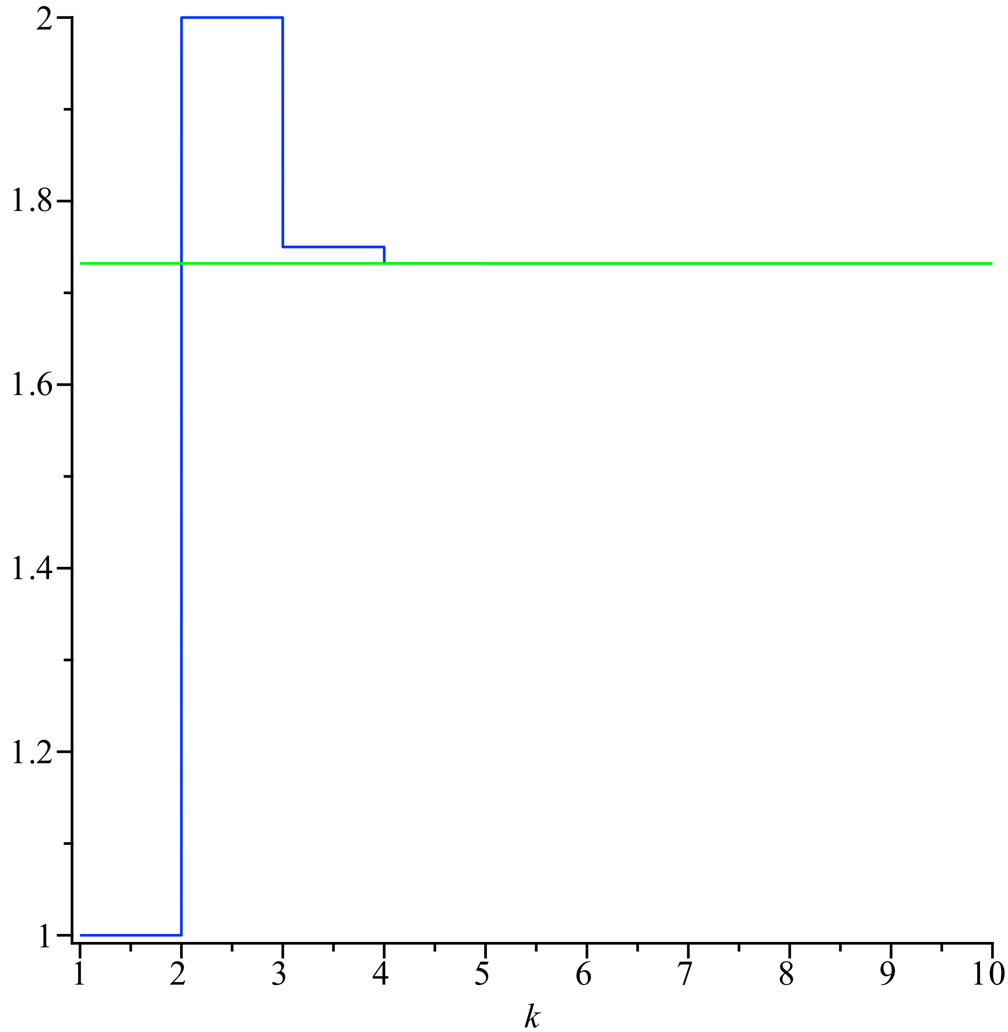
4.4 Exercices

Exercice 24

```
> restart;
> newton := proc(f, df, aa, bb, N)
  local a, b, q, k :
  global X :
  a := evalf(aa) : b := evalf(bb) :
  X := array(1 ..N) :
  X[1] := (a + b)/2 :
  for k from 1 to N − 1 do
  q := evalf(df(X[k])) :
  X[k + 1] := X[k] − evalf(f(X[k]))/q :
  od:
  [seq(X[k], k = 1 ..N)];
  end proc:
> g := x→x^2 − 3 : dg := x→2·x : N := 10 :
  newton(g, dg, 0, 2, N);
```

[1.000000000, 2.000000000, 1.750000000, 1.732142857, 1.732050810, 1.732050808, 1.732050808, 1.732050808, 1.732050808, 1.732050808] **(1.1.1)**

```
> plot([X[k], sqrt(3)], k = 1 ..N, color = [blue, green]);
```

