In[◦]:=
```
α = 1;
```

1) Problema Dirichlet (bordi a temperatura fissata = 0)

Problem: a bar of length L = 1, has initial temperature distribution at t=0 given by u(x,0) = 50 x (1-x), x in [0,1]. The two ends of the bar are kept at temperature = 0. Find the temperature distribution at t>0 . Assume the diffusivity coefficient is $\alpha = 1$.

To solve this problem we use the Fourier method.

In[12]:=
```
(* Compute Fourier coefficients *)
c[n_] = Integrate[ 50 x (1 - x) Sin[ n Pi x], {x, 0, 1}] 2 //
  Simplify[#, Assumptions → Element[n, Integers]] &
```

Out[12]=

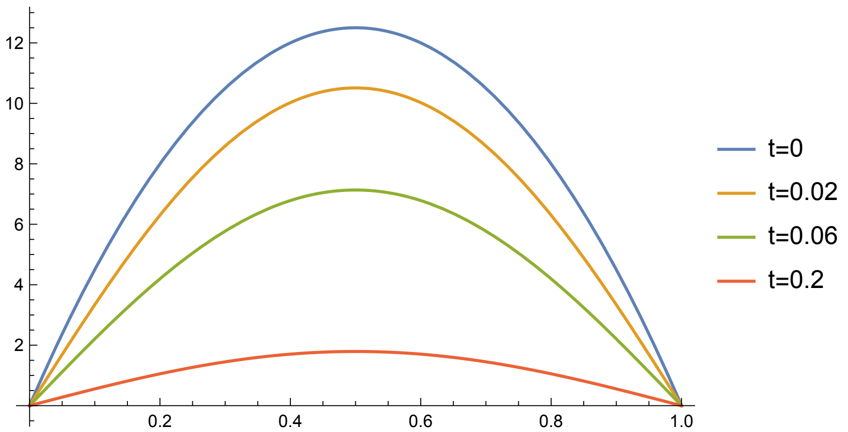
$$-\frac{200\left(-1+(-1)^{n}\right)}{n^{3}\,\pi^{3}}$$

In[13]:=
```
Clear[U]
α = 1;
U[x_, t_][NN_] := Sum[ Sin[n Pi x] E^(- n^2 Pi^2 α t) c[n], {n, 1, NN}]
```

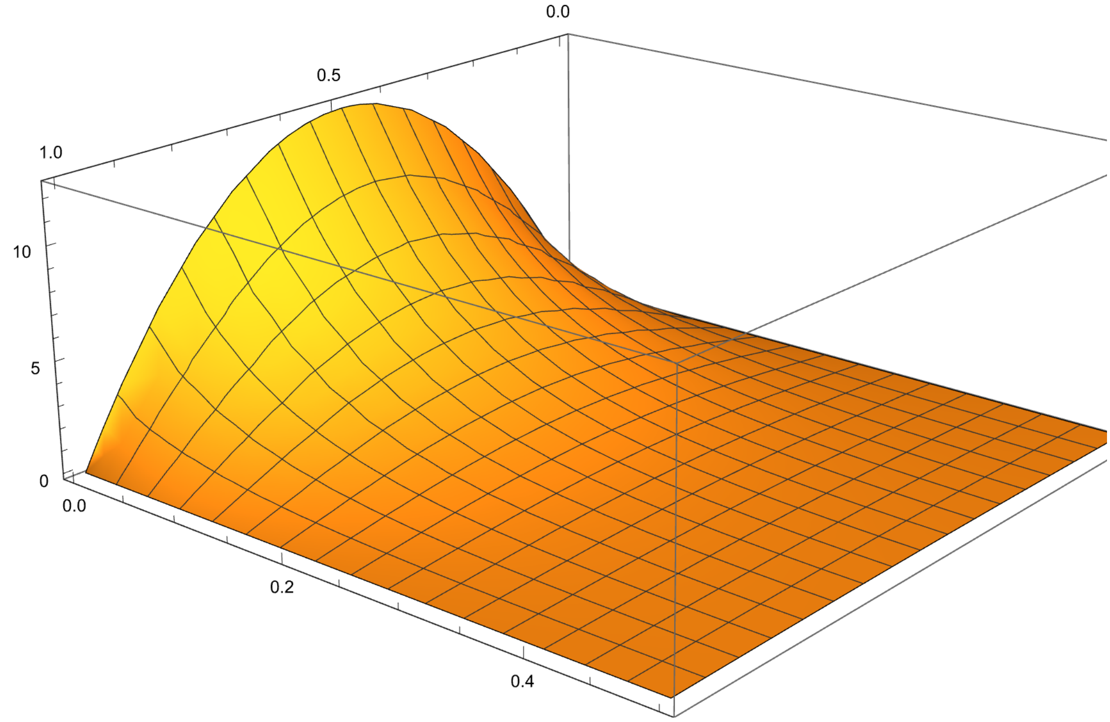
In[16]:=
```
Print["Solution for different times:"]
Quiet[Plot[{ U[x, 0.0][200],  U[x, 0.02][200],  U[x, 0.06][200], U[x, 0.2][200]} ,
  {x, 0, 1}, PlotLegends → {"t=0", "t=0.02", "t=0.06", "t=0.2"} ]]
```

```
Solution for different times:
```

