```
> restart:
with(plots):




#
Theta[0]:=0:
Theta[1]:=Pi/3:
Theta[2]:=-Pi/3:
Theta[3]:=0:
for i from 0 to 3 do
s[i]:=1/3:
T[i]:=matrix(2,2,[s[i]*cos(Theta[i]),-s[i]*sin(Theta[i]),s[i]*sin
(Theta[i]),s[i]*cos(Theta[i])]):

end do;
v0:=matrix(2,1,[1,0]);



for j from 0 to 3 do
A[j]:=evalf(evalm(add(evalm(T[i]&*v0),i=0..j))[1,1]);
B[j]:=evalf(evalm(add(evalm(T[i]&*v0),i=0..j))[2,1]);
C[j]:=evalf(s[j]*cos(Theta[j])):
E[j]:=evalf(s[j]*sin(Theta[j])):
end do:
A[-1]:=0:
B[-1]:=0:
C[-1]:=0:
E[-1]:=0:

Koxa:= proc(t)
local zz,i,eq1,eq2,j,jj,TT:

i:=0:


j:=1:
zz[1]:=0:
while zz[j]<1+t do
eq1[j],eq2[j],TT[j]:=KoxaUrav(i,zz[j]):
zz[j+1]:=zz[j]+t:
j:=j+1:
end do:


subs(x[0.]=0,y[0.]=0,solve({seq(eq1[jj],jj=1..j-1),seq(eq2[jj],
jj=1..j-1)},{seq(x[zz[jj]],jj=1..j-1),seq(y[zz[jj]],jj=1..j-1)}))
;

subs(%,[seq([x[zz[jj]],y[zz[jj]]],jj=1..j-1)]);
end proc:

KoxaUrav := proc(n,t)
local eq1,eq2,i,T,qq,k:
```

```
T:=t:
k:=4:

#for i from 1 by 1 to n-1 do
#eq1[i]:=A[trunc(k*T)-1]+C[trunc(k*T)]*eq1[i-1]-E[trunc(k*T)]*eq2
[i-1];
#eq2[i]:=B[trunc(k*T)-1]+E[trunc(k*T)]*eq1[i-1]+C[trunc(k*T)]*eq2
[i-1];
#T:=k*T-trunc(k*T):
#end do:
qq:=k*T-trunc(k*T):


#eq1[n]:=x[t]=subs(eq1[0]=A[trunc(k*T)-1]+C[trunc(k*T)]*x[qq]-E
[trunc(k*T)]*y[qq],eq2[0]=B[trunc(k*T)-1]+E[trunc(k*T)]*x[qq]+C
[trunc(k*T)]*y[qq],eq1[n-1]):
#eq2[n]:=y[t]=subs(eq1[0]=A[trunc(k*T)-1]+C[trunc(k*T)]*x[qq]-E
[trunc(k*T)]*y[qq],eq2[0]=B[trunc(k*T)-1]+E[trunc(k*T)]*x[qq]+C
[trunc(k*T)]*y[qq],eq2[n-1]):
#print([eq1[n],eq2[n]]);

if n=0 then
eq1[0]:=x[t]=A[trunc(k*T)-1]+C[trunc(k*T)]*x[qq]-E[trunc(k*T)]*y
[qq]:
eq2[0]:=y[t]=B[trunc(k*T)-1]+E[trunc(k*T)]*x[qq]+C[trunc(k*T)]*y
[qq]:
fi:

eq1[n],eq2[n],T;
end proc:
```

$$s_0 := \frac{1}{3}$$

$$T_0 := \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{bmatrix}$$

$$s_1 := \frac{1}{3}$$

$$T_1 := \begin{bmatrix} \frac{1}{6} & -\frac{1}{6}\sqrt{3} \\ \frac{1}{6}\sqrt{3} & \frac{1}{6} \end{bmatrix}$$

$$s_2 := \frac{1}{3}$$

$$T_2 := \begin{bmatrix} \frac{1}{6} & \frac{1}{6}\sqrt{3} \\ -\frac{1}{6}\sqrt{3} & \frac{1}{6} \end{bmatrix}$$