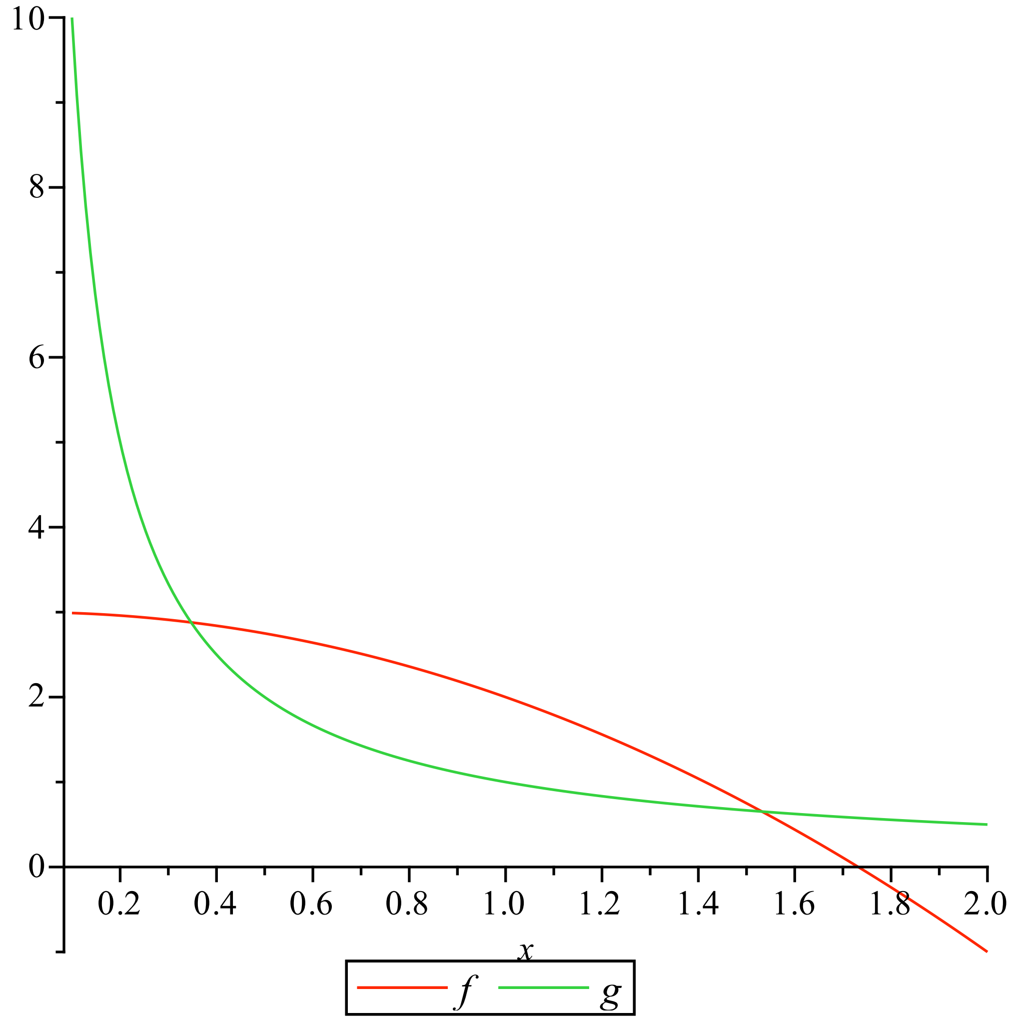


>

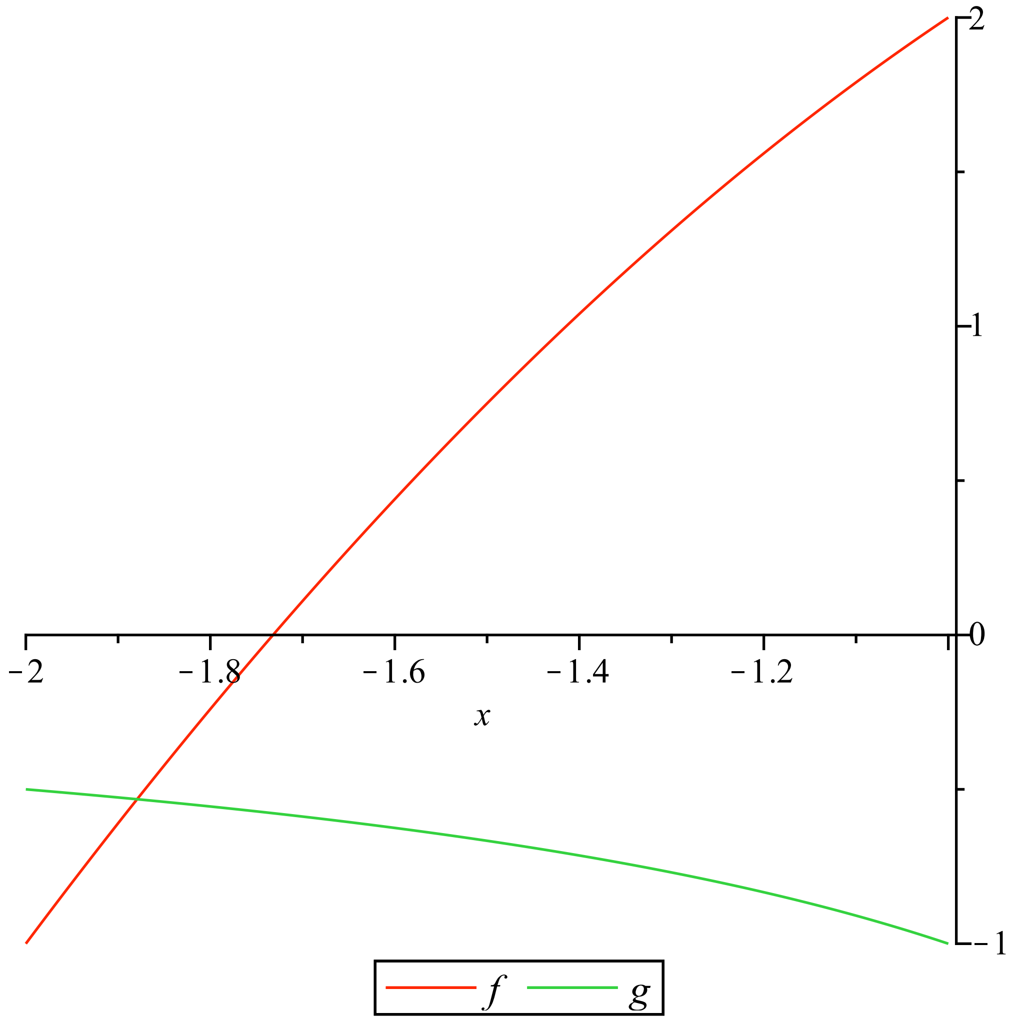
Exercice 25

> *#Illustration des 3 solutions*

> $f := x \to 3 - x^2 : g := x \to \frac{1}{x} :$
> $plot([f(x), g(x)], x = 0.1..2, legend = [f, g]);$



> $plot([f(x), g(x)], x = -2..-1, legend = [f, g]);$



> #On choisit d'utiliser une dichotomie pour trouver les solutions

> $dicho := \mathbf{proc}(f, aa, bb, N)$
local a, b, c, fa, fb, fc, n;
$a := evalf(aa)$:
$b := evalf(bb)$:
$c := \dfrac{(a+b)}{2}$:
$fa := evalf(f(a))$:
$fb := evalf(f(b))$:
for n **from** 2 **to** N **do**
$fc := evalf(f(c))$:
if $(fa \cdot fc < 0)$ **then**
$b := c$:
$fb := evalf(f(c))$:
elif$(fc \cdot fb < 0)$ **then**
$a := c$:
$fa := evalf(f(c))$:
else return c : **fi**:

$c := \frac{(a+b)}{2}$:

od:

c :

end proc:

> $h := x \to f(x) - g(x)$:

> #En évaluant h en x=-2 et x=0^- on voit que h change de signe, il y a une solution entre -2 et 0. On capture cette soltution
#par dichotomie