Out[17]=



In[18]:=
```
Plot3D[ U[x, t] [200], {x, 0, 1}, {t, 0, 1 / 2}, PlotRange → All]
```
Out[18]=



2) Problema con condizione iniziale con un punto di non differenziabilita`

Problem: We consider the same parameters as above, but now the initial condition is a sharp triangle: u(x,0) = x for x in [0,1/2] and (1-x) for x in [1/2, 1].
We can still use the Fourier method in the same way. Notice that the Fourier series captures well solutions with isolated points of non differentiability.

In[19]:=
```
(* Compute Fourier coefficients *)
c[n_] = 2 (Integrate[ Sin[n Pi x] x, {x, 0, 1 / 2}] +
      Integrate[ Sin[n Pi x] (1 - x), {x, 1 / 2, 1}] ) //
  FullSimplify[#, Assumptions → Element[n, Integers]] &
```
Out[19]=

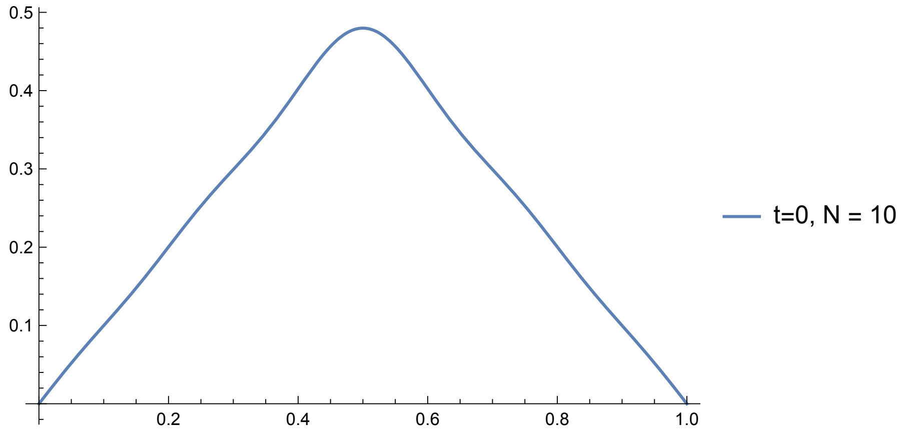
$$\frac{4\,\mathrm{Sin}\left[\frac{n\pi}{2}\right]}{n^2\,\pi^2}$$

In[20]:=
```
Clear[F4]
F4[x_, t_][NN_] := Sum[ Sin[n Pi x] E^(- n^2 Pi^2 α t) c[n], {n, 1, NN, 1}]
```

