3.- CONTINUIDAD DE UNA FUNCION REAL DE VARIABLE REAL PROCEDIMIENTOS

1.- ESTUDIO DE LA CONTINUIDAD DE UNA FUNCIÓN EN UN PUNTO

Descripción del método

- 1. Definición de la función y representación gráfica de la misma
- 2. Definición de los posibles puntos de discontinuidad
- 3. Cálculo de la imagen y de los límites laterales en los posibles puntos de discontinuidad
- 4. Estudio de la continuidad

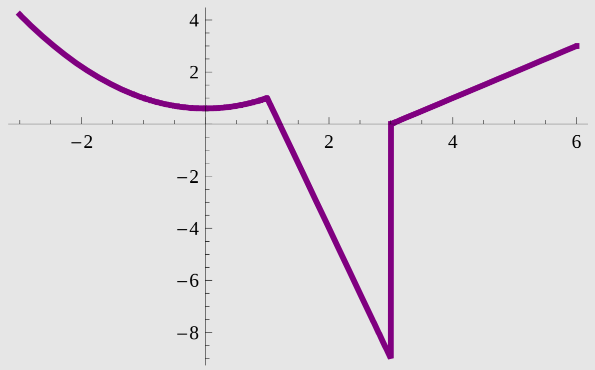
Procedimiento

Estudiar la continuidad de la función $f(x) = \begin{cases} \frac{2x^2+3}{5} & \text{si} \quad x \le 1 \\ 6-5x & \text{si} \quad 1 < x < 3 \\ x-3 & \text{si} \quad x \ge 3 \end{cases}$

⇒ 1. Definición de la función y representación gráfica de la misma

$$f[x\_] = \text{Which}\left[x \le 1,\ \frac{2x^2+3}{5},\ 1 < x < 3,\ 6-5x,\ x \ge 3,\ x-3\right];$$

```
Plot[f[x], {x, -3, 6}, PlotStyle → {Purple, Thickness[0.01]}]
```



Aunque el gráfico represente la función de forma contínua, puede que en el punto x=3 la función no lo sea ya que aparece una raya vertical que puede unir dos tramos de la función que en realidad están separados.

⇒ 2. Definición de los posibles puntos de discontinuidad

Estudiamos los posibles puntos de discontinuidad que son x = 1 y x = 3 ya que en estos puntos la función cambia de valor, aunque gráficamente se intuye que la función es continua en x = 1 y que no es continua en x = 3. Realizamos los pasos 3 y 4 primero parael punto x = 1 y después para el punto x = 3.

x = 1

⇒ 3. Cálculo de la imagen y de los límites laterales en los posibles puntos de discontinuidad

```
punto = 1; f[punto];
limiteizquierda = Limit[f[x], x → punto, Direction → 1];
limitederecha = Limit[f[x], x → punto, Direction → -1];
```

⇒ 4. Estudio de la continuidad

```
If[f[punto] ≠ Infinity &&
  f[punto] ≠ -Infinity && f[punto1] ≠ ComplexInfinity, If[
  limiteizquierda == limitederecha, If[limiteizquierda == f[punto],
   Print["la función es continua en x = ", punto],
   Print["la funcion tiene limite en x = ",
    punto, " pero no es continua en ese punto"]],
  Print["la funcion no tiene limite en el punto x = ",
   punto, " y por tanto no es continua en dicho punto"]],
 Print["la función no tiene imagen en el punto x = ",
  punto, " y por lo tanto no es continua en dicho punto"]]
```

```
la función es continua en x = 1
```

x = 3

⇒ 3. Cálculo de la imagen y de los límites laterales en los posibles puntos de discontinuidad

```
punto = 3; f[punto];
limiteizquierda = Limit[f[x], x → punto, Direction → 1];
limitederecha = Limit[f[x], x → punto, Direction → -1];
```

