

# Programação RETR0

QBasic

PSINF



06

Desenhar

# Screen

- ❑ É possível **desenhar** em **Qbasic** utilizando **funções** muito **simples**
- ❑ Podemos **começar** por **definir** a **largura** e **altura** de um **ecrã** bem como o **número** de **cores** que pode **mostrar**
- ❑ Conseguimos isso através da instrução **SCREEN N°**
  - ❑ SCREEN 0 modo de texto – ecrã por defeito
  - ❑ SCREEN 1 WxH 320x200 4 cores
  - ❑ SCREEN 2 640x200 P&B
  - ❑ SCREEN 7 320x200 16 cores
  - ❑ SCREEN 9 640x350 16 cores
  - ❑ SCREEN 11 640x480 P&B
  - ❑ SCREEN 12 640x480 16 cores
  - ❑ **SCREEN 13** 320x200 256 cores

Altura (height)

Largura (Width)

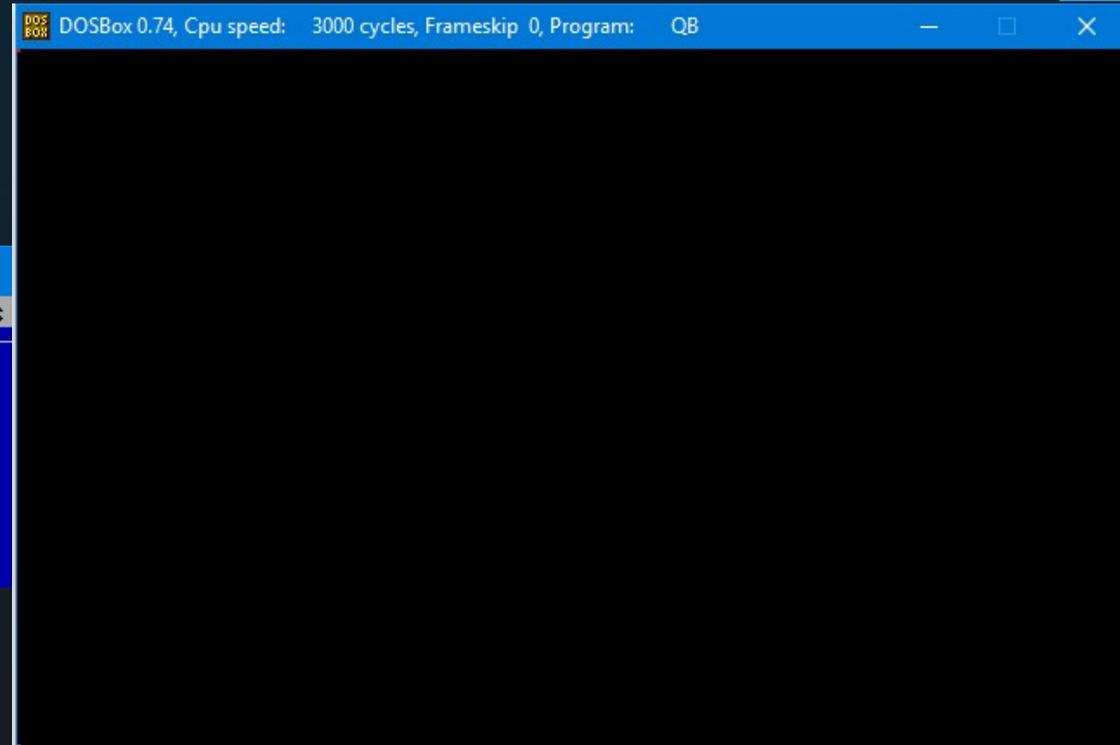
# Pset

- ❑ **PSET** - Pixel Set (pinta um pixel)
- ❑ Sintaxe

PSET(X,Y), cor

- ❑ Exemplo:

```
DOS  
BOX  
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: QB  
File Edit View Search Run Debug Calls Options  
Untitled  
SCREEN 13  
PSET (0, 0), 40  
PSET (0, 199), 30  
PSET (319, 199), 50  
PSET (319, 0), 100  
SLEEP
```

