

Model: Stopping Distance

```
> # Working with procedures
```

```
> sumUpTon:=proc(n)
```

```
  local s,i;
```

```
  s:=0;
```

```
  for i from 1 to n do
```

```
    s:=s+i^2;
```

```
  od;
```

```
  s;
```

```
end;
```

```
sumUpTon := proc(n) local s, i; s := 0; for i to n do s := s + i^2 end do; s end proc (1)
```

```
> sumUpTon(1003);
```

```
336845514 (2)
```

```
> restart;
```

```
> v:=[20,25,30,35,40,45,50,55,60,65,70,75,80];
```

```
dr:=[22,28,33,39,44,50,55,61,66,72,77,83,88];
```

```
db:=[20,28,40.5,52.5,72,92.5,118,148.5,182,220.5,266,318,376];
```

```
d:=[42,56,73.5,91.5,116,142.5,173,209.5,248,292.5,343,401,464];
```

```
v := [20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80]
```

```
dr := [22, 28, 33, 39, 44, 50, 55, 61, 66, 72, 77, 83, 88]
```

```
db := [20, 28, 40.5, 52.5, 72, 92.5, 118, 148.5, 182, 220.5, 266, 318, 376]
```

```
d := [42, 56, 73.5, 91.5, 116, 142.5, 173, 209.5, 248, 292.5, 343, 401, 464] (3)
```

	A	B	C	D	E
2	20	22	20	42	
3	25	28	28	56	
4	30	33	40.5	73.5	
5	35	39	52.5	91.5	
6	40	44	72	116	
7	45	50	92.5	142.5	
8	50	55	118	173	
9	55	61	148.5	209.5	
10	60	66	182	248	
11	65	72	220.5	292.5	
12	70	77	266	343	
13	75	83	318	401	
14	80	88	376	464	
15					

```

> makePoints:=proc(xlist,ylist)
  local p,i;
  p:=NULL;
  for i to nops(xlist) do
    p:=p,[xlist[i],ylist[i]]:
  od:
  [p];
end;

```

makePoints := **proc**(*xlist*,*ylist*) (4)

local *p*, *i*;

p := NULL; **for** *i* **to** nops(*xlist*) **do** *p* := *p*, [*xlist*[*i*],*ylist*[*i*]] **end do**; [*p*]

end proc

```

> functionToList:=proc(lst,f)
  local p,i;
  p:=NULL;
  for i to nops(lst) do
    p:=p,f(lst[i]):
  od:
  [p];
end;

```

functionToList := **proc**(*lst*,*f*) (5)

local *p*, *i*;

p := NULL; **for** *i* **to** nops(*lst*) **do** *p* := *p*,*f*(*lst*[*i*]) **end do**; [*p*]

end proc

> f:=x->x^2;

$$f:=x\rightarrow x^2$$

(6)

> functionToList(v,f);

[400, 625, 900, 1225, 1600, 2025, 2500, 3025, 3600, 4225, 4900, 5625, 6400]

(7)

> drv:=makePoints(v,dr);

drv := [[20, 22], [25, 28], [30, 33], [35, 39], [40, 44], [45, 50], [50, 55], [55, 61], [60, 66],
[65, 72], [70, 77], [75, 83], [80, 88]]

(8)

> f:=x->x^2;

functionToList(v,f);

dbv:=makePoints(%,db);

$$f:=x\rightarrow x^2$$

[400, 625, 900, 1225, 1600, 2025, 2500, 3025, 3600, 4225, 4900, 5625, 6400]