⇒ 4. Estudio de la continuidad

```
If[f[punto] ≠ Infinity &&
  f[punto] ≠ -Infinity && f[punto1] ≠ ComplexInfinity, If[
  limiteizquierda == limitederecha, If[limiteizquierda == f[punto],
   Print["la función es continua en x = ", punto],
   Print["la funcion tiene limite en x = ",
    punto, " pero no es continua en ese punto"]],
  Print["la funcion no tiene limite en el punto x = ",
   punto, " y por tanto no es continua en dicho punto"]],
 Print["la función no tiene imagen en el punto x = ",
  punto, " y por lo tanto no es continua en dicho punto"]]
```

```
la funcion no tiene limite en el punto x = 3 y por tanto no es continua en dicho punto
```

2.- ESTUDIO DE LA CONTINUIDAD DE UNA FUNCIÓN EN UN INTERVALO

▼ Descripción del método

- 1. Estudio de la continuidad en el intervalo abierto (a,b). Si el intervalo es cerrado realizar los pasos 2, 3 y 4.
- 2. Cálculo del límite lateral por la derecha de la función en el punto x=a
- 3. Cálculo del límite lateral por la izquierda de la función en el punto x=b
- 4. Estudio de la continuidad en el intervalo cerrado [a,b]

Procedimiento

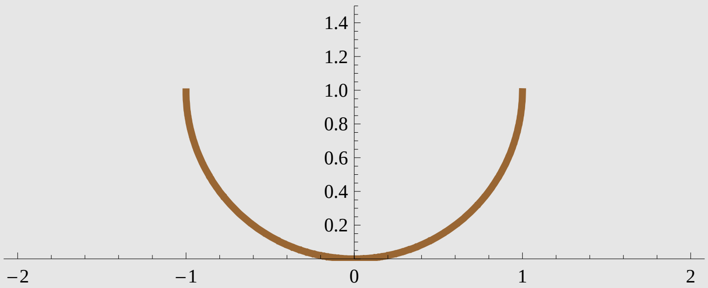
Estudiar la continuidad de la función $f(x) = 1 - \sqrt{1 - x^2}$ en el intervalo $[-1, 1]$

⇒ 1. Estudio de la continuidad en el intervalo abierto (a,b)

Se debe estudiar la continuidad en todos los puntos del intervalo (a,b), siguiendo los pasos vistos en el procedimiento anterior: *Estudio de la continuidad de una función en un punto*

```
f[x_] = 1 - √(1 - x^2) ; a = -1; b = 1;
```

```
Plot[f[x], {x, a - 1, b + 1}, PlotStyle → {Brown, Thickness[0.01]},
 AspectRatio -> Automatic, PlotRange → {0, 1.5}]
```



En el intervalo (-1,1) no hay ningún punto posible de discontinuidad, por tanto f(x) es continua en el intervalo (-1,1)

⇒ 2. Cálculo del límite lateral por la derecha de la función en el punto x=a

```
limitederecha = Limit[f[x], x → a, Direction -> -1];
```

⇒ 3. Cálculo del límite lateral por la izquierda de la función en el punto x=b

```
limiteizquierda = Limit[f[x], x → b, Direction -> 1];
```

⇒ 4. Estudio de la continuidad en el intervalo cerrado [a,b]

```
If[limitederecha == f[a] && limiteizquierda == f[b],
 Print["la función es continua en le intervalo [",
  a, ",", b, "]"], Print[
  "la función no es continua en le intervalo [", a, ",", b, "]"]]
```

```
la función es continua en le intervalo [-1,1]
```

3.- LOCALIZACIÓN DE RAÍCES DE ECUACIONES

Descripción del método

- 1. Definición de la función
- 2. Localización de la raíz
- 3. Representación gráfica de la función en el intervalo calculado

Procedimiento

Localizar el intervalo en el que existe una raíz de la función $f(x) = x^5 + x - 1$