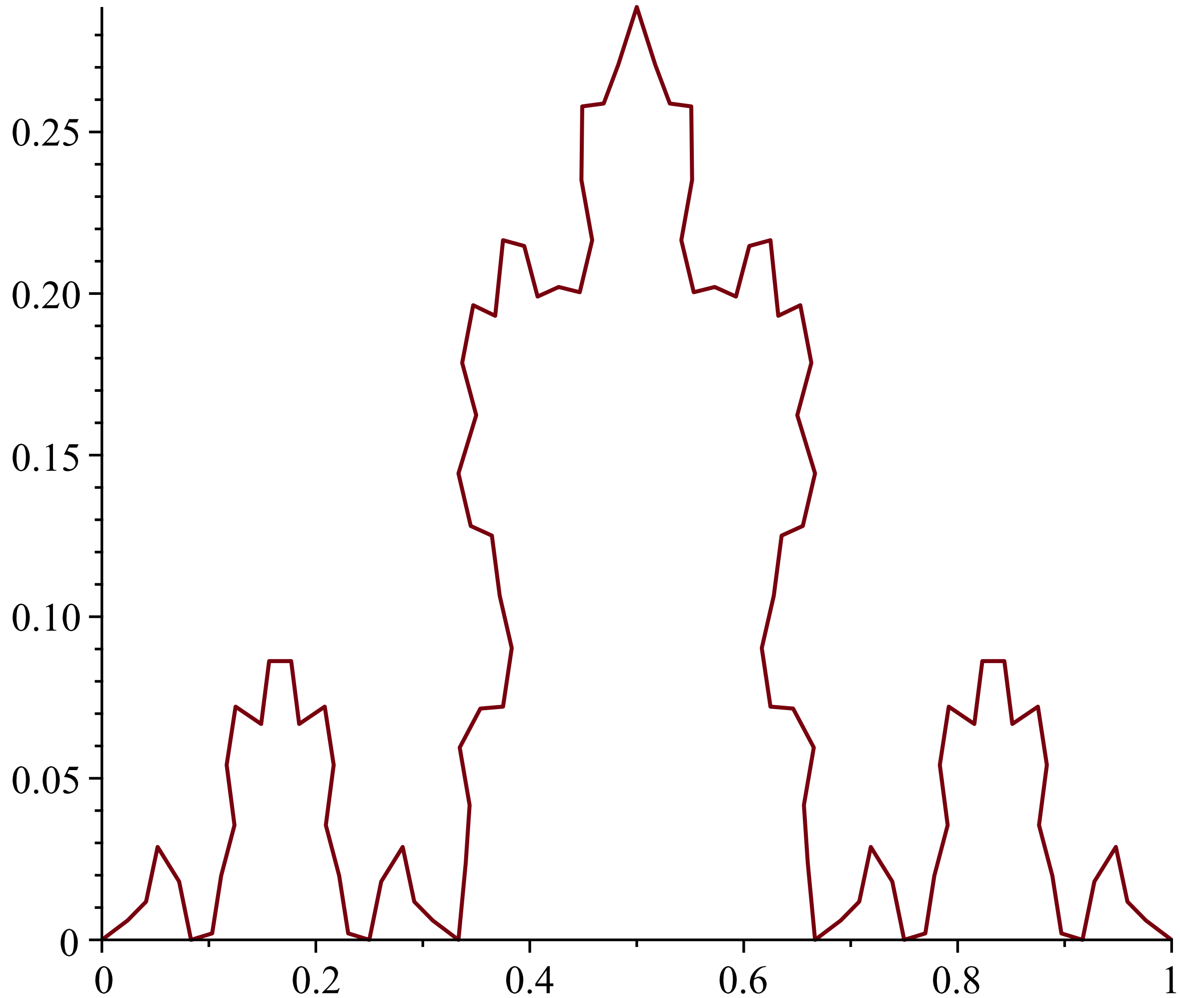
$$s_3 := \frac{1}{3}$$

$$T_3 := \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{3} \end{bmatrix}$$