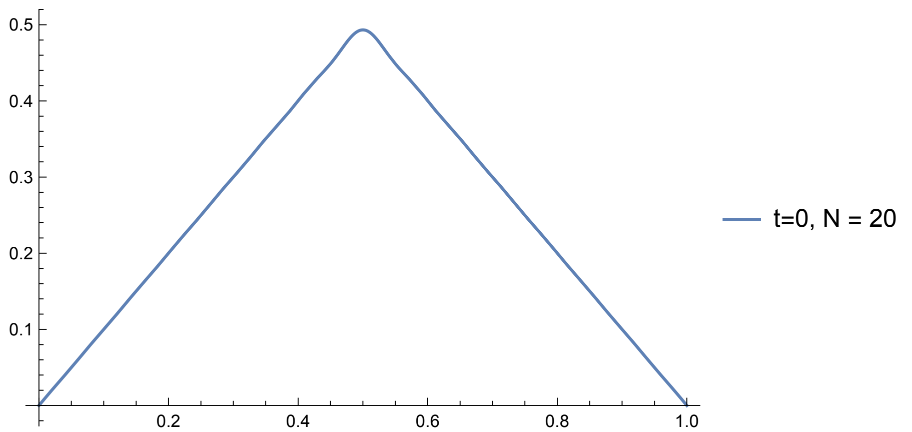
The following plots illustrate how the initial data is represented by the Fourier series. As we take more and more terms, the series reproduces the sharp angle of the initial data./

In[22]:=
```
Plot[{ F4[x, 0.0][10]} , {x, 0, 1}, PlotLegends → {"t=0, N = 10"} ]
Plot[{ F4[x, 0.0][30]} , {x, 0, 1}, PlotLegends → {"t=0, N = 20"} ]
Plot[{ F4[x, 0.0][100]} , {x, 0, 1}, PlotLegends → {"t=0, N = 100"} ]
```

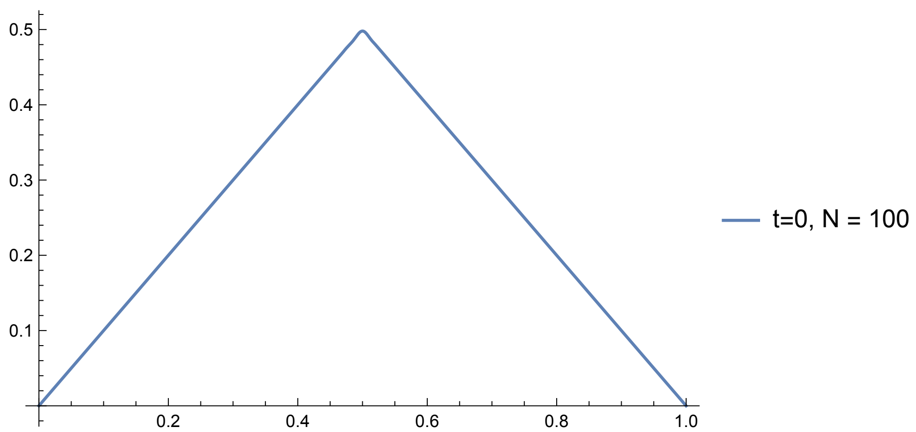
Out[22]=



Out[23]=



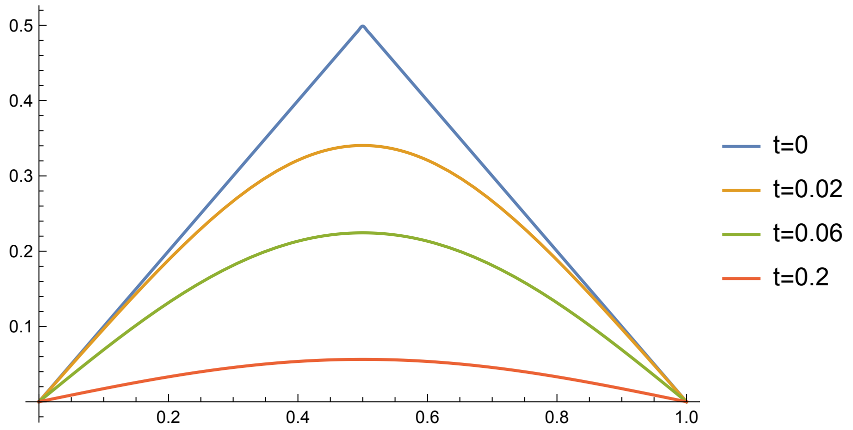
Out[24]=



Now we plot the solution at following times. Notice that the solution becomes smooth as soon as t>0.

