

```
function [v] = velocity(d,vektor)
load('Projekt_initial_data.mat')

if vektor*vektor==anna_speed_kmph'*anna_speed_kmph %Sätter den att räkna för anna

    %Felmedelanden ang. gränsvärden
    if d < min(anna_distance_km)
        error('The input distance is smaller then the smalest value in the vector')
    elseif d > max(anna_distance_km)
        error('The input distance is bigger then the biggest value in the vector')
    else
        distance=anna_distance_km;
    end
elseif vektor==elsa_speed_kmph %Sätter den att räkna för elsa

    %Felmedelanden ang. gränsvärden
    if d < min(elsa_distance_km)
        error('The input distance is smaller then the smalest value in the vector')
    elseif d > max(elsa_distance_km)
        error('The input distance is bigger then the biggest value in the vector')
    else
        distance=elsa_distance_km;
    end
else error('fel vektor')
end

%Hittar vilken position d har i speed-vektorn genom interpolation mellan punkterna
%större och lägre än d.
ta1=findpos(distance,d)-1; %hittar efter vilken positon stäcken kördes
ta2=findpos(distance,d); %hittar innan vilken positon stäcken kördes
x=[ta1 ta2];
x1=[distance(ta1) distance(ta2)];
a=[ones(2,1) x1'];
b=x';
p1=a\b;

pos_d=polyval(flip(p1),d); %pos_d är d:s position i distance

%Hittar de hastighetsvärdena som ligger i punkten innan och efter v
s1=vektor(ta1); %hittar vilken hastighet vid tiden ta1
s2=vektor(ta2); %hittar vilken hastighet vid tiden ta2
y=[s1 s2];

%Hittar vilken hastighet hon hade vid position hämtat från distans
a2=[ones(2,1) x'];
b2=y';
p=a2\b2;
p=p';
p=flip(p);
v=polyval(p,pos_d);

%Om man vill plotta för att se att det blivit rätt
% plot(x,y,'*',pos_d,v,'ro',x,polyval(p,x))
% ylabel('kmph')
```

```
% xlabel('position')  
% grid on  
  
end
```