⇒ 1. Definición de la función

```
f[x_] = x^5 + x - 1;
```

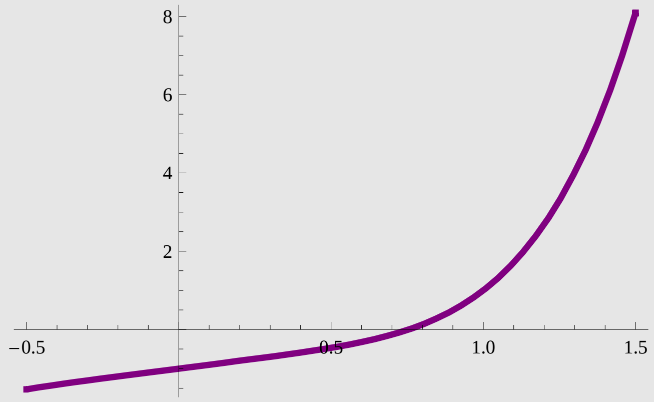
⇒ 2. Localización de la raíz

```
For[k = -10, k < 10, k++, {c, d} = {k, k + 1}; If[f[c] * f[d] < 0,
  Print["La raiz buscada está en el intervalo ", {c, d}]; k = 10,
  If[f[c] * f[d] == 0, If[f[c] == 0, Print["La raiz buscada es c=",
     c], Print["La raiz buscada es d=", d]]]]]
```

```
La raiz buscada está en el intervalo {0, 1}
```

⇒ 3. Representación gráfica de la función en el intervalo calculado

```
ϵ = 0.5;
Plot[f[x], {x, c - ϵ, d + ϵ},
 PlotStyle → {{Purple, Thickness[0.01]}}]
```



4.- MÉTODO DE BISECCIÓN

Descripción del método

- 1. Definición de los datos iniciales
- 2. Localización de la raíz
- 3. Cálculo de las distintas soluciones aproximadas y sus errores
- 4. Presentación de Resultados

Procedimiento

Calcular de forma aproximada la raiz de la ecuación $e^{-x} - x = 0$ que se encuentra en el intervalo $(0, 1)$, por el método de Bisección con un error inferior a 10^{-3}.

⇒ 1. Definición de los datos iniciales

Definimos la función y el error que podemos cometer y calculamos el número de iteraciones para obtener la raíz

```
f[x_] = E^-x - x;
```

```
it[p_] = IntegerPart[p * Log[10.] / Log[2.]] + 1;
```

```
p = 3;
er = 10^-p;
i = it[p] + 1;
```

⇒ 2. Localización de la raiz

Utilizamos el procedimiento 3, visto anteriormente, para la localización del intervalo en el que se encuentra la raíz.

```
For[k = -10, k < 10, k++, {c, d} = {k, k + 1}; If[f[c] * f[d] < 0,
  Print["La raiz buscada está en el intervalo {c,d}=", {c, d}];
  k = 10, If[f[c] * f[d] == 0,
   If[f[c] == 0, Print["La raiz buscada es c=", c],
    Print["La raiz buscada es d=", d]]]]]
```

```
La raiz buscada está en el intervalo {c,d}={0, 1}
```

Utilizamos en comando FindRoot para obtener una aproximación numérica de la raíz buscada.

```
FindRoot[f[x] == 0, {x, c}]
```

```
{x → 0.567143}
```