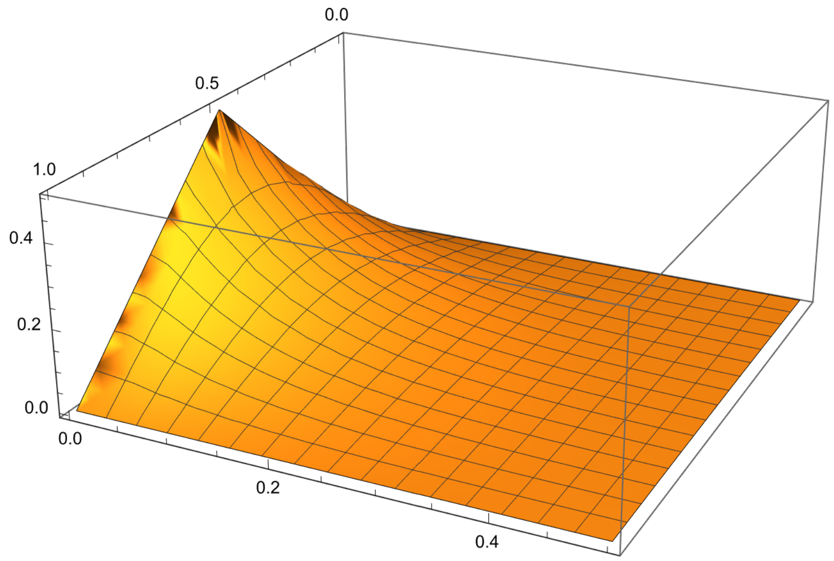
In[25]:=
```
Quiet[
 Plot[{ F4[x, 0.0][200],  F4[x, 0.02][200],  F4[x, 0.06][200],  F4[x, 0.2][200]} ,
  {x, 0, 1}, PlotLegends → {"t=0", "t=0.02",  "t=0.06",  "t=0.2"} ]]
```

Out[25]=



In[ ]:=
```
Plot3D[{F4[x, t][200]}, {x, 0, 1}, {t, 0, 1 / 2}, PlotRange → All]
```

Out[ ]=



3) Problema di Neumann (bordi isolati termicamente)

Problem: a bar of length L = 1, has initial temperature distribution at t=0 given by u(x,0) = 50 x (1-x), x in [0,1]. The two ends of the bar are kept insulated (= Neumann boundary condition). Find the temperature distribution at t>0 . Assume the diffusivity coefficient is $\alpha = 1$.

To solve this problem we use the Fourier method.

In[ ]:=
```
Clear[F5]
```

```
(* Find Fourier coeffcients *)
```

In[28]:=
```
(* The zeroth coefficient *)
a0 = 2 Integrate[ 1 x (1 - x), {x, 0, 1}] 50
```

Out[28]=

$$\frac{50}{3}$$

In[29]:=
```
(* the other coefficients *)
a[n_] = 2 Integrate[ Cos[n Pi x] x (1 - x), {x, 0, 1}] //
  Simplify[#, Assumptions → Element[n, Integers]] &
```

Out[29]=

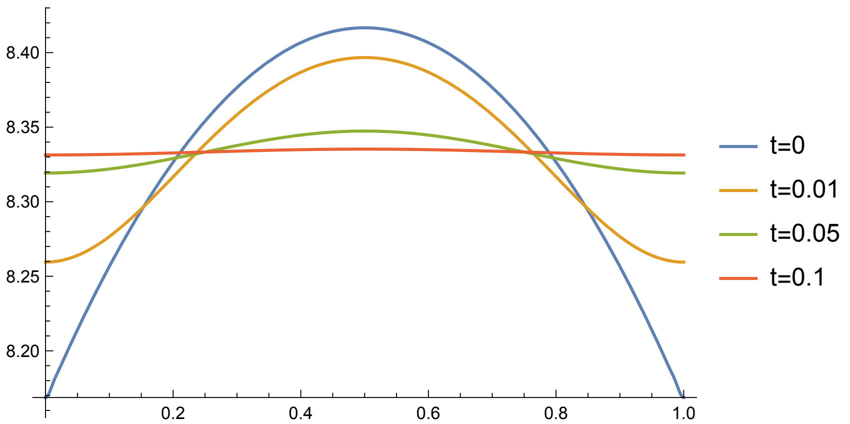
$$-\frac{2\left(1+(-1)^n\right)}{n^2\,\pi^2}$$

In[30]:=
```
F5[x_, t_][NN_] := a0 / 2 + Sum[ Cos[n Pi x] E ^ (- n ^ 2 Pi ^ 2 α t) a[n], {n, 1, NN, 1}]
```

Solution at different times. Notice that, despite the fact that the initial data did not satisfy the boundary condition (i.e., u_x was different from zero at the edges, the condition is satisfied as soon as t>0. This is due to the correct choice of eigenfunctions we expand on (the cosines). We can allow for such small violations of the boundary conditions in the initial data. The Fourier series will take care to restore them as t>0.