> $N := 1000$:
$x1 := dicho(h, -2, -0.0001, N)$;

$$x1 := -1.879385242 \quad (1.2.1)$$

>

> #En évaluant h en x=0^+ et $x = 1$ on voit que h change de signe, il y a une solution entre 0 et 1. On capture cette soltution
#par dichotomie

> $x2 := dicho(h, 0.000001, 1, N)$;

$$x2 := 0.3472963553 \quad (1.2.2)$$

> #En évaluant h en x=1 et x=2 on voit que h change de signe, il y a une solution entre 1 et 2. On capture cette soltution
#par dichotomie

> $x3 := dicho(h, 1, 2, N)$;

$$x3 := 1.532088886 \quad (1.2.3)$$

>

Exercice 26

> #On a deux équations.

> #La longueur de l'arc circulaire vaut 2.1 ce qui se traduit par
$\theta \cdot r = 2.1$
#où theta est l'angle POQ et r est le rayon du cercle

> #La distance en ligne droite entre les deux points vaut 2cm ce qui se traduit par
$\#sin\left(\frac{\text{theta}}{2}\right) = \frac{2}{2 \cdot r} = \frac{1}{r}$

#En appliquant une formule de trigo au triangle de sommets P et Q et ayant le diamètre du cercle pour base.