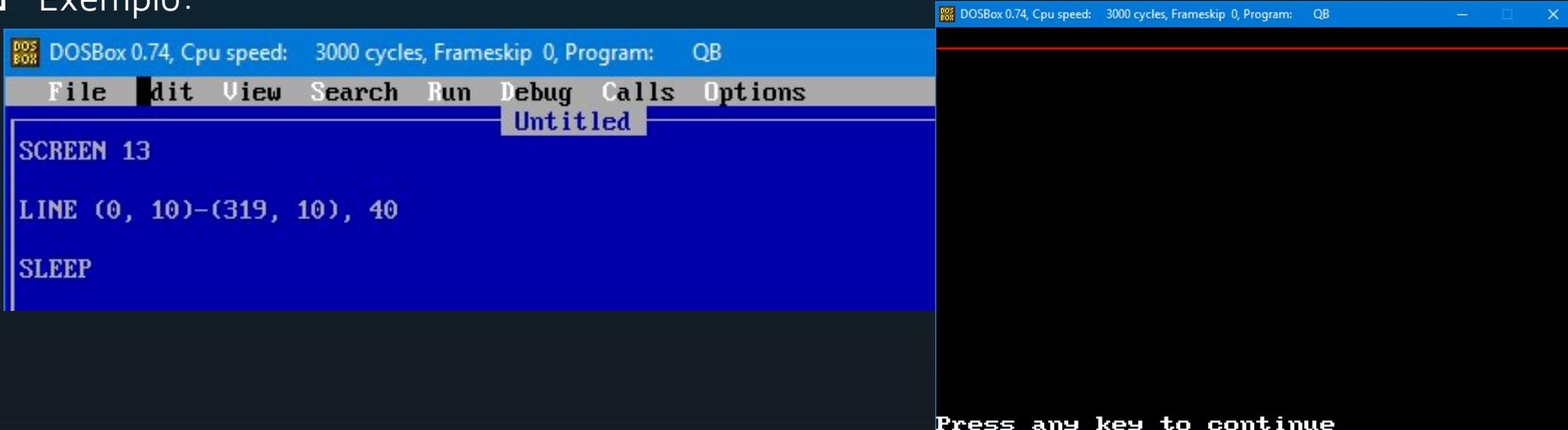


# Line

- **LINE** (traça uma linha)
- Sintaxe

$\text{LINE}(X_{\text{inicial}}, Y_{\text{inicial}})-(X_{\text{final}}, Y_{\text{final}}), [\text{cor da linha}]$

- Exemplo:



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: QB
File Edit View Search Run Debug Calls Options
Untitled
SCREEN 13
LINE (0, 10)-(319, 10), 40
SLEEP
Press any key to continue
```

# Line

- Diagonal

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: QB
File Edit View Search Run Debug Calls Options Help
BOX.BAS
SCREEN 13
LINE (0, 10)-(319, 50), 40
SLEEP
```