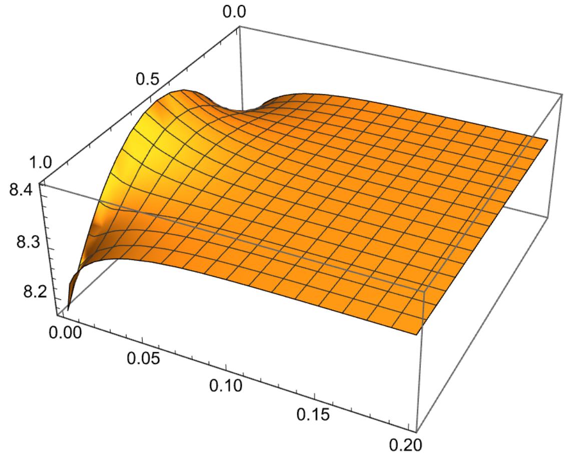
In[31]:=
```
Quiet[Plot[ {F5[x, 0][100], F5[x, 0.01][100],
   F5[x, 0.05][100], F5[x, 0.1][100]}, {x, 0, 1}, PlotRange → All,
  PlotLegends → {"t=0", "t=0.01", "t=0.05", "t=0.1"}]]
```

Out[31]=



In[32]:=
```
Plot3D[F5[x, t][100], {x, 0, 1}, {t, 0, 0.2}, PlotRange → All]
```

Out[32]=



Notice that the value of the temperature after a very long time is given by the mean of the initial data over the interval. This is due to the fact that with Neumann boundary conditions (no heat transfer at the boundaries) thermal energy is conserved. This conserved quantity is given by the integral of the temperature over the interval.

In[34]:=
```
a0 / 2
```

Out[34]=

$$\frac{25}{3}$$

In[35]:=
```
Integrate[(50 x (1 - x)), {x, 0, 1}]
```

Out[35]=

$$\frac{25}{3}$$

4) temperatura iniziale uniforme, raffreddamento dai bordi

In[36]:=
```
(* Compute Fourier coefficients *)
c[n_] = Integrate[Sin[n Pi x], {x, 0, 1}] 2 //
  Simplify[#, Assumptions → Element[n, Integers]] &
```

Out[36]=

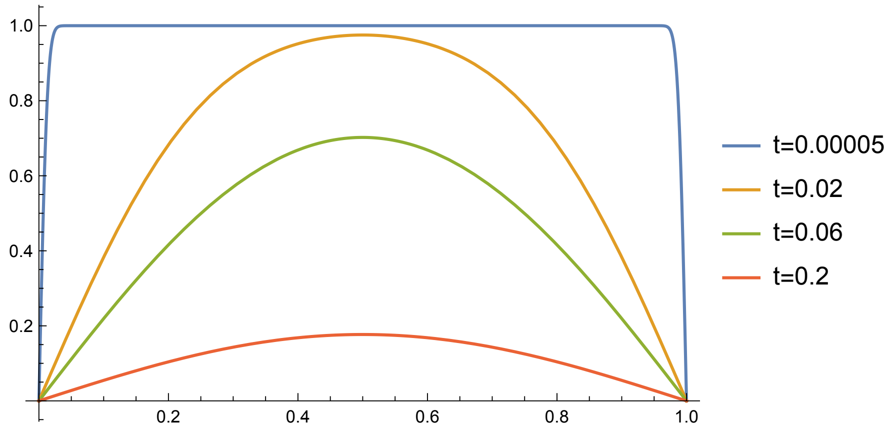
$$-\frac{2\left(-1+(-1)^{n}\right)}{n\,\pi}$$

In[37]:=
```
Clear[F]
α = 1;
F[x_, t_][NN_] := Sum[Sin[n Pi x] E^(- n^2 Pi^2 α t) c[n], {n, 1, NN}]
```