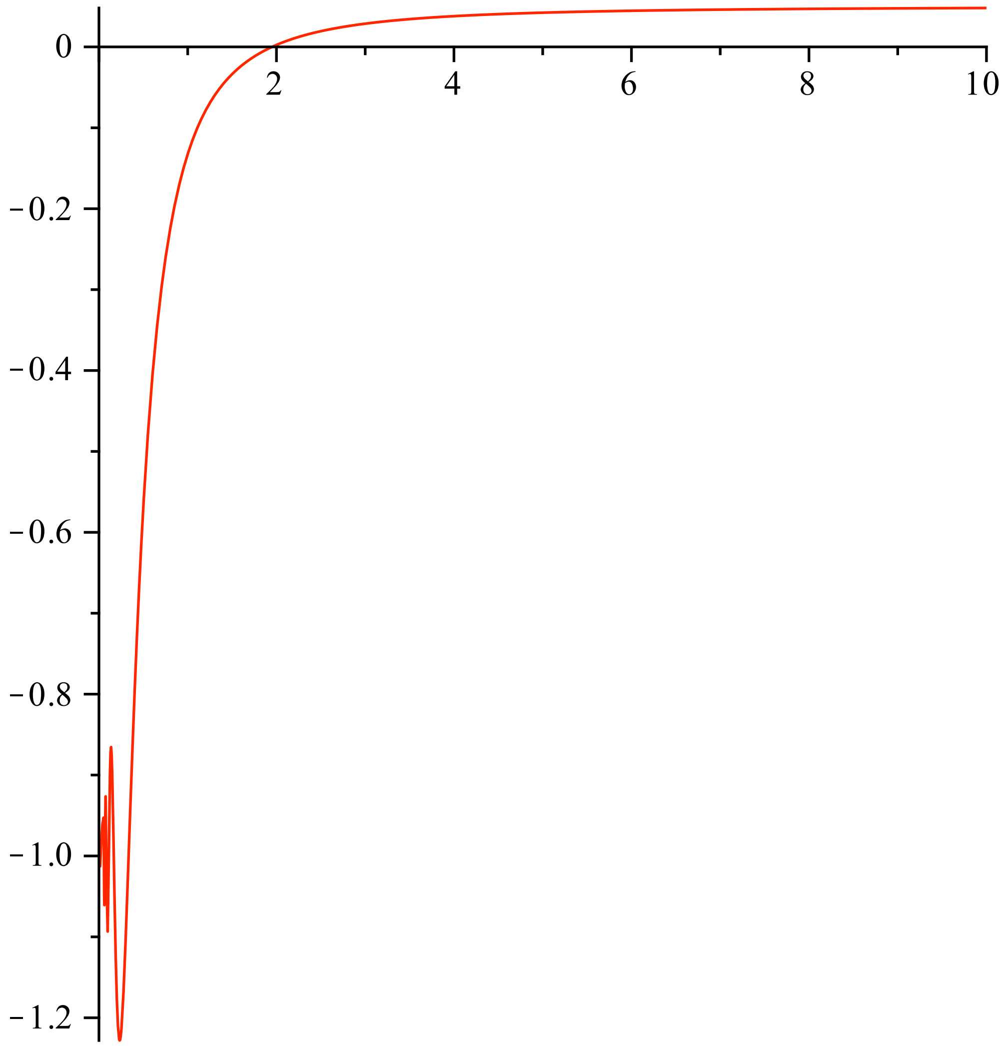
> #On doit donc trouver $r > 0$ satisfaisant
$\#r \cdot sin\left(\frac{2.1}{2}\right) = 1$

Error, missing denominator

#On doit donc trouver $r > 0$ satisfaisant

$\#r*sin(\frac{2.1}{\;})= 1$

> $h := r \rightarrow r \cdot \sin\left(\frac{2.1}{2 \cdot r}\right) - 1 :$
$plot(h, 0..10);$



> $Digits := 30 :$

> $dicho :=$ **proc**(f, aa, bb, N)
local $a, b, c, fa, fb, fc, n;$
$a := evalf(aa) :$
$b := evalf(bb) :$
$c := \frac{(a + b)}{2} :$
$fa := evalf(f(a)) :$
$fb := evalf(f(b)) :$
for n **from** 2 **to** N **do**
$fc := evalf(f(c)) :$
if $(fa \cdot fc < 0)$ **then**

```
b := c :
fb := evalf(f(c)) :
elif(fc·fb < 0) then
a := c :
fa := evalf(f(c)) :
else return c : fi:
c := (a + b)/2 :
od:
c :
end proc:
```

> $N := 1000$:

> $r1 := dicho(h, 0.001, 10, N)$;

$$r1 := 1.9501806033298952049766923340\,6 \tag{1.3.1}$$

> #Verification

> $theta1 := \frac{2.1}{r1}$;

$$\theta 1 := 1.0768233446760218555944771388\,7 \tag{1.3.2}$$

> $r1 \cdot \sin\left(\frac{theta1}{2}\right)$;

$$1.00000000000000000000000000000 \tag{1.3.3}$$

>

5.7 Exercices

Exercice 30

> restart : with(plots) :

> Digits := 100 :

> #Solution par la méthode d'Euler explicite

```
> penduleEuler := proc(t0, T, theta0, thetaprim0, N)
local h, y, i, g, l :
global thetaeul :
h := evalf((T − t0)/N) :
g := 9.8 : l := 1. :
thetaeul := array(1 ..N) :
y := array(1 ..N) :
thetaeul[1] := theta0 :
y[1] := thetaprim0 :
for i from 1 to N − 1 do
thetaeul[i + 1] := evalf(thetaeul[i] + h·y[i]) :
```