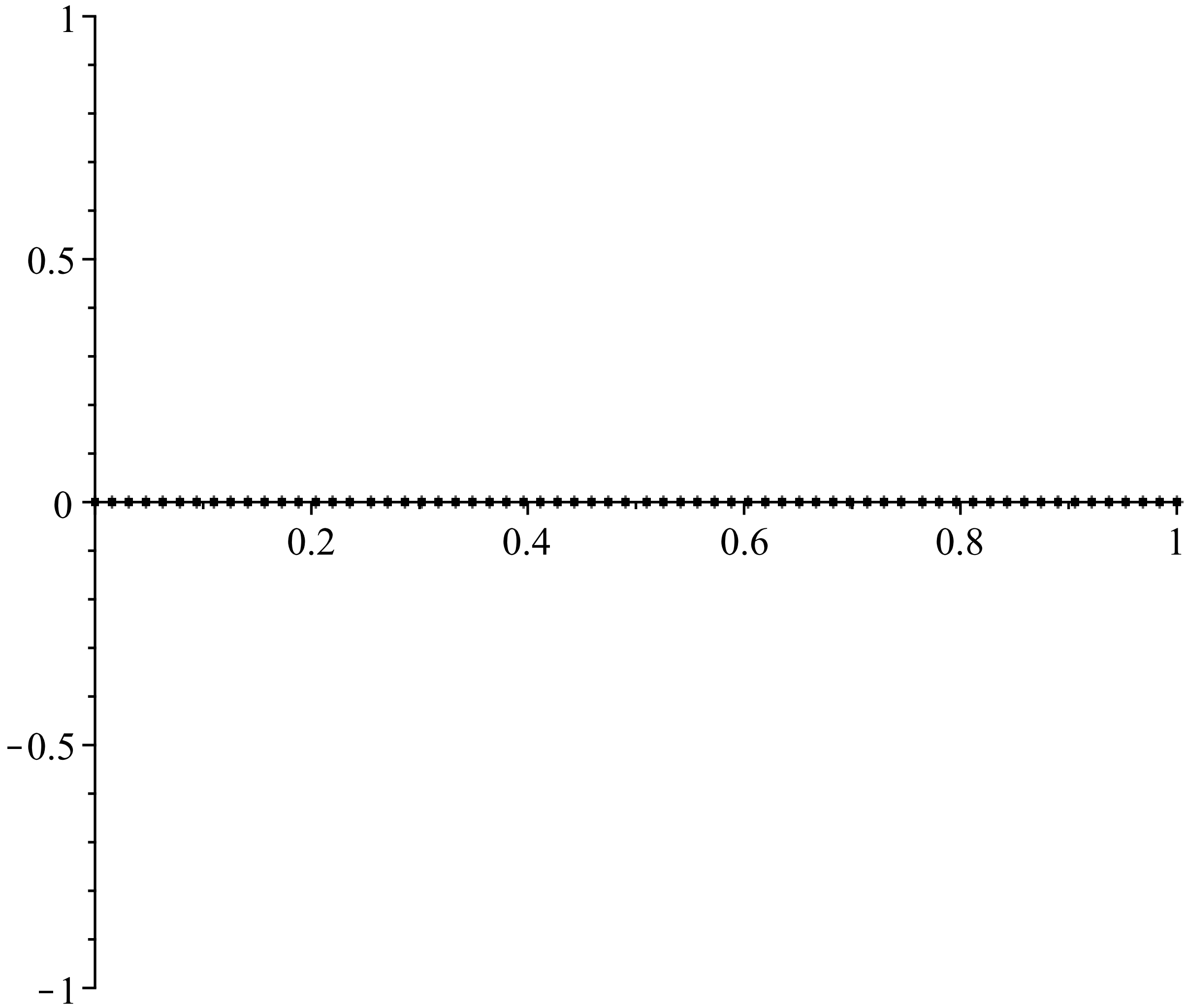
$$v0 := \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{1}$$

```
> Koxa(0.1);
  plot(Koxa(0.01));
```