In[41]:=
```
Print["Solution for different times:"]
Quiet[
 Plot[{ F[x, 0.00005][200],  F[x, 0.02][200],  F[x, 0.06][200], F[x, 0.2][200]} ,
  {x, 0, 1}, PlotLegends → {"t=0.00005", "t=0.02", "t=0.06", "t=0.2"} ]]
```

```
Solution for different times:
```

Out[42]=



5) problema finestra scaldata da un lato

Si consideri un problema in cui consideriamo la temperatura interna a una lastra di vetro (una finestra), inizialmente a temperatura uniforme u(x,t=0) = 0. Le condizioni di bordo sono le seguenti: da un lato (x = 0) la temperatura e` mantenuta a T1 = 0. Da un altro lato la temperatura viene gradualmente aumentata con un andamento descritto dalla funzione f(t), quindi u(L, t) = f(t), con f(t) una funzione arbitraria del tempo.

In[43]:=
```
T2 = 3;
(* in questo esempio per concretezza scegliamo la condizione
 di bordo dipendente dal tempo con una andamento lineare. Basta
 modificare questa linea di codice per considerare casi diversi.*)
f[t_] = t + T2;
(* cond. bordo x=1 : U =  f[t] . cond bordo x=0: U = 0. *)
```

In[45]:=
```
(* trasliamo la soluzione di questa quantita` eplicita` *)
ub[x_, t_] = x (f[t])
```

Out[45]=
```
(3 + t) x
```

Abbiamo ridefinito u(x,t) = ub(x,t) + v(x,t). Ora, v(x,t) avra` cond. di bordo omogenee. Dobbiamo ora ridefinire condizione iniziale per v(x,t) e considerare il fatto che la PDE per v(x,t) ora diventa inomogenea.

In[46]:=
```
(* dato iniziale originale *)
u0[x_] = 0;
(* dato iniziale ridefinito per v*)
v0[x_] = u0[x] - ub[x, 0]
```
Out[47]=

$-3\,x$

Ora v soddisfa PDE: v_t - alpha v_{xx} = F0[x,t]

In[48]:=
```
(* termine forzante indotto da ridefinzione con ub *)
F0[x_, t_] = -D[ub[x, t], t]
```
Out[48]=

$-x$

In[49]:=
```
(* decomposizione termine forzante in armoniche *)
f[n_, t_] = FullSimplify[ 2 Integrate[ Sin[ n Pi x] F0[x, t], {x, 0, 1}],
  Assumptions → Element[n, Integers]]
```
Out[49]=

$$\frac{2\,(-1)^n}{n\,\pi}$$

Quindi per v[x,t] abbiamo PDE inomogenea con dato iniziale non banale. Spezziamo la soluzione in soluzione problema omogeneo con stesso dato iniziale + piu` soluzione eq. inomogenea con dato iniziale nullo.

In[50]:=
```
(* primo step:
  coefficiente armonico soluzione omogenea con dato iniziale ridefinito *)
a[n_] = FullSimplify[ 2 Integrate[ Sin[ n Pi x] v0[x], {x, 0, 1}],
  Assumptions → Element[n, Integers]]
```
Out[50]=

$$\frac{6\,(-1)^n}{n\,\pi}$$

In[52]:=
```
(* secondo step: per risolvere il problema inomogeneo
   usiamo metodo f. di Green . Il coefficente armonico della
   soluzione inomogenea con dato iniziale nullo e` dato da*)
Grn[n_, t_] = Integrate[E^(-α (n Pi)^2 (t - s)) f[n, s], {s, 0, t}] // FullSimplify
```
Out[52]=

