```
> restart:
```
We show how to find the first 30 iterations of some function f starting from x0. Than we represent them using the command pointplot.

```
> x:=x0;
> for i from 1 to 30 do x:=f(x): psi(i):=x: print(x); od:
> points:=[[n,psi(n)]$n=1..30]:with(plots): pointplot(points,
  symbol=circle);
>
>
>
>
> restart:
```

We will apply Euler's method to solve the initial value problem y'=2xy+exp(x^2), y(0)=1, in the interval [0,1.5] with step size h=0.1. Since we would like to compare with the exact values, first we find the exact solution of this IVP and we represent the graph of the solution and the direction field of the differential equation.

```
> with(DEtools):
> f:=(x,y)->2*x*y+exp(x^2);
```
this is the right-hand side of the differential equation

```
> dsolve({diff(y(x),x)=f(x,y(x)),y(0)=1});phi:=unapply(rhs(%),x);
> DEplot(diff(y(x),x)=f(x,y(x)),y(x),x=0.. 1.5, [[y(0)=1]],
  y=1..25);
> h:=0.1;
```
this is the stepsize. Hence, the number of steps necessary to cover the interval [0, 1.5] is 15

```
> x:=0; y:=1;
```
(0,1) is the starting point, as given by the initial condition

```
> for i from 1 to 15 do y:=y+h*f(x,y): psi(i):=y: x:=x+h:
  print(x,y,phi(x),abs(y-phi(x))); od:
> points:=[[n,psi(n)]$n=1..15]:with(plots): pointplot(points,
  symbol=point);plot(phi(t),y=1..1.5);
> restart:
```
we will apply the now the improved Euler's numerical method. We have to change only the numerical formula. We must restart!

```
> phi:=x->(x+1)*exp(x^2);
> h:=0.1; x:=0; y:=1;
> f:=(x,y)->2*x*y+exp(x^2);
> for i from 1 to 15 do y:=y+h/2*f(x,y)+h/2*f(x+h,y+h*f(x,y)):
  psi(i):=y: x:=x+h: print(x,y,phi(x),abs(y-phi(x))); od:
> points:=[[n,psi(n)]$n=1..15]:with(plots): pointplot(points,
  style=point);plot(phi(t),t=1..1.5);
```