$y[i+1] := evalf\left(y[i] - \frac{h \cdot g \cdot \sin(thetaeul[i])}{l}\right):$

od:

end proc:

\>

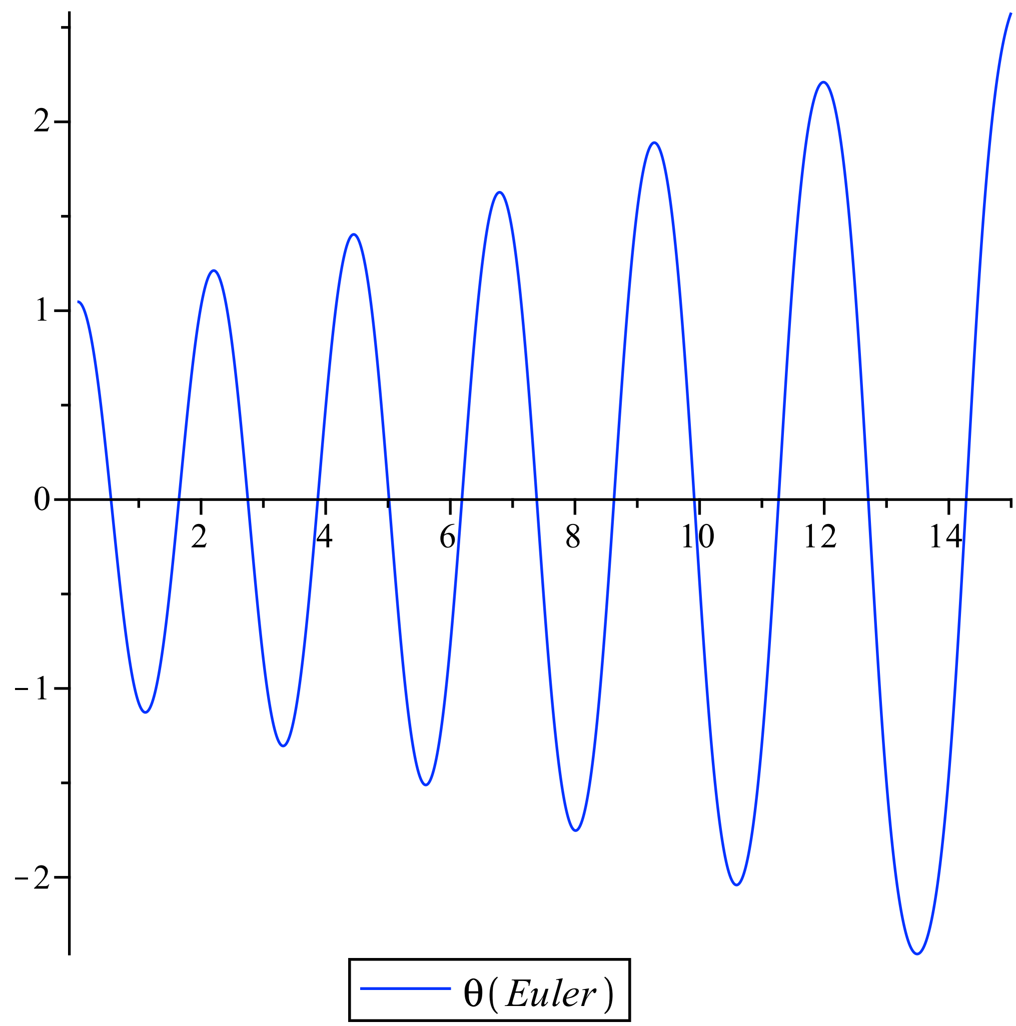
\> $t0 := 0 : T := 15; N := 1000 :$

$$T := 15 \tag{2.1.1}$$

\> $penduleEuler\left(t0, T, \frac{\text{Pi}}{3}, 0, N\right):$

\> $G1 := plot\left(\left[seq\left(t0 + n \cdot \frac{(T - t0)}{N}, n = 1..N\right)\right], [seq(thetaeul[n], n = 1..N)], color = blue, legend = \text{theta}(Euler)\right):$

\> $display(G1);$



\> *#Solution par la méthode de Runge explicite*

\> $penduleRunge :=$ **proc**$(t0, T, theta0, thetaprim0, N)$

local $h, y, i, g, l, k1a, k1b, k2a, k2b$:

global $thetaR$:

$h := evalf\left(\frac{(T - t0)}{N}\right):$
$g := 9.8 : l := 1 :$
$thetaR := array(1..N):$
$y := array(1..N):$
$thetaR[1] := theta0:$
$y[1] := thetaprim0:$
for i **from** 1 **to** $N - 1$ **do**
$k1a := y[i]:$