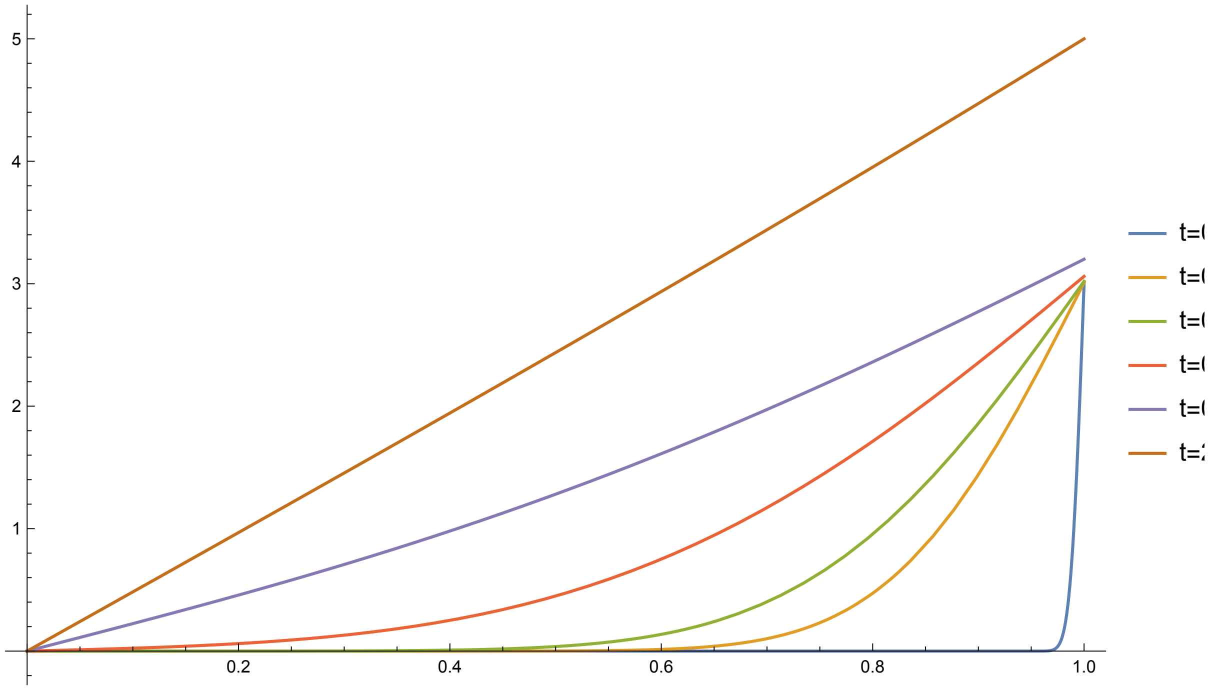
$$\frac{2\,(-1)^n\left(1-e^{-n^2\pi^2 t}\right)}{n^3\,\pi^3}$$

In[53]:=
```
α = 1;
(* soluzione completa: *)
U[x_, t_][NN_] :=
 ub[x, t] + Sum[ Sin[n Pi x] ( E^(- n^2 Pi^2 α t) a[n] + Grn[n, t]), {n, 1, NN, 1}]
```

In[55]:=
```
Print["Solution for different times:"]
Quiet[Plot[{ U[x, 0.00005][200], U[x, 0.01][200], U[x, 0.02][200],
   U[x, 0.06][200], U[x, 0.2][200], U[x, 2][200]} , {x, 0, 1},
  PlotLegends → {"t=0.00005", "t=0.01", "t=0.02", "t=0.06", "t=0.2", "t=2"} ]]
```

Solution for different times:

Out[56]=



6) problema diffusione

Consider a drop of a substance diffusing in a hypothetical one-dimensional medium confined to an interval [0,L]. The initial condition is given by a Dirac delta. The boundary conditions are Neumann boundary conditions, since there is no diffusion through the boundaries.

In[73]:=
```
(* Compute Fourier coefficients *)
x0 = 1 / 3;
(* questi sono coeff. Fourier di una delta di Dirac nel punto x0,
espansi nelle armoniche appropriate per un problema di Neumann *)
c[n_] = 2 Cos[ n Pi x0]
```

