

In deze tweede les van Module 5 leer je over het plotten van punten en het verschil tussen de instructies Pnt-Aan en Pxl-Aan.

Doelen:

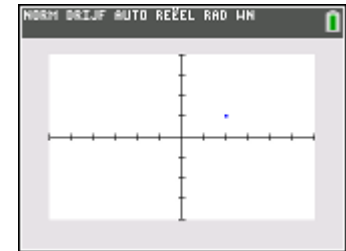
- De plotopdrachten punt en pixel gebruiken.
- Formules opstellen om plaatjes in programma's te gebruiken.

De instructie Punt

Het het menu [DRAW] PNTN (POINTS).

Pnt-Aan(x,y) plot een punt, bepaald door de instellingen van het grafiekenvenster. Het betekent 'zet het punt aan'. **Pnt-Aan(2,1)** plot het punt in het eerste kwadrant van het scherm rechts. **Pt-Aan(x,y,stijl,kleur)** heeft optionele argumenten: Stijl (1 t/m 4) en kleur. Bekijk de syntax-help door te drukken op $\boxed{+}$ terwijl je staat op de instructie in het menu [DRAW].

Pt-On(100,100) zal het punt plotten ook al valt het buiten het huidige kijkvenster.



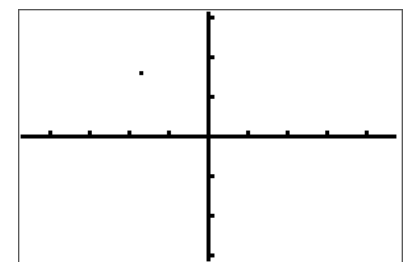
De instructie Pixel

Pxl-Aan() gebruikt de scherpixels en negeert de vensterinstellingen. **Pxl-Aan(2,3)** plot het piepkleine pixeltje in de linkerbovenhoek van hetzelfde scherm op rij 2, kolom 3. Het is bijna niet te zien! De coördinaten staan niet in de gebruikelijke (x,y) volgorde: Ze staan 'achterstevoren' omdat ze verwijzen naar (rij#, kolom#). **Pxl-Aan(x,y,color)** heeft slechts een optioneel argument voor kleur.

Er zijn ook corresponderende instructies **-Uit()**, **-Change()**, en **-Test** die we hier niet zullen behandelen.

Pixels

Afhankelijk van jouw type TI 84 Plus serie rekenmachine, heeft je scherm een vast aantal pixelkolommen en -rijen: TI-84 Plus: 96 kolommen x 64 rijen en TI-84 Plus C/CE: 265 kolommen x 165 rijen. Rijen zijn horizontaal (van links naar rechts) en kolommen zijn verticaal (van boven naar onder). De instellingen voor een gesplitst scherm, hebben invloed op het aantal rijen en/of kolommen afhankelijk van de instelling. Het grafiekscherm van de **TI-84 Plus** gebruikt de meest rechtse kolom en de onderste rij niet voor het *tekenen van grafieken*, zodat er een oneven aantal punten is in het grafiekengebied. Dit zorgt ervoor dat er een centraal punt (oorsprong) is. Het pixel helemaal linksboven is (0,0). **Pxl-Aan(0,0)** zet het pixel in kolom 0 en rij 0 aan. De nummering van de rijen en kolommen start met 0 en niet met 1.



*Het grafiekscherm van de **TI-84 Plus** heeft grotere pixels zodat ze makkelijker te zien zijn. Het scherm hierboven toont het resultaat van **Pxl-Aan(15,30)**. Dat is rij 15, kolom 30 van het scherm. Het pixel rechtsonderaan is (63,95).*

Docenten Tip: Pixel-gerelateerde opdrachten kunnen verwarrend zijn omdat de volgorde van de coördinaten is omgedraaid: de y-waarde (rijnummer) komt eerst en de x-waarde (kolomnummer) is het tweede getal. Ook is de y-richting ondersteboven: rij 0 is bovenaan en rij 165 (of 63 op de TI-84 Plus) is onderaan het scherm. Dit is hetzelfde als de oriëntatie van het rooster op het beginscherm wanneer de **Output()** opdracht wordt gebruikt (lijn#, kolom#, met lijn 1 bovenaan). De opdracht **Text()** (die tekst tekent op het grafiekscherm (zie een volgende les) gebruikt ook pixels en geen punten, voor het positioneren van de tekst.

