

Aufgabenblatt 4 (Bresenham-Algorithmus)

```
> restart;  
> with(plots):  
Warning, the name changecoords has been redefined
```

- Variante mit Runden

Zunächst die Anfangs- und Endkoordinaten definieren:

```
> x0 := 0;  
y0 := 0;  
x1 := 9;  
y1 := 7;  
  
x0 := 0  
y0 := 0  
x1 := 9  
y1 := 7
```

Danach Steigung und y-Abschnitt:

```
> dx := x1-x0;  
dy := y1-y0;  
m := dy/dx;  
b := y0 - m*x0;  
  
dx := 9  
dy := 7  
m :=  $\frac{7}{9}$   
b := 0
```

Und dann alle Punkte streng nach Rechenvorschrift ermitteln:

```
> pxl := [];  
> for x from x0 to x1 do  
  y := round(m*x + b);  
  pxl := [op(pxl), [x,y]]  
end do;  
  
pxl := [ ]  
y := 0  
pxl := [[0, 0]]  
y := 1  
pxl := [[0, 0], [1, 1]]  
y := 2  
pxl := [[0, 0], [1, 1], [2, 2]]  
y := 2  
pxl := [[0, 0], [1, 1], [2, 2], [3, 2]]
```

```

y := 3
pxl := [[0, 0], [1, 1], [2, 2], [3, 2], [4, 3]]
y := 4
pxl := [[0, 0], [1, 1], [2, 2], [3, 2], [4, 3], [5, 4]]
y := 5
pxl := [[0, 0], [1, 1], [2, 2], [3, 2], [4, 3], [5, 4], [6, 5]]
y := 5
pxl := [[0, 0], [1, 1], [2, 2], [3, 2], [4, 3], [5, 4], [6, 5], [7, 5]]
y := 6
pxl := [[0, 0], [1, 1], [2, 2], [3, 2], [4, 3], [5, 4], [6, 5], [7, 5], [8, 6]]
y := 7
pxl := [[0, 0], [1, 1], [2, 2], [3, 2], [4, 3], [5, 4], [6, 5], [7, 5], [8, 6], [9, 7]]

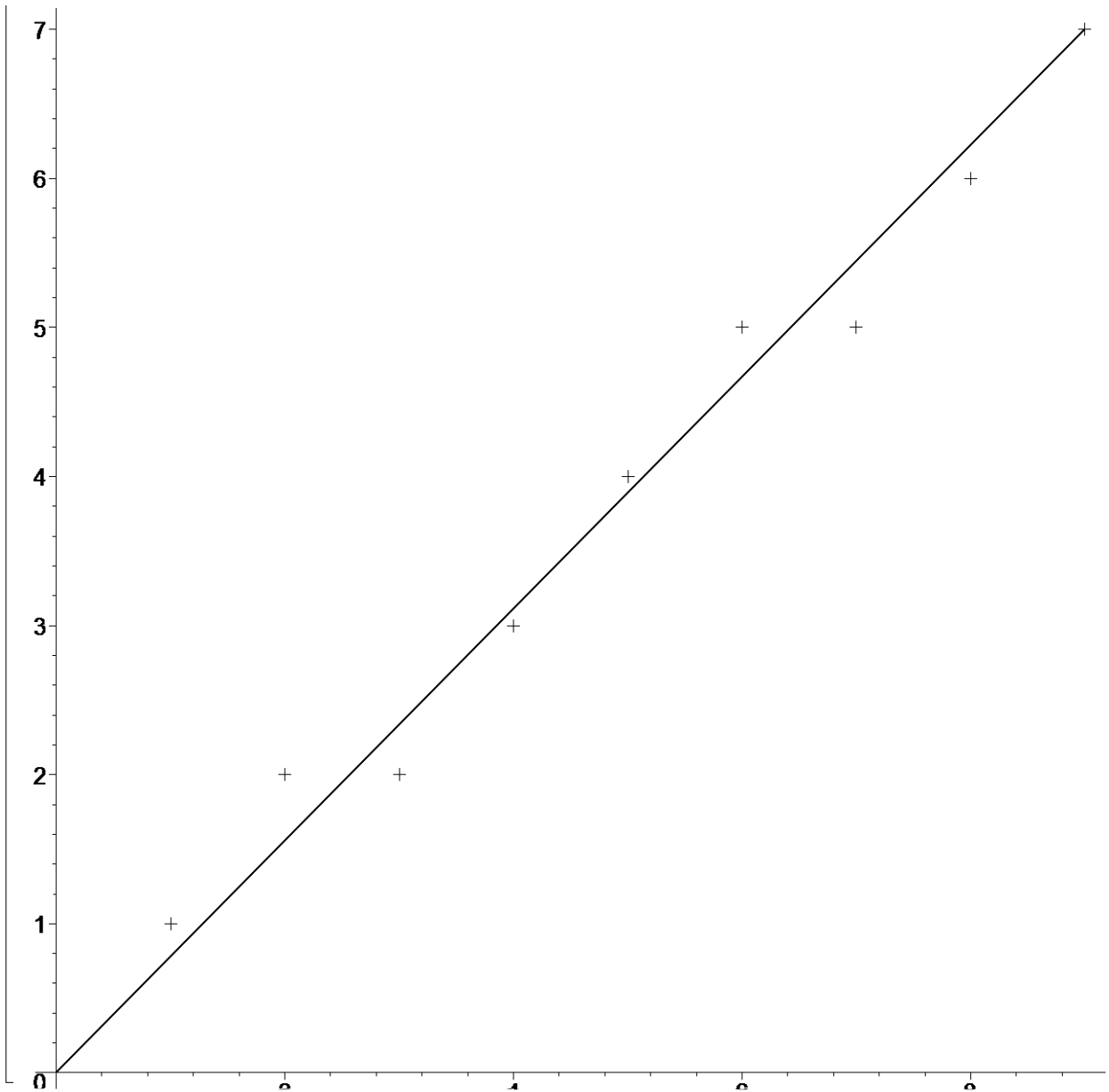
```

