

```

> restart;

> #convert x to base three expansion
ftob3:=proc(x)
  local i,a,z;
  z:=x;
  for i from 1 to 32 do
    z:=z*3;
    a[i]:=floor(z);
    z:=z-a[i];
  od;
  a;
end;

      ftob3 := proc(x)
      local i, a, z;
      z := x;
      for i to 32 do z := 3*z; a[i] := floor(z); z := z - a[i]; end do;
      a;
      end proc;

> #convert base two expansion to z
b2tof:=proc(b)
  local i,z;
  z:=0.0;
  for i from 32 by -1 to 1 do
    z:=(b[i]+z)/2;
  od;
  z;
end;

      b2tof := proc(b)
      local i, z;
      z := 0.;
      for i from 32 by -1 to 1 do z := 1/2*b[i] + 1/2*z end do;
      z;
      end proc;

> #compute the lebesgue singular function
lebesgue:=proc(x)
  local a,b,i,j;
  if x>=1 then return 1
  elif x<=0 then return 0 fi;
  a:=ftob3(x);
  for i from 1 to 32 do
    if(a[i]=2) then b[i]:=1
    elif(a[i]=0) then b[i]:=0
    else
      b[i]:=1;
      for j from i+1 to 32 do
        b[j]:=0;

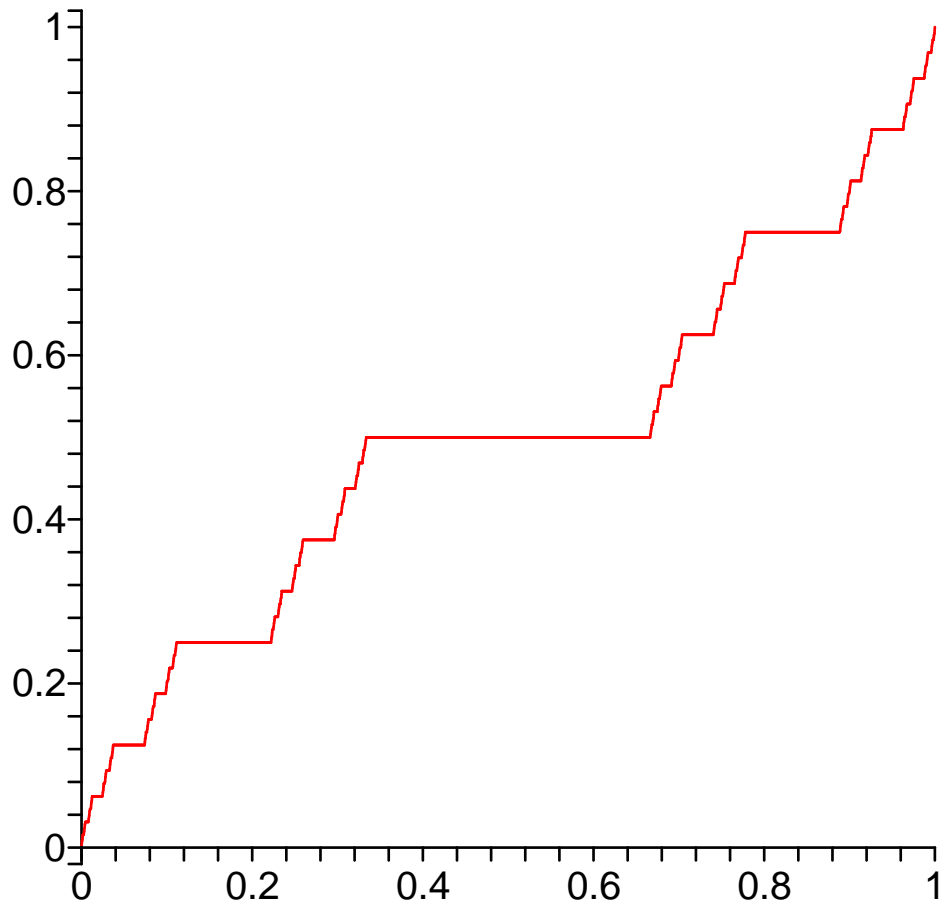
```

```
        od;  
        i:=32;  
    end  
od;  
b2tof(b);  
end;
```

>

```
lebesgue := proc(x)  
local a, b, i, j;  
if 1 <= x then return 1 elif x <= 0 then return 0 end if;  
a := ftob3(x);  
for i to 32 do if a[i] = 2 then  
    b[i] := 1  
    elif a[i] = 0 then  
    b[i] := 0  
    else  
    b[i] := 1;  
    for j from i + 1 to 32 do b[j] := 0 end do;  
    i := 32;  
    end if;  
end do;  
b2tof(b);  
end proc;
```

> plot(lebesgue, 0..1, numpoints=2000);



>