De **-Test**-opdrachten worden gebruikt als voorwaarden in **If**-opdrachten of lussen om te testen of een punt of pixel aan staat. **Pnt-Test(0,0)** zal waar zijn als het punt (0,0) 'aan'

staat. Merk op dat een punt 'aan' kan staan en wit kan zijn, zodat het niet op het scherm verschijnt!

Op een TI-84 Plus hebben pixels en punten dezelfde grootte. Op de TI-84 Plus C/CE zijn punten groter dan pixels.

Programmeren met punten

Laten we een programma schrijven dat het grafiekenscherm willekeurig vult met punten. Dit programma zal een oneindige lus hebben dus druk op **[ON]** om uit het programma te onderbreken.

We hebben een **algoritme** (formule) nodig om een willekeurig punt op het actuele grafiekenscherm te krijgen binnen de schermgrenzen: **willekeurig** is te vinden in

het menu KANS van **M** en genereert een toevalsgetal tussen 0 en 1.

willekeurig*(Xmax-Xmin) genereert een toevalsgetal tussen 0 en Xmax-Xmin dus we zullen er **Xmin** bij optellen.

willekeurig*(Xmax-Xmin)+Xmin genereert een toevalsgetal tussen **Xmin** en

Xmax. ...en we schrijven een vergelijkbare formule voor de Y-coördinaat.

En dit is precies waarom **wiskunde** zo belangrijk is voor programmeren!

Opmerking:

AssenUit is te vinden op het scherm **[FORMAT]**.

FnUit is te vinden in het On/Off... menu van **[VARS]** Y-VARS.

PlotsUit is te vinden in het menu **[STAT PLOT]**

WisTekenen is te vinden in het menu **[DRAW]**.

While 1 creëert een oneindige lus omdat waar wordt voorgesteld als 1 op de rekenmachine.

De toevalsgetallen worden opgeslagen in **A** en **B** om te worden geplot.

Vergeet niet om op **[ON]** te drukken om het programma te onderbreken. Dit

programma werkt op dezelfde manier op elke rekenmachine uit TI-84 Plus serie!

PrgmVULPNTN verrijken met Kleur

Op de **TI-84 C/CE** kun je een willekeurige kleur toevoegen aan de opdracht

Pnt-Aan(. Het laagste getal voor een kleur is 10 en het hoogste is 24

Schrijf een opdracht die een willekeurig geheel getal genereert tussen 10 en 24 met behulp van **randInt()** en voeg dat als *derde* argument toe aan de opdracht

Pnt-Aan(.

Voeg de volgende opdracht toe voor **Pnt-Aan(A,B)**.

Veranderen **Pnt-Aan(A,B)** in **Pnt-Aan(A,B,C)**.

<uw kleur generator > → **C**

*Opmerking: Er zijn twee optionele argumenten voor **Pnt-Aan**: stijl en kleur. Stijl kan lopen van 1 tot en met 4 en kleur van 10 tot en met 24. Dus als het derde*

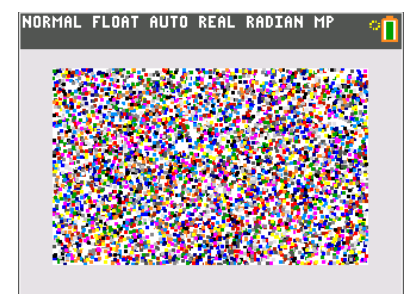
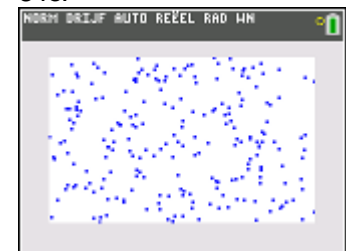
argument een waarde is van 1 tot en met 9 dan is het een stijl. Als het tussen 10

en 24 ligt is het een kleur. Andere waarden veroorzaken een fout.

Antwoord: randint(10,24) →C



Dit programma werkt op alle TI-84s.



Veelkleurige punten