Und das ganze plotten:

```

> p1 := pointplot(pxl, symbol=cross, symbolsize=30):
> p2 := pointplot([[x0,y0], [x1,y1]], connect=true,
  thickness=3):
> display(p1,p2);

```



Zum Vergleich das Ganze mittels des **seq**-Befehls:

```
> [seq([x,round(m*x+b)], x=x0..x1)];
```

```
[[0,0],[1,1],[2,2],[3,2],[4,3],[5,4],[6,5],[7,5],[8,6],[9,7]]
```

```
>
```

- Variante nach Bresenham

```
[ > restart;
```

Wieder die Koordinaten festlegen:

```
> x0 := 0;
```

```
> y0 := 0;
```

```
> x1 := 9;
```

```
> y1 := 7;
```

```
x0 := 0
```

```
y0 := 0
```

```
x1 := 9
```

```
y1 := 7
```

Jetzt die Parameter für den Bresenham-Algorithmus festlegen:

```
> dx := x1-x0;  
> dy := y1-y0;  
> d := 2*dy-dx;  
> deltaE := 2*dy;  
> deltaNE := 2*(dy - dx);
```

```
dx := 9
```

```
dy := 7
```

```
d := 5
```

```
deltaE := 14
```

```
deltaNE := -4
```

Und die Startkonfiguration herstellen:

```
> x := x0;  
> y := y0;  
> pxl := [[x0,y0]];
```

```
x := 0
```

```
y := 0
```

```
pxl := [[0,0]]
```

Bresenham-Hauptschleife:

```
> for x from x0+1 to x1 do  
  if d<0 then  
    d := d+deltaE  
  else  
    d := d+deltaNE;  
    y := y+1  
  end if;  
  pxl := [op(pxl), [x,y]]  
end do;
```

```
pxl := [[0,0],[1,1]]
```

```
pxl := [[0,0],[1,1],[2,2]]
```

```
pxl := [[0,0],[1,1],[2,2],[3,2]]
```

```
pxl := [[0,0],[1,1],[2,2],[3,2],[4,3]]
```

```
pxl := [[0,0],[1,1],[2,2],[3,2],[4,3],[5,4]]
```

```
pxl := [[0,0],[1,1],[2,2],[3,2],[4,3],[5,4],[6,5]]
```

```
pxl := [[0,0],[1,1],[2,2],[3,2],[4,3],[5,4],[6,5],[7,5]]
```

```
pxl := [[0,0],[1,1],[2,2],[3,2],[4,3],[5,4],[6,5],[7,5],[8,6]]
```