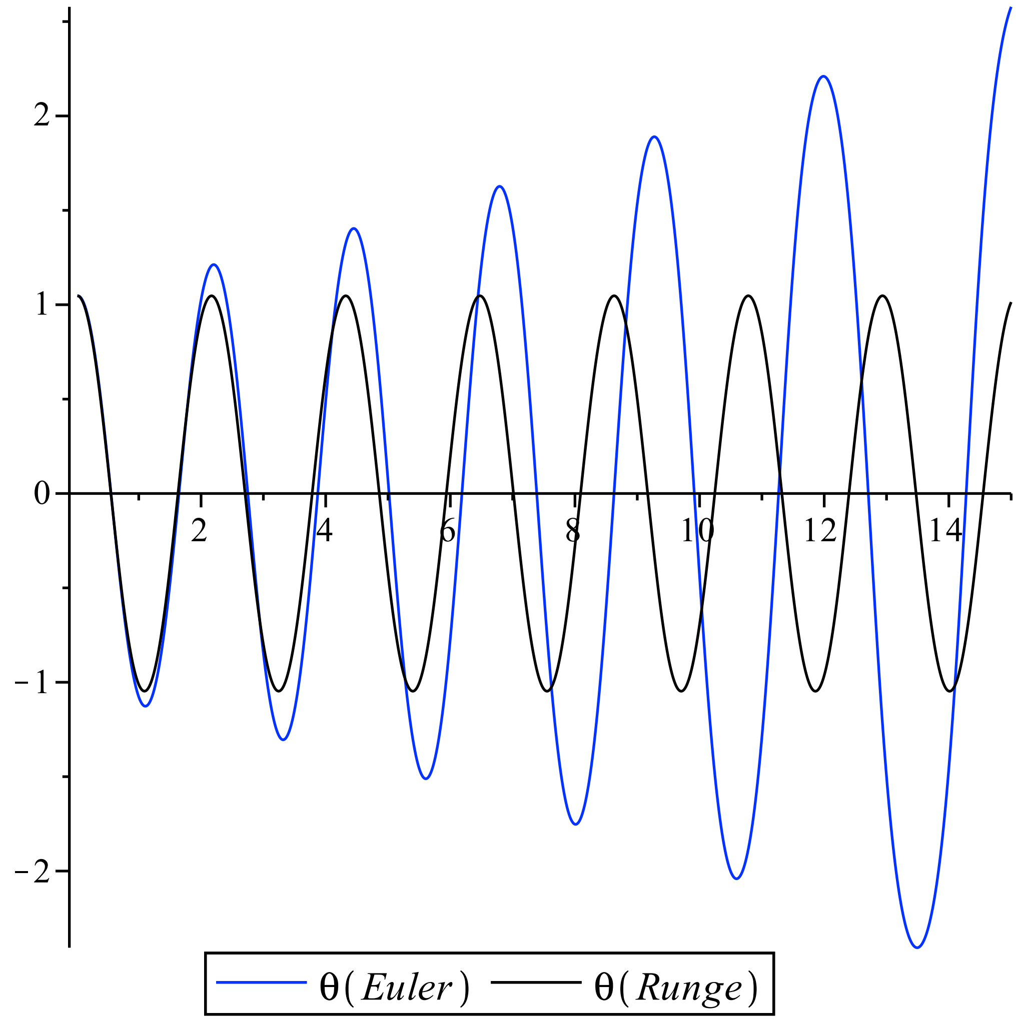
$k1b := evalf\left(-\frac{g \cdot \sin(thetaR[i])}{l}\right):$
$k2a := y[i] + \frac{h}{2} \cdot k1b:$
$k2b := evalf\left(-\frac{g \cdot \sin\left(thetaR[i] + \frac{h}{2} \cdot k1a\right)}{l}\right):$
$thetaR[i+1] := evalf(thetaR[i] + h \cdot \mathrm{k2a}):$
$y[i+1] := evalf(y[i] + h \cdot k2b):$
od:
end proc:

>

> $penduleRunge\left(t0, T, \frac{\mathrm{Pi}}{3}, 0, N\right):$

> $G2 := plot\left(\left[seq\left(t0 + n \cdot \frac{(T - t0)}{N}, n = 1..N\right)\right], [seq(thetaR[n], n = 1..N)], color = black, legend = \mathrm{theta}(Runge)\right):$