[[0., 0.], [0.1250000000, 0.07216878368], [0.2500000000, 0.], [0.3750000000, 0.07216878367], [0.3749999999, 0.2165063511], [0.5000000000, 0.2886751347], [0.6250000001, 0.2165063511], [0.6250000000, 0.07216878367], [0.7500000000, 0.], [0.8750000000, 0.07216878368], [1., 0]]

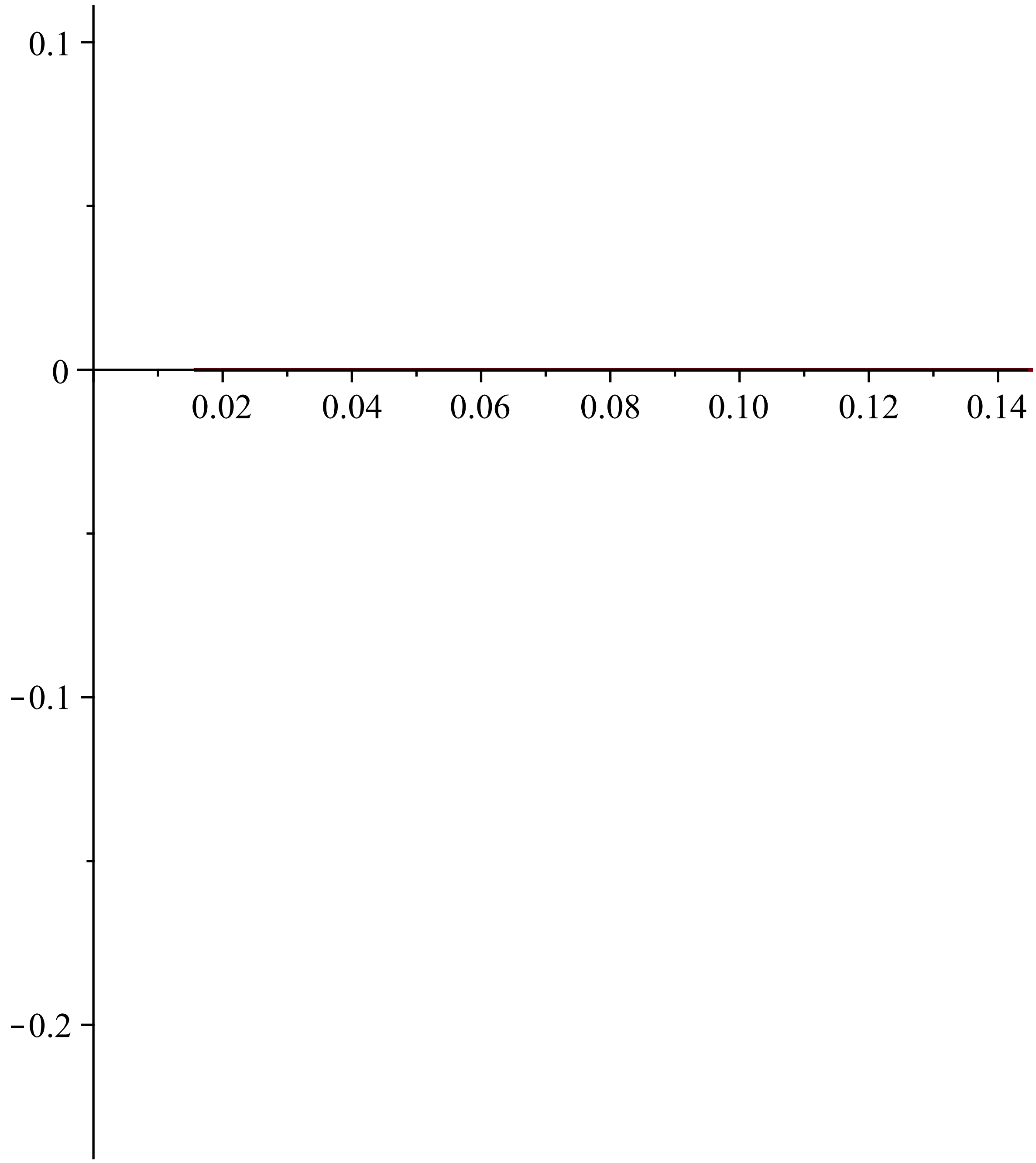



```
> with(plots):

  pointplot([seq([Koxa(i/1000)[1],Koxa(i/1000)[2]],i=1..999)],
  symbol = diamond,symbolsize=1);
  pointplot([seq(Koxa(i/10),i=1..9)],symbol = diamond,symbolsize=1)
  ;

  #animate(plot, [[Koxa(tt)[1],Koxa(tt)[2], tt = 0.1 .. A]], A =
  0.2 .. 0.9, scaling = constrained, frames = 10);
```