⇒ 3. Cálculo de las distintas soluciones aproximadas y sus errores

```
{a[0],b[0]}={c,d};
```

```
For[n = 1, n < i, n++, x[n] = (a[n - 1] + b[n - 1]) / 2.;
 e[n] = (b[n - 1] - a[n - 1]) / 2.;
 If[f[a[n - 1]] * f[x[n]] < 0, Print["El nuevo intervalo es:"];
  Print[" {a[n],b[n]}=", {a[n], b[n]} = {a[n - 1], x[n]}],
  Print["El nuevo intervalo es:"];
  Print[ " {a[n],b[n]}=", {a[n], b[n]} = {x[n], b[n - 1]}]];
 If[f[x[n]] == 0, Print["La raiz buscada es: {x[n],e[n]}=",
   {x[n], e[n]}]; n = i,
  If[e[n] < er, Print["La raiz aproximada es:"];
   Print[" {x[n],e[n]}=", {x[n], e[n]}];
   Print[" en ", n, " iteraciones"]; n = i,
   Print["La nueva aproximación es:"];
   Print[" {x[n],e[n]}=", {x[n], e[n]}]; Print[
    " en la iteración ", n]; Print["**********************"]]]]
```

```
El nuevo intervalo es:

 {a[n],b[n]}={0.5, 1}

La nueva aproximación es:

 {x[n],e[n]}={0.5, 0.5}

 en la iteración 1

**********************

El nuevo intervalo es:

 {a[n],b[n]}={0.5, 0.75}

La nueva aproximación es:

 {x[n],e[n]}={0.75, 0.25}

 en la iteración 2

**********************

El nuevo intervalo es:

 {a[n],b[n]}={0.5, 0.625}
```

```
La nueva aproximación es:
 {x[n],e[n]}={0.625, 0.125}
 en la iteración 3
**********************
El nuevo intervalo es:
 {a[n],b[n]}={0.5625, 0.625}
La nueva aproximación es:
 {x[n],e[n]}={0.5625, 0.0625}
 en la iteración 4
**********************
El nuevo intervalo es:
 {a[n],b[n]}={0.5625, 0.59375}
La nueva aproximación es:
 {x[n],e[n]}={0.59375, 0.03125}
 en la iteración 5
**********************
El nuevo intervalo es:
 {a[n],b[n]}={0.5625, 0.578125}
La nueva aproximación es:
 {x[n],e[n]}={0.578125, 0.015625}
 en la iteración 6
**********************
```

El nuevo intervalo es:

{a[n],b[n]}={0.5625, 0.570313}

La nueva aproximación es:

{x[n],e[n]}={0.570313, 0.0078125}

en la iteración 7

****

El nuevo intervalo es:

{a[n],b[n]}={0.566406, 0.570313}

La nueva aproximación es:

{x[n],e[n]}={0.566406, 0.00390625}

en la iteración 8

****

El nuevo intervalo es:

{a[n],b[n]}={0.566406, 0.568359}

La nueva aproximación es:

{x[n],e[n]}={0.568359, 0.00195313}

en la iteración 9

****

El nuevo intervalo es:

{a[n],b[n]}={0.566406, 0.567383}

La raiz aproximada es:

{x[n],e[n]}={0.567383, 0.000976563}

en 10 iteraciones

⇒ 4. Presentación de Resultados

Recopilamos en forma de tabla los diferentes resultados: número de iteración, intervalo, solución aproximada y error cometido.

```
A={{"n","a[n-1]","b[n-1]","x[n]","e[n]","f[x[n]]"}};
B=Table[{n,a[n-1],b[n-1],x[n],e[n],f[x[n]]},{n,1,i-1}];
resultados=TableForm[Join[A,B]]
```

n	a[n−1]	b[n−1]	x[n]	e[n]	f[x[n]]
1	0	1	0.5	0.5	0.106531
2	0.5	1	0.75	0.25	−0.277633
3	0.5	0.75	0.625	0.125	−0.0897386
4	0.5	0.625	0.5625	0.0625	0.00728282
5	0.5625	0.625	0.59375	0.03125	−0.0414975
6	0.5625	0.59375	0.578125	0.015625	−0.0171758
7	0.5625	0.578125	0.570313	0.0078125	−0.00496376
8	0.5625	0.570313	0.566406	0.00390625	0.0011552
9	0.566406	0.570313	0.568359	0.00195313	−0.00190536
10	0.566406	0.568359	0.567383	0.000976563	−0.000375349

Son necesarias 10 iteraciones y al menos tres dígitos son exactos.

Representación gráfica

```
g2 = Plot[f[x], {x, 0.5, 0.75},
   PlotStyle → {Thickness[0.01], RGBColor[0, 1, 0.5]},
   DisplayFunction → Identity];
```

```
g[k_] := ListPlot[{{x[k], f[x[k]]},
    {x[k + 1], f[x[k]]}, {x[k + 1], f[x[k + 1]]}}, Joined → True,
   PlotStyle → {Thickness[0.01`], RGBColor[1, 0.5`, 0.1` k]},
   DisplayFunction → Identity];
```

```
Show[g /@ Table[k, {k, 1, i - 2}], g2, Frame → True,
 PlotRange → {-0.3`, 0.15`}, DisplayFunction → $DisplayFunction]
```