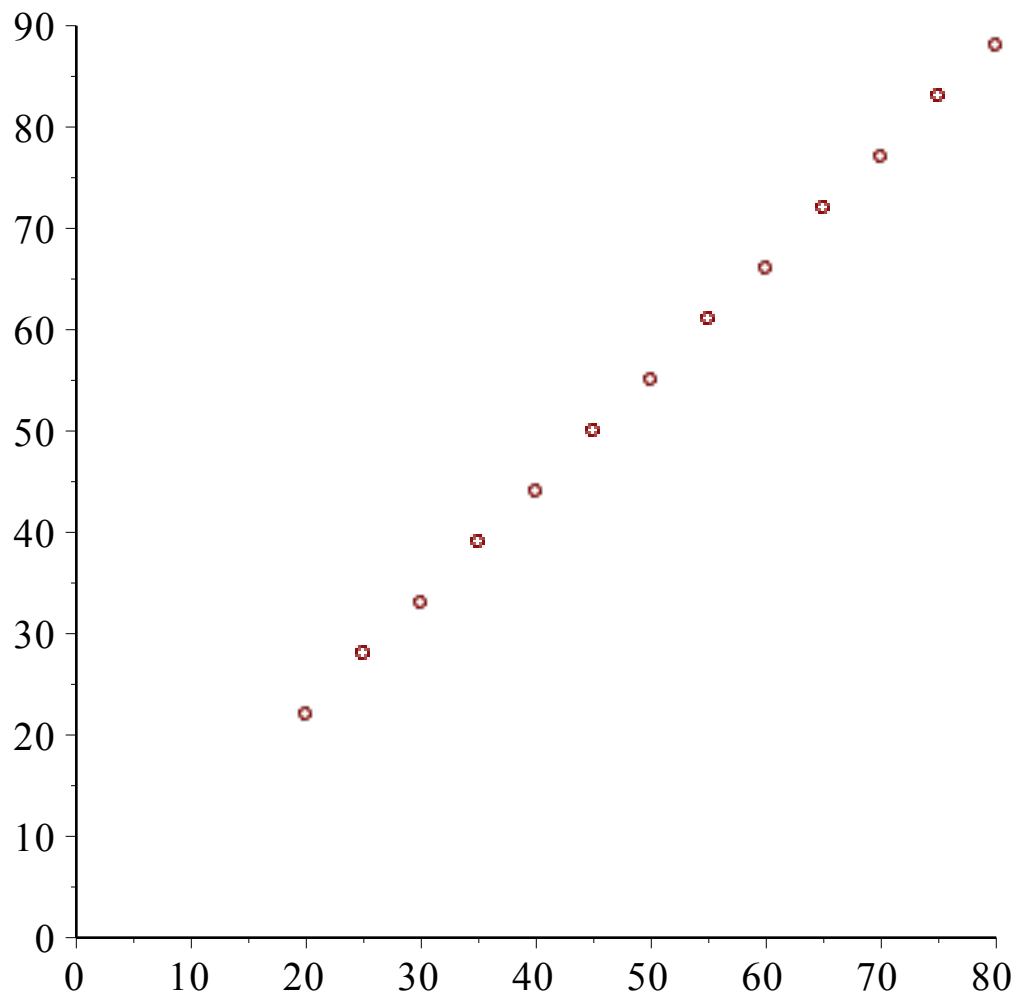
dbv := [[400, 20], [625, 28], [900, 40.5], [1225, 52.5], [1600, 72], [2025, 92.5], [2500,
118], [3025, 148.5], [3600, 182], [4225, 220.5], [4900, 266], [5625, 318], [6400, 376]]

(9)

