

Metodo di Heun

Enrico Bertolazzi

[-] Carica le librerie

```
> restart ;  
with(plots):  
Warning, the name changecoords has been redefined
```

[-] Definisce la procedura Heun

```
> heun := proc(f, x0, y0, h, n)  
  local i, xi, yi, xx, yy, res ;  
  xi := evalf(x0) ;  
  yi := evalf(y0) ;  
  # inizializza la lista che conterra` la soluzione  
  res := [ [xi, yi] ] ;  
  for i from 1 to n do  
    # passo di Eulero  
    yy := evalf( yi + h * f(xi,yi) ) ;  
    xx := evalf( xi + h ) ;  
    # correzione  
    yi := evalf( yi + (h/2) * (f(xi,yi)+f(xx,yy)) ) ;  
    xi := xx ;  
    # aggiunge il punto calcolato alla lista  
    res := [ op(res), [xi, yi] ] ;  
  end do;  
  return res ;  
end proc ;
```

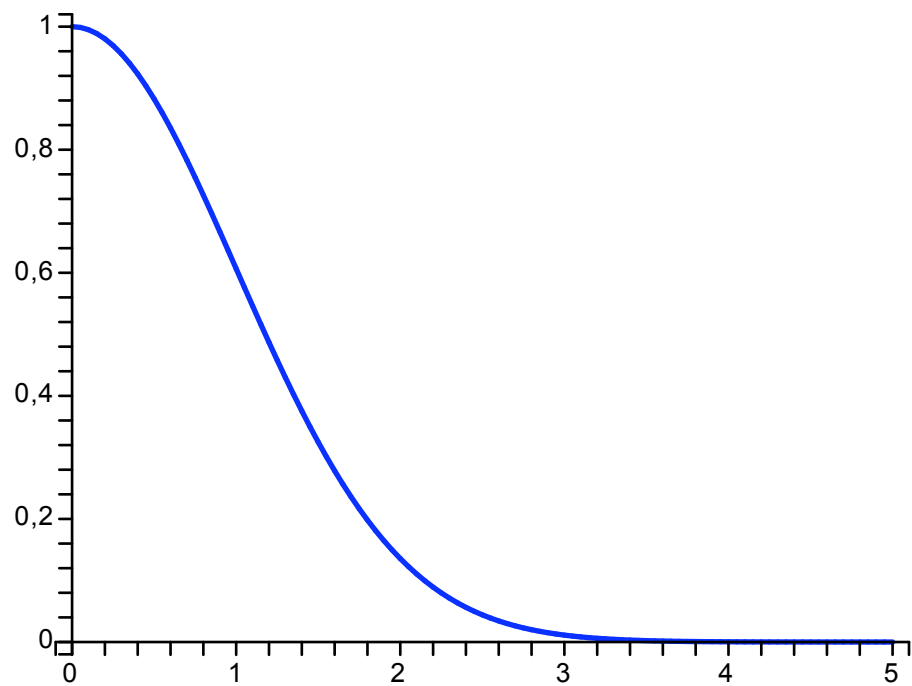
```
heun := proc(f, x0, y0, h, n)  
local i, xi, yi, xx, yy, res;  
xi := evalf(x0);  
yi := evalf(y0);  
res := [[xi, yi]];  
for i to n do yy := evalf(yi + h*f(xi, yi));  
xx := evalf(xi + h);  
yi := evalf(yi + 1/2*h*(f(xi, yi) + f(xx, yy)));  
xi := xx;  
res := [op(res), [xi, yi]];  
end do;  
return res;  
end proc;
```

[-] Esempio d'uso

```
> # Definisce la funzione da approssimare
f := (x,y) -> -x*y ;
                                     f := (x, y) → -y x

> # risolve il problema
pts := heun(f,0,1,0.05,100) :

> # disegna la soluzione
plot(pts,style=line,thickness=2,color=blue);
```



```
>
```