$
    $pm \leftarrow (f(b) * a - f(a) * b)/f(b) - f(a))$
    $f_{pm} \leftarrow f(pm)$
    $E \leftarrow |pm - p_0|$
    $cont \leftarrow cont + 1$
**End While**
$x \leftarrow pm$
$iter \leftarrow cont$
$err \leftarrow E$
**Return** $x, iter, err$
```

z

Algorithm 4 FIXED POINT

```
**Input:** $g, x_0, tol, N_{max}$
**Output:** $x, iter, err$
$x_{ant} \leftarrow x0$
$E \leftarrow 1000$
$cont \leftarrow 1$
**While** $(E > tol)$ & $(cont < N_{max})$ **do**
    $x_{act} \leftarrow g(x_{ant})$
    $E \leftarrow |x_{act} - x_{ant}|$
    $cont \leftarrow cont + 1$
    $x_{ant} \leftarrow x_{act}$
**End While**
$x \leftarrow x_{act}$
$iter \leftarrow cont$
$err \leftarrow E$
**Return** $x, iter, err$
```

Algorithm 5 NEWTON

```
Input: $f, df, x_0, tol, N_{max}$
Output: $x, iter, err$
$x_{ant} \leftarrow x0$
$f_{ant} \leftarrow f(x_{ant})$
$E \leftarrow 1000$
$cont \leftarrow 0$
While $(E > tol)$ & $(cont < N_{max})$ do
    $x_{act} \leftarrow (x_{ant} - f_{ant})/(df(x_{ant}))$
    $f_{act} \leftarrow f(x_{act})$
    $E \leftarrow |x_{act} - x_{ant}|$
    $cont \leftarrow cont + 1$
    $x_{ant} \leftarrow x_{act}$
    $f_{ant} \leftarrow f_{act}$
End While
$x \leftarrow x_{act}$
$iter \leftarrow cont$
$err \leftarrow E$
Return $x, iter, err$
```

Algorithm 6 SEC

```
Input: $f, x_0, x_1, tol, N_{max}$
Output: $x, iter, err$
$f_0 \leftarrow f(x_0)$
$f_1 \leftarrow f(x_1)$
$E \leftarrow 1000$
$cont \leftarrow 1$
While $(E > tol)$ & $(cont < N_{max})$ do
    $x_{act} \leftarrow x1 - f1 * (x_1 - x_0)/(f_1 - f_0)$
    $f_{act} \leftarrow f(x_{act})$
    $E \leftarrow |x_{act} - x_1|$
    $cont \leftarrow cont + 1$
    $x_0 \leftarrow x_1$
    $f_0 \leftarrow f_1$
    $x_1 \leftarrow x_{act}$
    $f_1 \leftarrow f_{act}$
End While
$x \leftarrow x_{act}$
$iter \leftarrow cont$
$err \leftarrow E$
Return $x, x_{act}, err$
```

Algorithm 7 MULTIPLE ROOTS

Input: $f, f', f'', x0, tol, N_{max}$

Output: $x, iter, err$

```
$x_{ant} \leftarrow x0$
$f_{ant} \leftarrow f(act)$
$E \leftarrow 1000$
$cont \leftarrow 0$
**While** $(E > tol)$ & $(cont < N_{max})$ **do**
    $x_{act} \leftarrow (x_{ant} - f_{ant}) * f''(x_{ant})/(f'(x_{ant})^2 - f_{ant} * f''(x_{ant}))$
    $fa_{ct} \leftarrow f(x_{act})$
    $E \leftarrow |x_{act} - x_{ant}|$
    $cont \leftarrow cont + 1$
    $x_{ant} \leftarrow x_{act}$
    $f_{ant} \leftarrow f_{act}$
**End While**
$x \leftarrow x_{act}$
$iter \leftarrow cont$
$err \leftarrow E$
**Return** $x, iter, err$
```

Algorithm 8 GAUSSIAN ELIMINATION

Input: A, b

Output: x

```
$n \leftarrow size(A, 1)$
$M \leftarrow [Ab]$
**For** $i \leftarrow 0, n - 1$ **do**
    **For** $j \leftarrow i + 1, n$ **do**
        **if** $M(j, i) \neq 0$ **Then**
            $M(j, i : n + 1) \leftarrow M(j, i : n + 1) - (M(j, i)/M(i, i)) * M(i, i : n + 1)$
        **End if**
    **End For**
**End For**
$x \leftarrow backsubstitution(M)$
**Return** $x$
```

Algorithm 9 GAUSSIAN ELIMINATION WITH PARCIAL PIVOTING

```
Input: $A, b$
Output: $x$
  $n \leftarrow size(A,1)$
  $M \leftarrow [Ab]$
  For $i \leftarrow 1, n-1$ do
    $[aux0, aux] \leftarrow max|M(i+1:n,1)|$
    if $aux_0 > |M(i,i)|$ Then
      $aux_2 \leftarrow M(i+aux, i:n+1)$
      $M(aux+i, i:n+1) \leftarrow M(i, i:n+1)$
      $M(i, i:n+1) \leftarrow aux_2$
    End if
    For $j \leftarrow i+1, n$ do
      if $M(j,i) \neq 0$ Then
        $M(j, i:n+1) \leftarrow M(j, i:n+1) - (M(j,i)/M(i,i)) * M(i, i:n+1)$
      End if
    End For
  End For
  $x \leftarrow backsubstitution(M)$
  Return $x$
```

Algorithm 10 GAUSSIAN ELIMINATION WITH TOTAL PIVOTING

```
Input: $A, b$
Output: $x$
  $n \leftarrow size(A,1)$
  $M \leftarrow [Ab]$
  $aux \leftarrow []$
  For $i \leftarrow 1, n-1$ do
    $[a, b] \leftarrow find(|M(i:n, i:n)|) \equiv max(|M(i:n, i:n)|))$ Cambio de columna
    if $b(1)+i-1 \neq i$ Then
      $aux \leftarrow += [ib(1)+i+1]$
      $aux_2 \leftarrow M(:, b(1)+i+1)$
      $M(:, b(1)+i-1) \leftarrow$
      $M(:, i) \leftarrow$
    End ifCambio de fila
    if $a(1)+i \neq i$ Then
      $aux_2 \leftarrow M(i+a(1)-1, i:n+1)$
      $M(a(1)+i-1, i:n+1) \leftarrow M(i, i:n+1)$
      $M(i, i:n+1) \leftarrow aux_2$
    End if
    For $j \leftarrow i+1, n$ do
      if $M(j,i) \neq 0$ Then
        $M(j, i:n+1) \leftarrow M(j, i:n+1) - (M(j,i)/M(i,i)) * M(i, i:n+1)$
      End if
    End For
  End For
  $x \leftarrow backsubstitution(M)$
  Return $x$
```