> $display(G1, G2);$

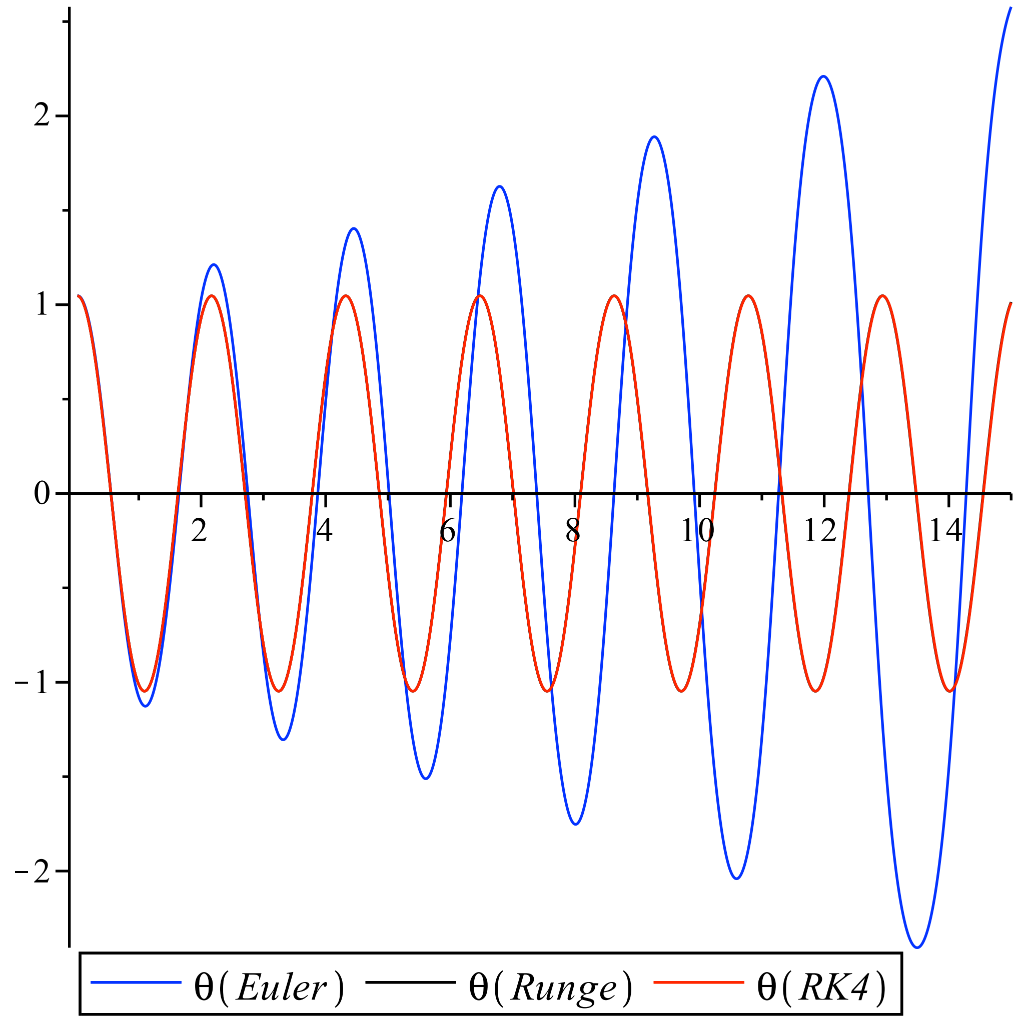


> #Solution par la méthode de Runge-Kutta 4

> $penduleRK4 := \mathbf{proc}(t0, T, theta0, thetaprim0, N)$
$\mathbf{local}\ h, y, i, g, l, k1a, k2a, k3a, k4a, k1b, k2b, k3b, k4b :$
$\mathbf{global}\ thetaRK :$
$h := evalf\left(\frac{(T - t0)}{N}\right) :$
$g := 9.8 : l := 1. :$
$thetaRK := array(1..N) :$
$y := array(1..N) :$
$thetaRK[1] := theta0 :$
$y[1] := thetaprim0 :$
$\mathbf{for}\ i\ \mathbf{from}\ 1\ \mathbf{to}\ N - 1\ \mathbf{do}$
$k1a := \mathrm{y[i]} :$
$k1b := -\frac{g}{l} \cdot \sin(thetaRK[i]) :$
$k2a := y[i] + \frac{h}{2} \cdot k1b :$
$k2b := -\frac{g}{l} \cdot \sin\left(thetaRK[i] + \frac{h}{2} \cdot k1a\right) :$

```
k3a := y[i] + (h/2)·k2b :
k3b := - (g/l)·sin(thetaRK[i] + (h/2)·k2a) :

k4a := y[i] + h·k3b :
k4b := - (g/l)·sin(thetaRK[i] + h·k3a) :
thetaRK[i + 1] := evalf(thetaRK[i] + h·(k1a + 2·k2a + 2·k3a + k4a)/6) :
y[i + 1] := evalf(y[i] + h·(k1b + 2 ·k2b + 2·k3b + k4b)/6) :
od:
end proc:
>
> penduleRK4(t0, T, Pi/3, 0., N) :
> G3 := plot([seq(t0 + n·(T − t0)/N, n = 1 ..N)], [seq(thetaRK[n], n = 1 ..N)],
    color = red, legend = theta(RK4)) :
> display(G1, G2, G3);
```



> #On remarque que le schéma de Runge et RK4 donne des résultats nettement meilleurs que le schéma d'Euler explicite dans lequel l'amplitude des oscillation s'amplifie, ce qui est absurde.

