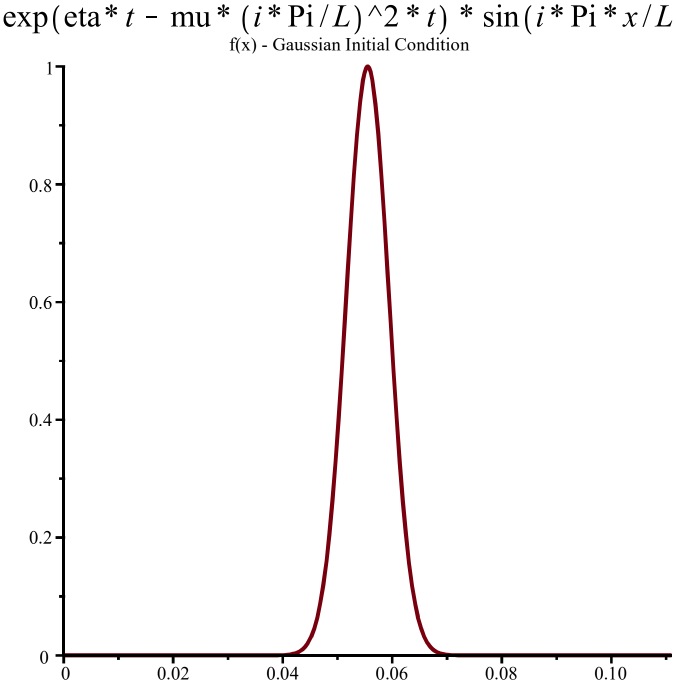
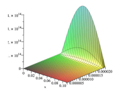
```
> restart;

 # --- Initial condition: Gaussian profile`
f := x → A * exp( -lambda * ( (x - L/2) / (L/2) )^2 ) :
L0 := 0.111 :
L := L0 :
A := 1 :
lambda := 100 :

 # Plot the Gaussian function
plot( f(x), x = 0 ..L, title = "f(x) - Gaussian Initial Condition" );

 #  Symbolic solution of Fourier coefficient a[p]
eqn3 := (2/L) * Int( f(x) * sin( p * Pi * x / L), x = 0 ..L) =
        (2/L) * int( f(x) * sin( p * Pi * x / L), x = 0 ..L) assuming p :: integer :

 # -----------------------------------
unassign('p') :
N := 30 :
L := L0 :
lambda := 100 :
mu := 2.3446e+05 :
eta := 1.8958e+08 :
Lcrit := evalf(Pi * sqrt(mu / eta) ) :

for p from 1 by 1 to N do
    aa[p] := evalf(rhs(eqn3));
end do:

 # -- Time varying solution: n(x,t)`
n := sum(aa[i] * exp(eta* t - mu * (i * Pi / L)^2 * t) * sin(i * Pi * x / L), i = 1 ..N) :
```

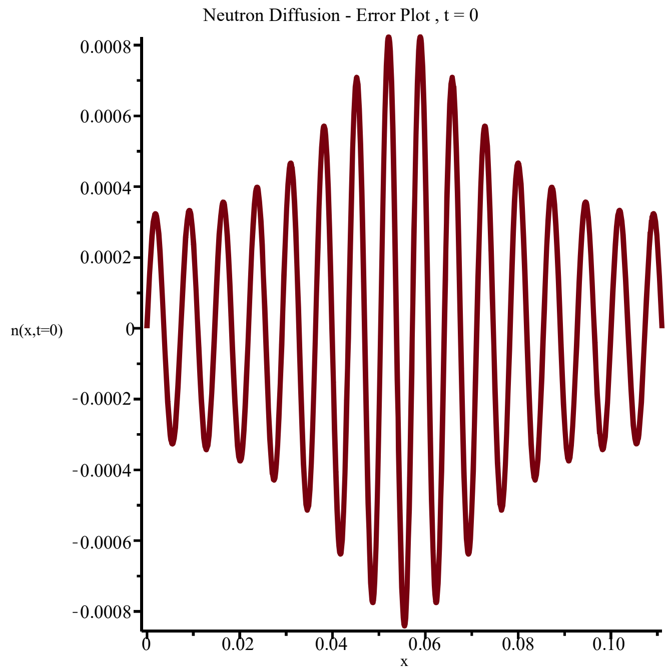


```
> plot3d(n, x = 0 ..L, t = 0 ..2e-5,
        axes = framed,
        title = "Neutron Diffusion - L = 11.1 cm, N = 30",
        labels = ["x", "t", "n(t,x)"],
        orientation = [-48, 69, 1]);
```

Neutron Diffusion - L = 11.1 cm, N = 30



```
> plot(subs(t = 0, n) - f(x), x = 0 ..L,
      axes = framed,
      title = "Neutron Diffusion - Error Plot , t = 0",
      labels = ["x", "n(x,t=0)"],
      thickness = 2);
```



Neutron Diffusion - Error Plot , t = 0
0.0008
0.0006
0.0004
0.0002
n(x,t=0)
0
-0.0002
-0.0004
-0.0006
-0.0008
0
0.02
0.04
0.06
0.08
0.10
x