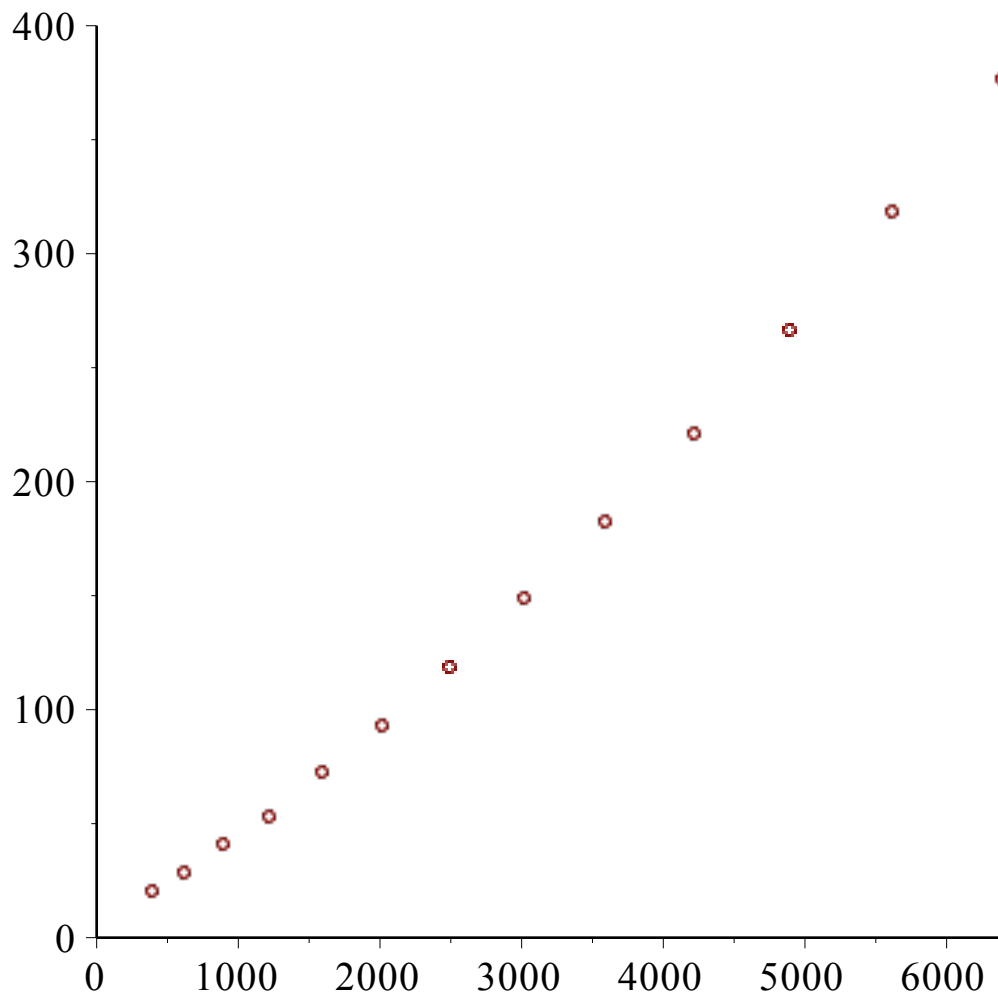
> with(plots):

p1:=plot(drv,0..80,0..90,style=point,symbol=circle):

display(p1);



```
> p2:=plot(dbv,0..6400,0..400,style=point,symbol=circle):  
display(p2);
```



```

> plotPropModel:=proc(p,i)
  local slpe,liner,xmax,j,p1,p2:
  slpe:=(p[i][2]-0)/(p[i][1]-0):
  print(`Slope = `,evalf(slpe));
  liner:=t->slpe*t:
  xmax:=-infinity:
  for j to nops(p) do
    if p[j][1] > xmax then
      xmax:=p[j][1]:
    fi:
  od:
  p1:=plot(p,style=point,symbol=circle):
  p2:=plot(liner,0..xmax,color=navy):
  display({p1,p2});
end;

```

plotPropModel := **proc**(*p*, *i*)

(10)

local *slpe*, *liner*, *xmax*, *j*, *p1*, *p2*;

slpe := *p*[*i*][2]/*p*[*i*][1];

 print(*Slope* = , evalf(*slpe*));

```
liner := t → slope * t;
```

```
xmax := - ∞;
```

```
for j to nops(p) do if xmax < p[j][1] then xmax := p[j][1] end if end do;
```

```
p1 := plot(p, style = point, symbol = circle);
```