> #On ne constate pas de différence entre le schéma de Runge explicite et RK4, on va donc réduire le nombre de points pour voir si on obtient une différence:

> $N2 := 100:$

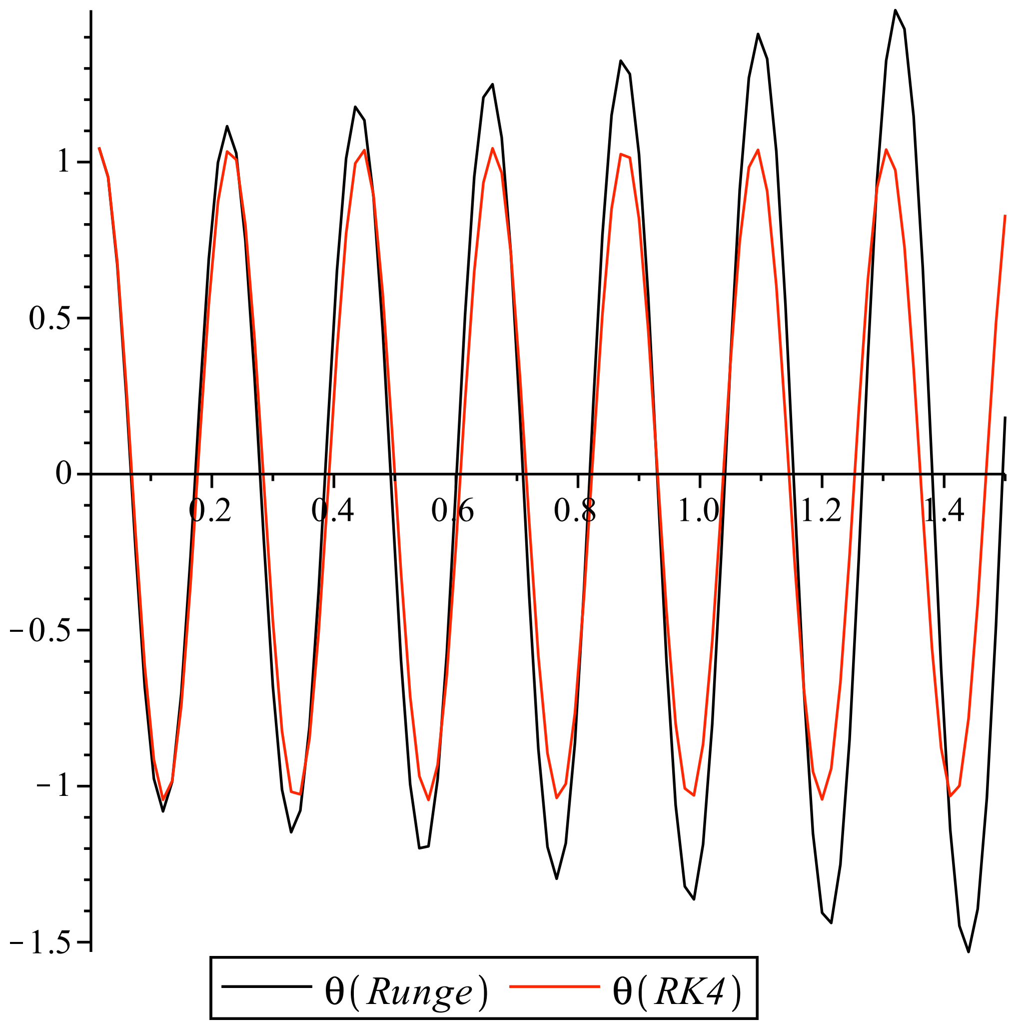
> $penduleRunge\left(t0, T, \frac{\text{Pi}}{3}, 0, N2\right):$

> $G5 := plot\left(\left[seq\left(t0 + n \cdot \frac{(T - t0)}{N}, n = 1..N2\right)\right], [seq(thetaR[n], n = 1..N2)], color = black, legend = \text{theta}(Runge)\right):$

> $penduleRK4\left(t0, T, \frac{\text{Pi}}{3}, 0., N2\right):$

> $G6 := plot\left(\left[seq\left(t0 + n \cdot \frac{(T - t0)}{N}, n = 1..N2\right)\right], [seq(thetaRK[n], n = 1..N2)], color = red, legend = \text{theta}(RK4)\right):$

> *display(G5, G6);*



> *#On constate que RK4 donne une meilleure approximation de la solution que le schéma de Runge explicite.*

>