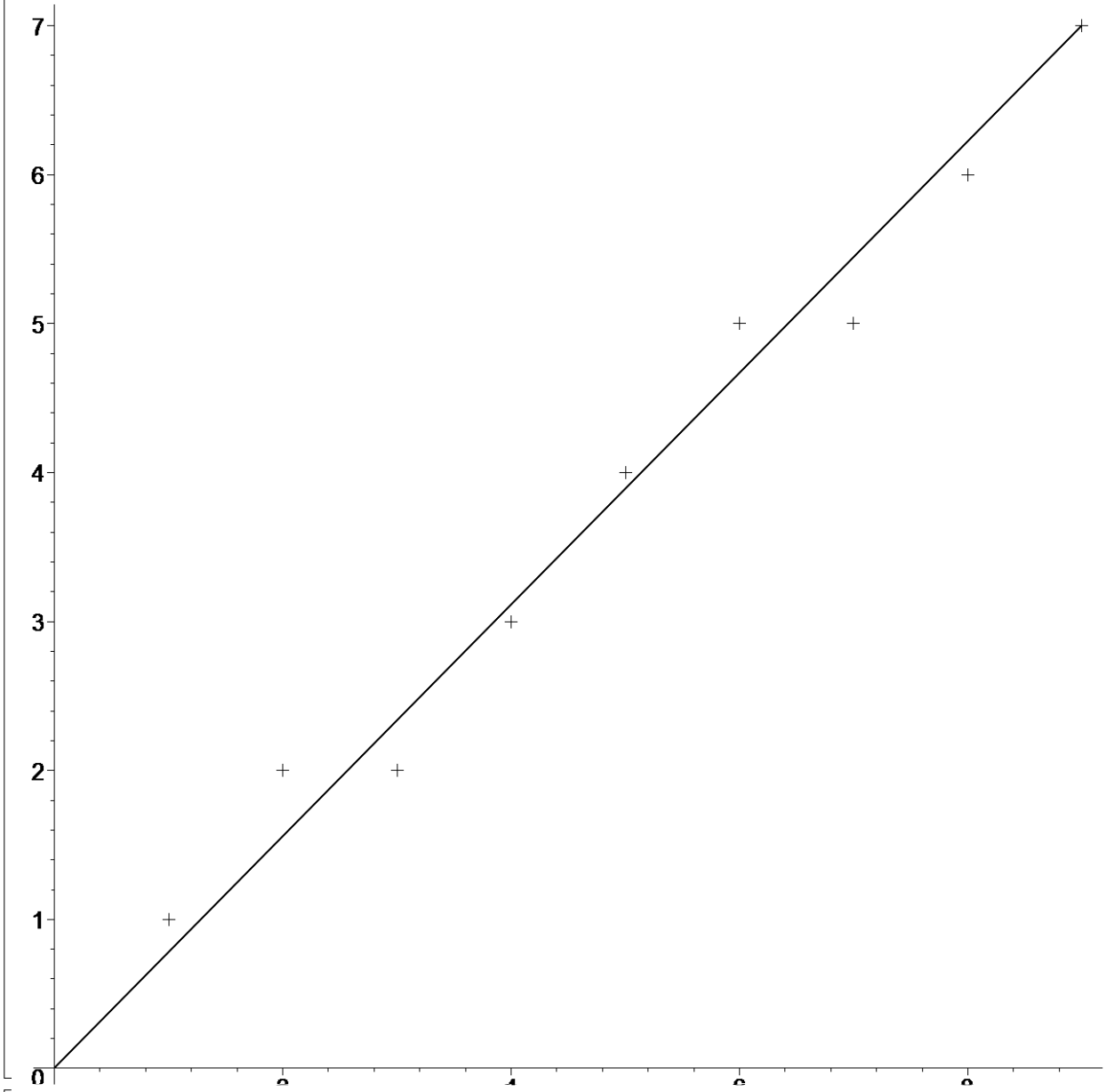
```
pxl := [[0,0],[1,1],[2,2],[3,2],[4,3],[5,4],[6,5],[7,5],[8,6],[9,7]]
```

Und wieder plotten:

```
> with(plots):  
> p1 := pointplot(pxl, symbol=cross, symbolsize=30):  
  p2 := pointplot([[x0,y0], [x1,y1]], connect=true,
```

```
thickness=3):  
display(p1,p2);
```

```
Warning, the name changecoords has been redefined
```



```
[ >  
[ >
```