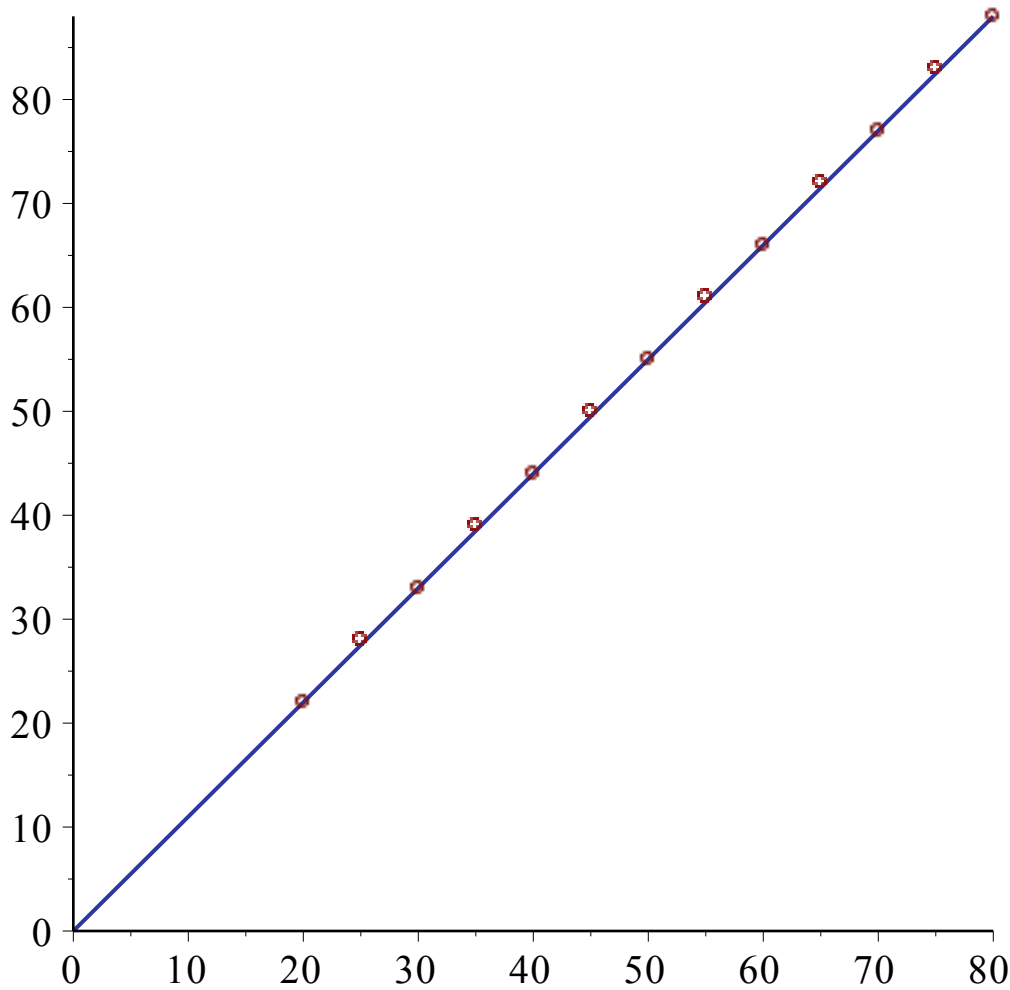
```
p2 := plot(liner, 0 .. xmax, color = navy);
```

```
plots:-display( {p1, p2} )
```

```
end proc
```

