

## Kapitel 5 Grafik

I denna andra lektion för kapitel 5 kommer du att lära dig att plotta punkter och skillnaden mellan Pkt-På and Pxl-På-kommandon.

### Punkt-kommando

Gå till ritmenyn genom att trycka  $\boxed{2nd}\boxed{[draw]}$  och välja <PUNKTER>.

**Pkt-På(x,y)** plottar en punkt med koordinaterna (x, y). På betyder "sätt på punkten". **Pkt-På(2,1)** plottar punkten i första kvadranten som på bilden till höger. **Pkt-På(x,y,stil,färg)** har valfria argument: stil (1 till 4) och färg. Se syntaxhjälp genom att trycka på  $\boxed{+}$  när markören är på själva kommandot i  $\boxed{[draw]}$ -menyn.

**Pkt-På(100,100)** kommer att plotta en punkt även om den ligger utanför det nuvarande fönstret.

### Pixelkommandon

**Pxl-På** använder skärmens pixlar och struntar i FÖNSTER- inställningar.

**Pxl-På(2,3)** plottar en tunn liten pixel i det övre vänstra hörnet vid rad 2, kolumn 3. Den är svår att se! Koordinaterna är inte i den vanliga (x, y)-ordningen: Det är "baklänges" eftersom de refererar till (rad#, kolumn#).

**Pxl-På(x,y,färg)** har bara ett valfritt argument för färg. Det finns också motsvarande **-Av**(, **-Växla**( och **-Test**-kommandon som vi inte tar upp här.

### Pixelar

Din TI 84 Plus-räknare har, beroende på modell, ett visst antal pixel-kolumner och rader: TI-84 Plus: **96** kolumner x **64** rader och TI-84 Plus C/CE-T: **265** kolumner x **165** rader. Rader går horisontellt och kolumner går vertikalt. Inställning med delad skärm påverkar antalet rader och/eller kolumner. Grafskärmen på **TI-84 Plus** använder inte kolumnen längst till höger eller understa raden för *grafritning* så det finns ett udda antal punkter i grafområdet. Detta gör att det finns en mittpunkt. Övre vänstra pixeln är (0,0). **Pxl-På(0,0)** sätter på pixeln vid kolumn 0, rad 0. Rader och kolumner är numrerade från noll.

**Lärarkommentar:** Pixel-relaterade satser kan vara förvirrande eftersom ordningen för koordinaterna är omvänd: y-värdet (radnummer) kommer först och x-värdet (kolumnnummer) kommer därefter. Riktningen är också upp och ner: rad 0 är högst upp och rad 165 (eller 63 på en TI-84 Plus) är längst ner på skärmen. Det är samma som orienteringen på startskärmen när man använder Output(-satsen (linje#, kolumn#, med linje 1 högst upp). Text(-satsen som "ritar" text i graffönstret (kommer i senare lektion) använder också pixlar för positionera text och inte punkter.

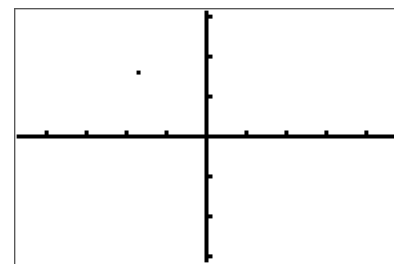
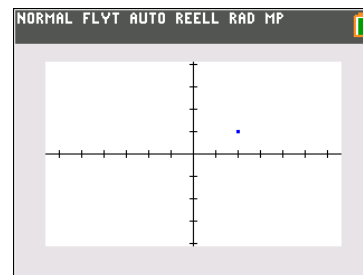
Test-satser används som villkor i If-satser eller loopar för att se om en punkt eller pixel är "PÅ". **Pkt-Test(0,0)** är sann om punkten (0,0) är "PÅ". Observera att en punkt kan vara "på" och ha färgen vit och då syns den inte på skärmen.

På en TI-84 Plus har pixlar och punkter samma storlek. På TI-84 Plus C/CE-T är punkter större än pixlar.

## Övning 2: Punkter och Pixlar

### Syfte:

- Använda punkt- och pixelplottningskommandon.
- Utveckla formler för att använda grafik i program.



Denna **TI-84 Plus** grafskärm har större pixlar så att de är lättare att se. Denna skärm visar resultatet av kommandot **Pxl-På(15,30)**. Det är rad 15 och kolumn 30 på skärmen. Längst ner till höger har vi pixel (63,95).

### Programmering med Punkter

Lås oss skriva ett program som slumpmässigt fyller GRAF-skärmen med PUNKTER. Detta program har en oändlig loop så du måste trycka på **on** för att avsluta programkörningen. Vi kommer att behöva en **algoritm** (formel) för att få en slumpalstrad punkt inom graffönstrets gränser.

**slump** hittar man genom att trycka på **math** och välja SAN (förk. SANNO-LIKHET). **Slump** genererar ett slumpantal mellan 0 and 1. **rand\*(Xmax-Xmin)** genererar ett slumpantal mellan 0 och Xmax-Xmin så vi får lägga till **Xmin**.

**slump\*(Xmax-Xmin)+Xmin** genererar alltså ett slumpantal mellan **Xmin** och **Xmax**. Vi skriver en liknande formel för Y-koordinaten.

Efter det här resonemanget är det lätt att förstå att **matematik** är viktigt i programmering!

*Obs:*

**AxlarAv** hittar du genom att trycka på **2nd**[format].

**FnAv** hittar du genom att trycka på **vars**, välja Y-VAR och sedan 4 På/Av

**DiagrAv** finns på menyn för statistikplottning. Tryck **2nd** [stat plot].

**RensaRitn** finns på DRAW-menyn. Tryck på **2nd** [draw].

**While 1** skapar en oändlig loop eftersom *Sant* representeras som 1 på räknaren.

Slumpvärdenas lagras i **A** och **B** för plottning.

Kom ihåg att du måste trycka på **on** för att avbryta programkörningen.

Detta program fungerar på samma sätt på alla räknaren i familjen TI-84 Plus!

### Utvidga programmet FYLLPKT med färg

På **TI-84 C/CE-T-modellerna** kan man lägga till ett slumpmässigt färgargument till **Pkt-On**(-satsen. Det lägsta färgnumret är 10 och det högsta 24.

Skriv en sats som genererar ett slumpmässigt heltal mellan 10 och 24 med kommandot **slumpHel( )** och lägg till detta som det tredje argumentet till **Pkt-På**(-satsen. Denna sats blir då **Pkt-På(A,B,C)**.

*Obs: det finns två valfria argument till Pkt-På: stil och färg. Stil kan ha värden mellan 1 och 4 och färg från 10 to 24.*

**Lärarkommentar:** Så här blir satserna i programmet när du använder räknarens slumpalsgenerator för att generera olika färger:

**:slumpHel(10,24)→C**

**:Pkt-På(A,B,C)**

```
NORMAL FLYT AUTO REELL RAD MP
PROGRAM: FYLLPKT
:AxlarAv
:FnAv
:DiagrAv
:RensaRitn
:While 1
:slump*(Xmax-Xmin)+Xmin→A
:slump*(Ymax-Ymin)+Ymin→B
:Pkt-På(A,B)
:End
```