Out[74]=

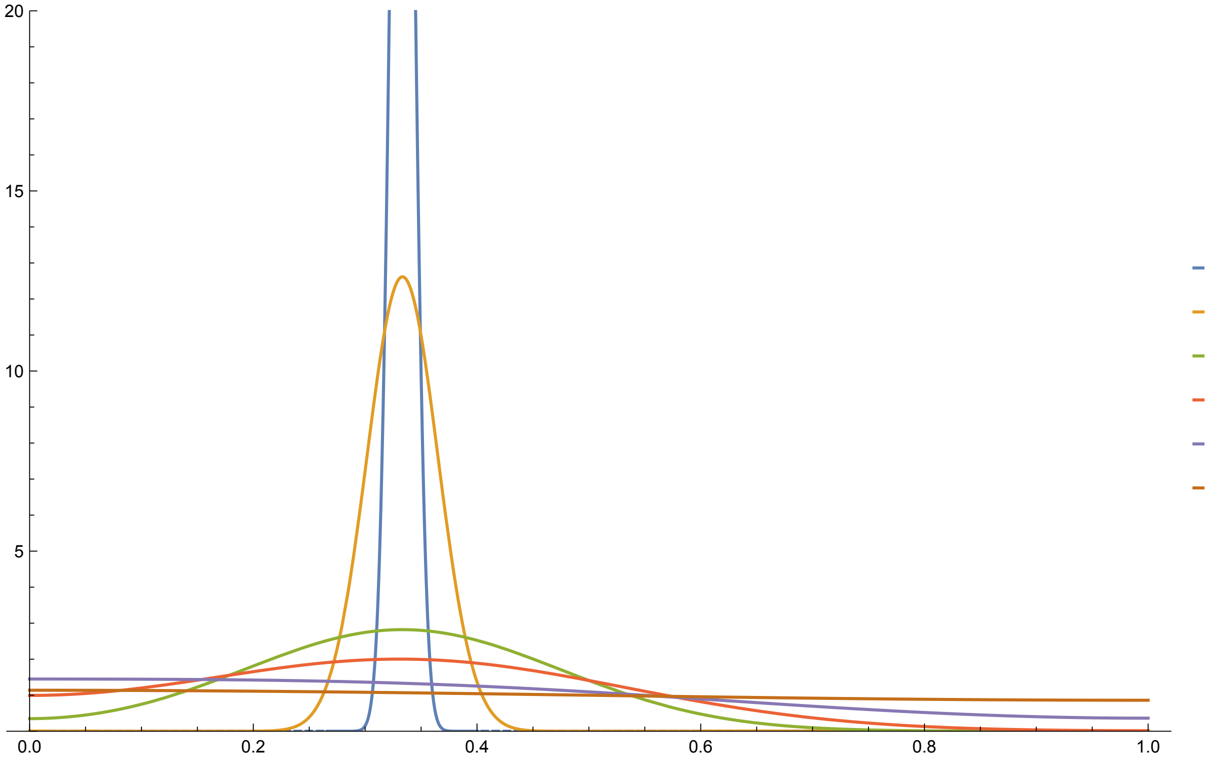
$$2\, \mathrm{Cos}\left[\frac{n\,\pi}{3}\right]$$

In[75]:=
```
α = 1;
U[x_, t_][NN_] :=
 1 / 2 c[0] + Sum[ Cos[n Pi x] E^(- n^2 Pi^2 α t) c[n], {n, 1, NN, 1}]
```

In[81]:=
```
Print["Solution for different times:"]
Quiet[Plot[{ U[x, 0.00005][200], U[x, 0.0005][200],  U[x, 0.01][200],
   U[x, 0.02][200],  U[x, 0.06][200], U[x, 0.2][200]} , {x, 0, 1},
  PlotLegends → {"t=0.00005", "t=0005", "t=0.01", "t=0.02", "t=0.06", "t=0.2"} ,
  PlotRange → {0, 20}]]
```

```
Solution for different times:
```

Out[82]=



7) Problema sferico: raffreddamento di sfera 3D di raggio R.

In[ ]:=
```
(* prima armonica sferica *)
BesselJ[1 / 2, r] / Sqrt[r]
```

Out[ ]=

$$\frac{\sqrt{\frac{2}{\pi}}\ \mathrm{Sin}[r]}{r}$$

In[63]:=
```
c[n_] = 2 / R Integrate[ Sin[ n Pi r / R] r , {r, 0, R}] //
  Simplify[#,  Assumptions → Element[n, Integers]] &
```

Out[63]=

$$-\frac{2\,(-1)^{n}\,R}{n\,\pi}$$

In[64]:=
```
R = 1; α = 1;
```

In[65]:=
```
FS[r_, t_][NN_] := Sum[ c[n]  Sin[ n Pi r / R] / r E^(-α (n Pi / R)^2 t) , {n, 1, NN}]
```

In[66]:=
```
Quiet[ Plot[
  { FS[x, 0.0001][400],  FS[x, 0.02][200],  FS[x, 0.06][200], FS[x, 0.2][200]} ,
  {x, 0, 1}, PlotLegends → {"t=0.0001", "t=0.02", "t=0.06", "t=0.2"} ]]
```

Out[66]=