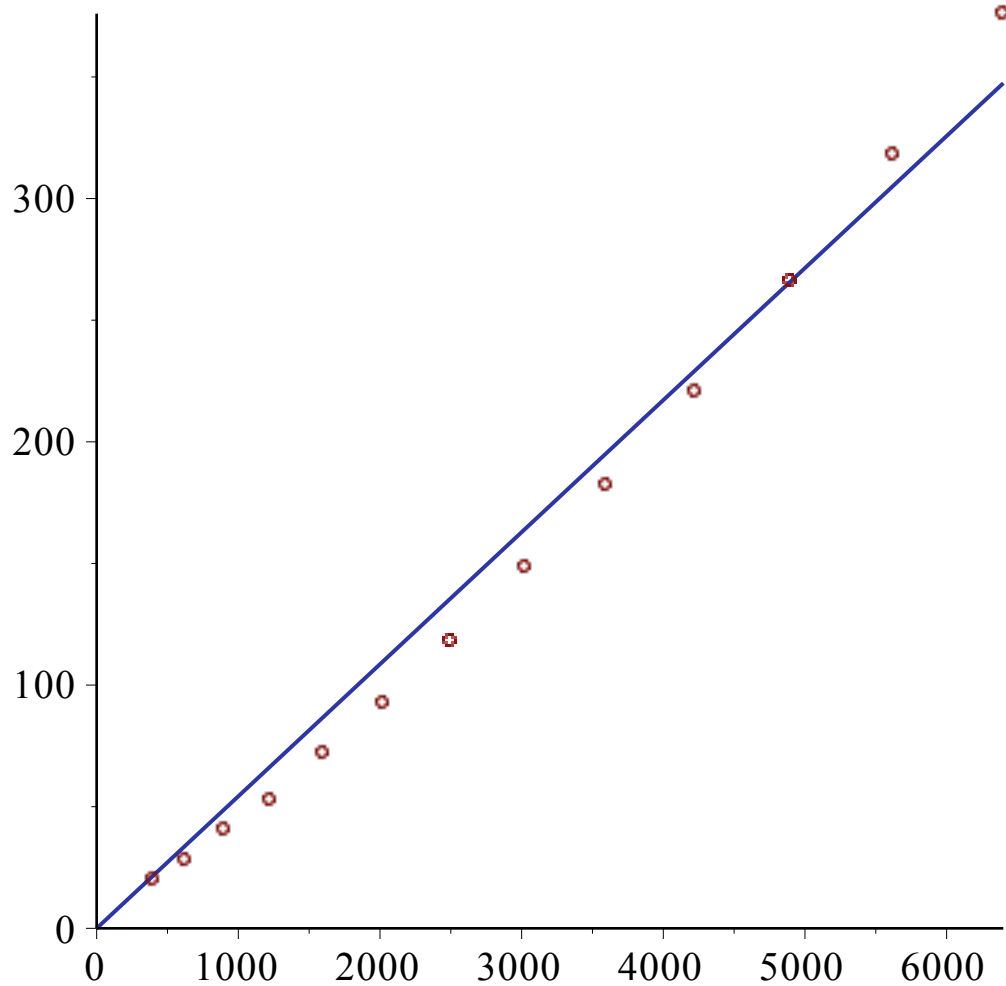
```
> plotPropModel(drv, 5);
```

Slope = , 1.100000000



```
> plotPropModel(dbv, 11);
```

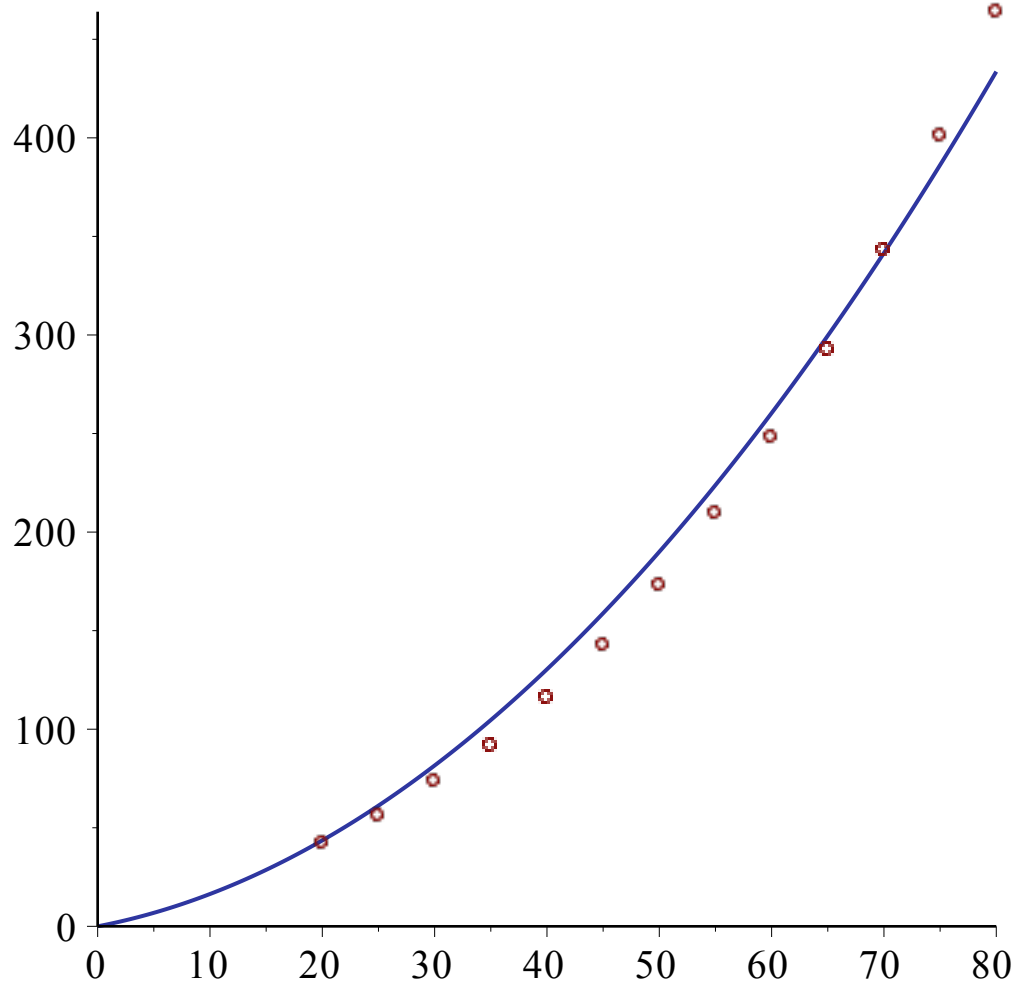
Slope = , 0.05428571429



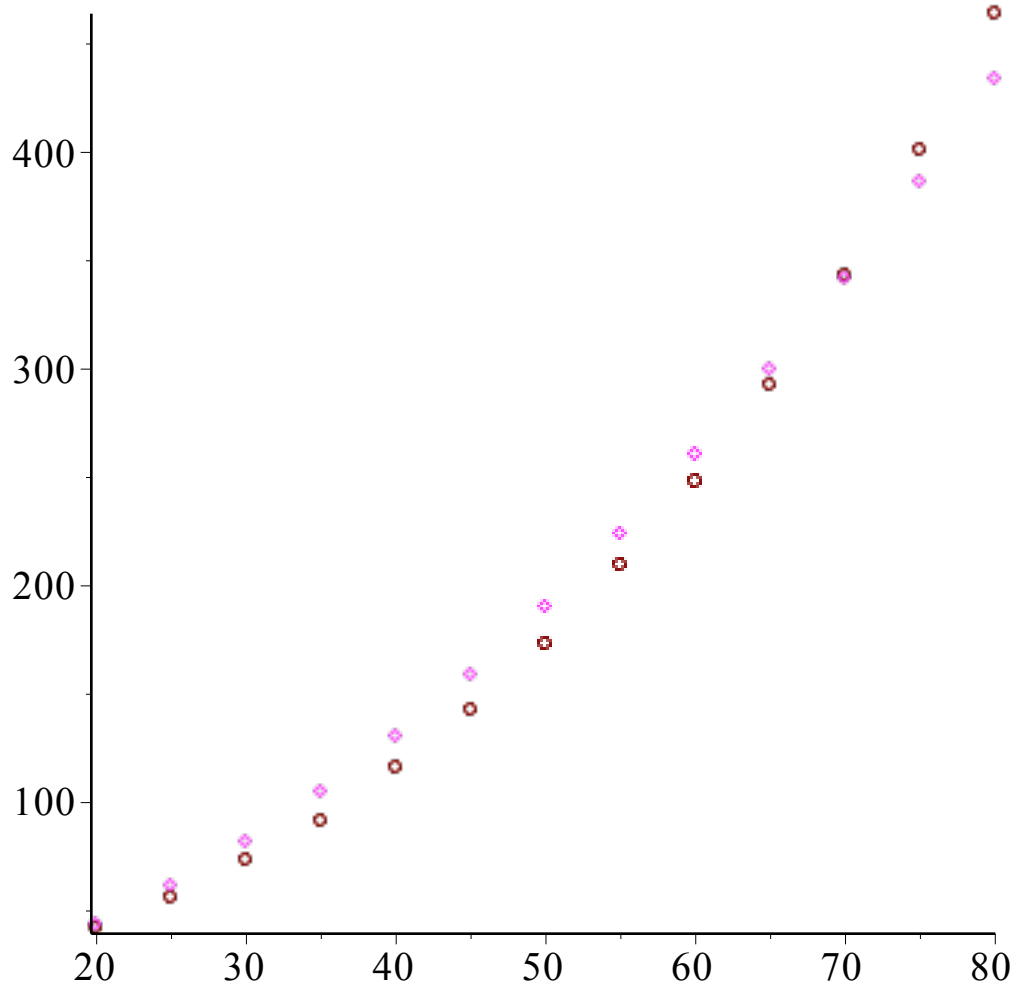
```
> dv:=makePoints(v,d);
q1:=plot(dv,style=point,symbol=circle):
mdl:=v->1.1*v+0.054*v^2;
q2:=plot(mdl,0..80,colour=navy):
display({q1,q2});
```

```
dv := [[20, 42], [25, 56], [30, 73.5], [35, 91.5], [40, 116], [45, 142.5], [50, 173], [55,
209.5], [60, 248], [65, 292.5], [70, 343], [75, 401], [80, 464]]