Error, (in plots:-pointplot) incorrect number of coordinates in points data

```
> plot([Koxa(x)[1],Koxa(x)[2],x=0.1..0.8]);
```

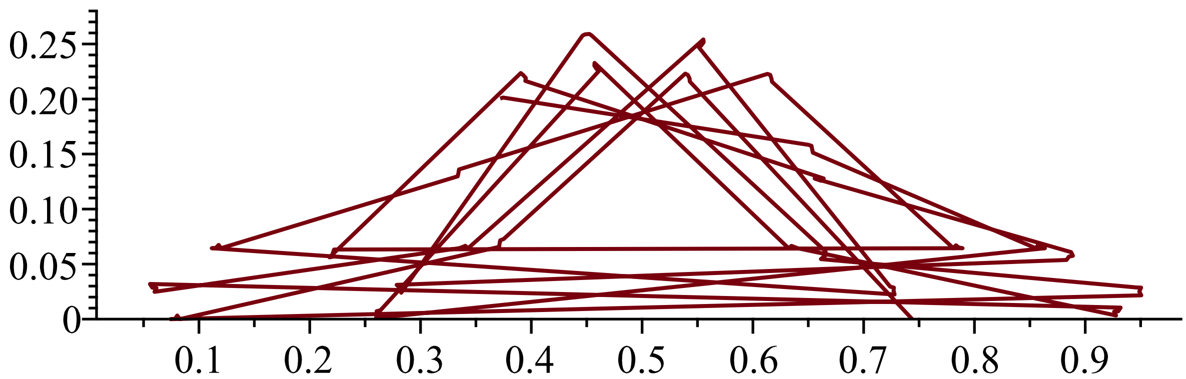



```
> seq(sqrt((Koxa((2*k+1)/1000)[1]-Koxa(2*k/1000)[1])^2+(Koxa((2*k+1)/1000)[2]-Koxa(2*k/1000)[2])^2),k=1..98);
```

[0.3441215323, 0.04232885065], [0.7692307693, 0.], [0.1130867710, 0.003695698736], [0.6294452348, 0.2081910289], [0.1728307254, 0.09239246843], [0.3705547653, 0.2081910290], [0.8869132289, 0.003695698739], [0.2307692307, 0.], [0.6558784677, 0.04232885067] **(2)**

```
> animate(plot, [[Koxa(tt)[1],Koxa(tt)[2], tt = 0.1 .. CCC]], CCC = 0.2 .. 0.9, scaling = constrained, frames = 50);
```

$CCC = 0.2$



> $Koxa(0.1);$

[[0., 0.], [0.1250000000, 0.07216878368], [0.2500000000, 0.], [0.3750000000, 0.07216878367], [0.3749999999, 0.2165063511], [0.5000000000, 0.2886751347], [0.6250000001, 0.2165063511], [0.6250000000, 0.07216878367], [0.7500000000, 0.], [0.8750000000, 0.07216878368], [1., 0]] **(3)**

> A

A **(4)**

>