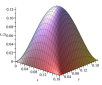
```
> restart :
  with(Physics) :
  with(plots) :

  # --- Parameters and Initial Condition ---
  L0 := 19.2 / 100 : # Length in meters (converted from cm)
  L := L0 :
  A := 1 :
  xi := 100 :
> f := (x, y, z) → (1 - (x/L)) * (1 - (y/L)) * (1 - (z/L)) * x * y * z / (L/2)^3 :
> f2 := value(subs(z = L/2, f(x, y, z))) :
> unassign('L') :
> eqn3 := (8/L^3) *
      int(int(int(f(x, y, z) * sin(p * Pi * x/L) * sin(q * Pi * y/L) * sin(r * Pi * z/L),
      x = 0..L), y = 0..L), z = 0..L) :

> eqn3 := simplify(eqn3, size) assuming p :: integer, q :: integer, r :: integer :

> unassign('p', 'q', 'r') :
> L := L0 :
  mu := 2.3446e5 :
  eta := 1.8958e8 :
> Lcrit := evalf(Pi * sqrt(3 * mu / eta)) :
> N := 4 :
> for p from 1 to N do
    for q from 1 to N do
      for r from 1 to N do
        aa[p, q, r] := evalf(subs([L = L0, p = p, q = q, r = r], eqn3));
      end do:
    end do:
  end do:
> n := add(add(add(
      aa[i, j, k] *
      exp(eta * t - mu * Pi^2 * ((i/L)^2 + (j/L)^2 + (k/L)^2) * t) *
      sin(i * Pi * x/L) * sin(j * Pi * y/L) * sin(k * Pi * z/L),
      i = 1..N), j = 1..N), k = 1..N) :
Warning, a multi-line expression was interpreted as each line
being multiplied together; use a semi-colon to split the
expression into separate statements if desired, or use an
explicit * to eliminate this warning
> L := L0 :
  z := L/2 :
  t := 2e-7 :
> plot3d(
    n, x = 0..L, y = 0..L,
    axes = framed,
    title = "3D Neutron Diffusion - L = 19.2 cm, N = 4, t = 2e-7",
```

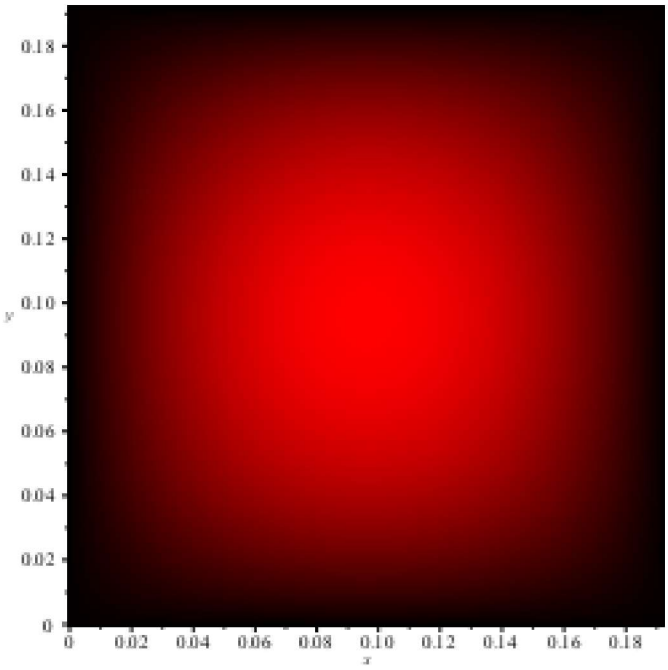
```
 labels = ["x", "y", "n(t=2e-7, x, y, z=L/2)"],
 orientation = [ -48, 69, 1 ]
);
```


```
> t := 0 :
plot3d(
 f2, x = 0 ..L, y = 0 ..L,
 axes = framed,
 title = "Neutron Distribution at t = 0",
 labels = ["x", "y", "f(x,y,z=L/2)"],
 orientation = [ -48, 69, 1 ]
);
```

Neutron Distribution at t = 0



```
> densityplot(
   f2, x = 0 ..L, y = 0 ..L,
   grid = [ 125, 125 ],
   style = patchnogrid,
   color = red
  );
```


0.18
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0
y
0
0.02
0.04
0.06
0.08
0.10
0.12
0.14
0.16
0.18
x

>