```

```
mdl := v → 1.1 v + 0.054 v2
```



```
> p:=NULL:
  for i to nops(v) do
    p:=p,[v[i],mdl(v[i])]:
  od:
q3:=plot([p],style=point,symbol=diamond,colour=magenta):
display({q1,q3});
```



```
> p;
[20, 43.600], [25, 61.250], [30, 81.600], [35, 104.650], [40, 130.400], [45, 158.850], [50,
190.000], [55, 223.850], [60, 260.400], [65, 299.650], [70, 341.600], [75, 386.250], [80,
433.600]
```

```
> evalf( $\frac{44 \cdot 3600}{5280 \cdot 20}$ );
1.500000000
```

```
> evalf( $\frac{61.25 \cdot 3600}{5280 \cdot 25}$ );
1.670454545
```

```
> convertToSeconds:=proc(d,v)
evalf(d*3600/(5280*v));
end;
convertToSeconds := proc(d, v) evalf(15/22 * d/v) end proc
```

```
> convertToSeconds(61.25,25);
1.670454545
```

```
> p:=[p]:  
  for i from 1 to nops(p) do  
    v:=p[i][1]:  
    d:=p[i][2]:  
    print(v,convertToSeconds(d,v));  
  od:  
20, 1.486363636  
25, 1.670454545  
30, 1.854545455  
35, 2.038636364  
40, 2.222727273  
45, 2.406818182  
50, 2.590909091  
55, 2.775000000  
60, 2.959090909  
65, 3.143181818  
70, 3.327272727  
75, 3.511363636  
80, 3.695454545
```

(16)

```
>
```