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: QB
Press any key to continue
```

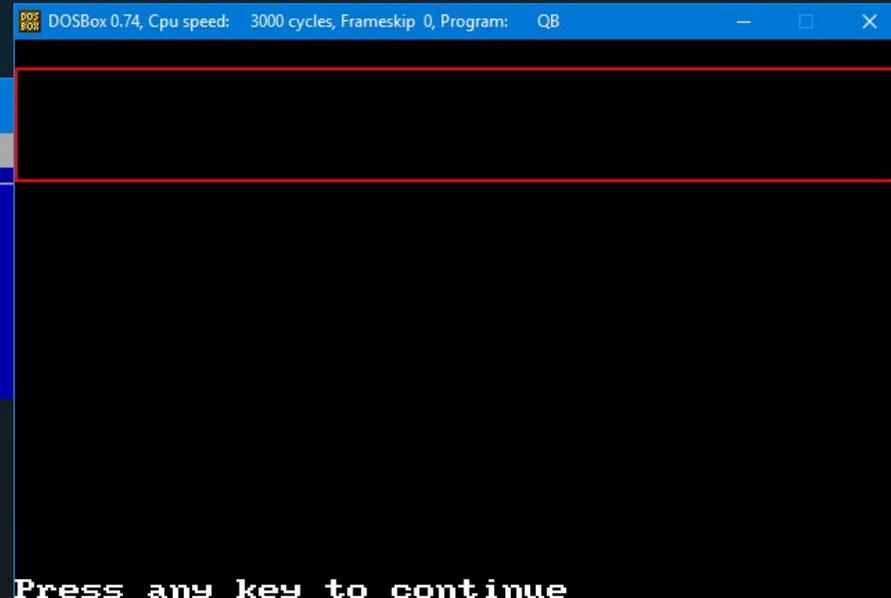
# Line BOX

- ❑ Line do tipo **BOX** faz uma **caixa com** ou **sem pintar** o seu **conteúdo**.
- ❑ Sintaxe

`LINE(Xinicial,Yinicial)-(Xfinal,Yfinal), [cor da linha], [BOX Fill]`

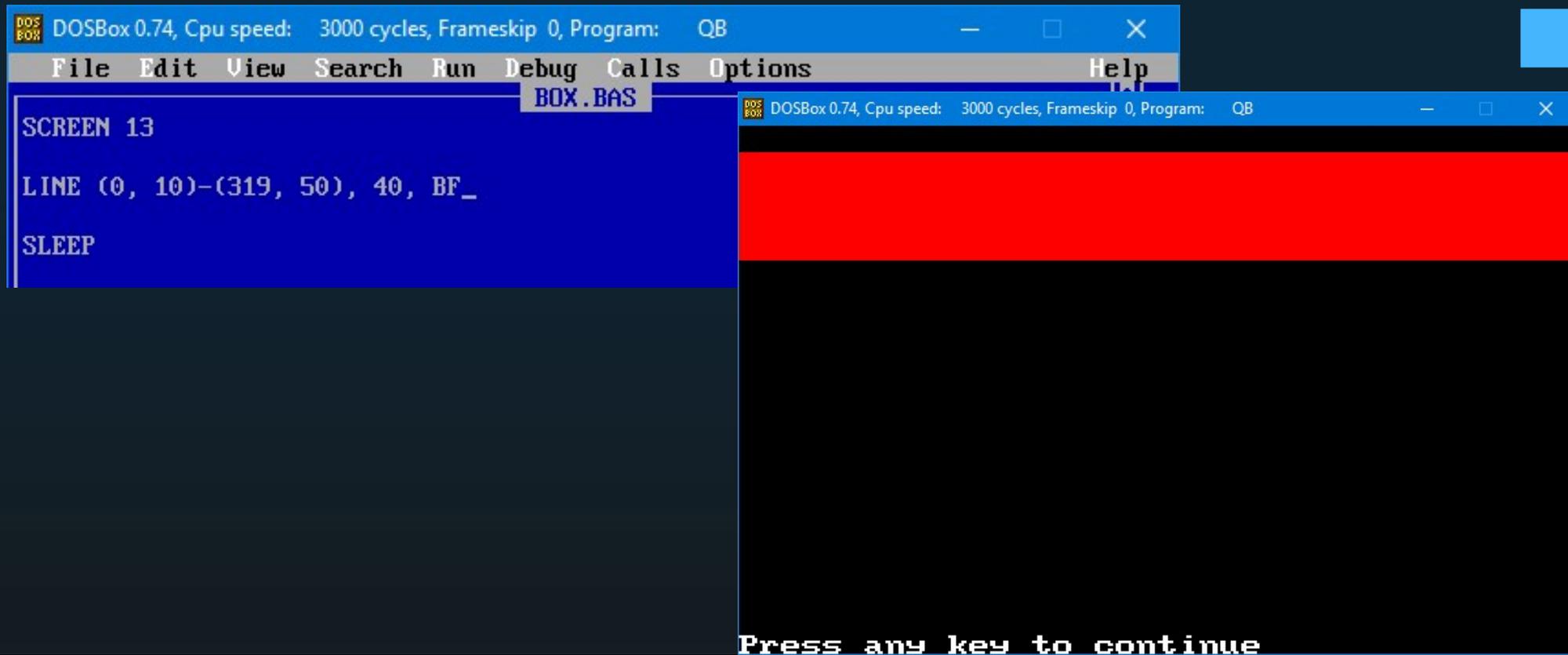
- ❑ Exemplo:

```
DOS  
BOX  
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: QB  
File Edit View Search Run Debug Calls Options  
BOX.BAS  
SCREEN 13  
LINE (0, 10)-(319, 50), 40, B_  
SLEEP
```



# Line BOX FILL

- Se quisermos com o seu conteúdo pintado:



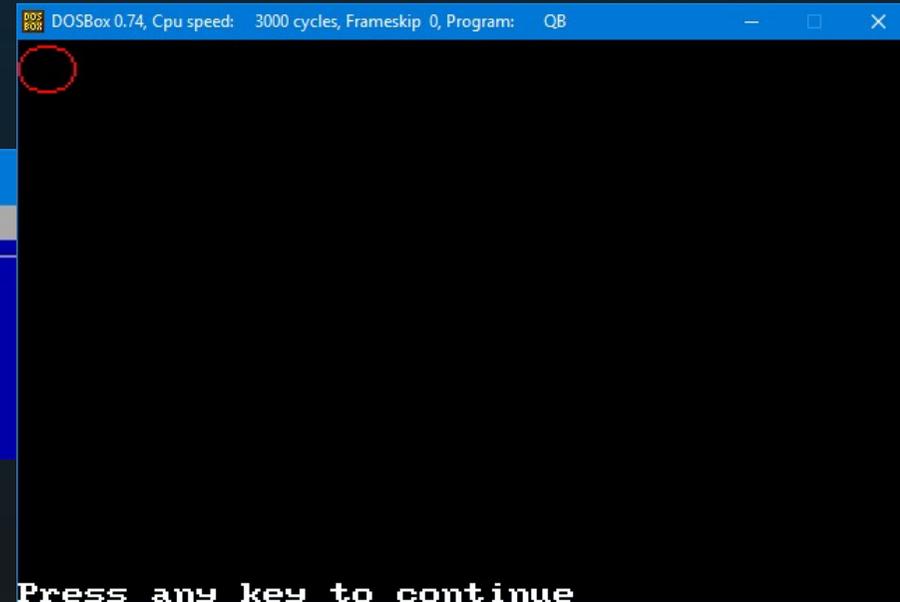
# Circle

- ❑ **CIRCLE** - Desenha um círculo
- ❑ Sintaxe

Circle( $X_{centro}$ ,  $Y_{centro}$ ), raio, cor da linha

- ❑ Exemplo

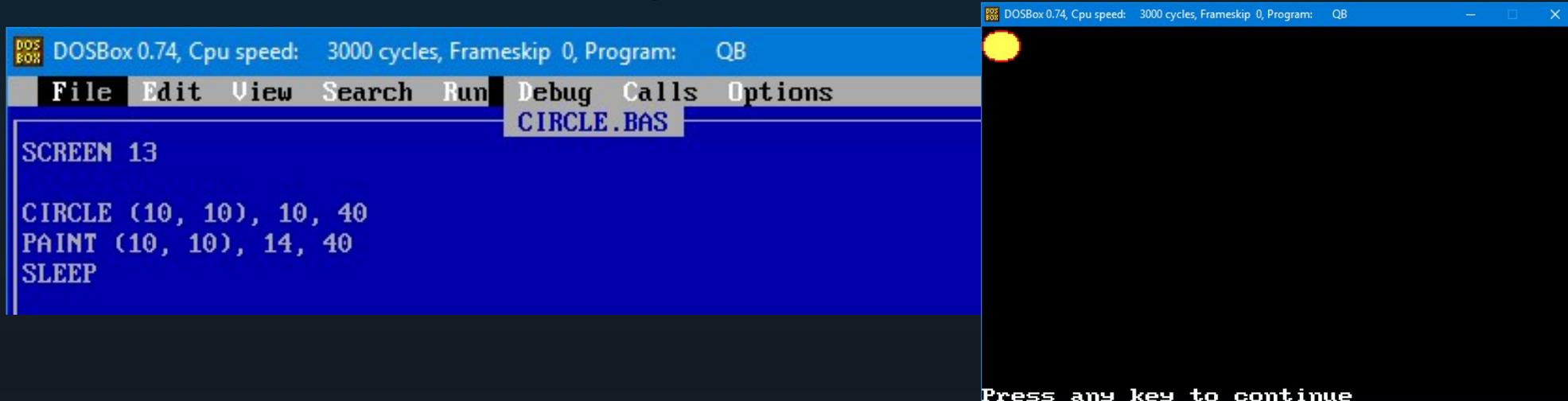
```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: QB
File Edit View Search Run Debug Calls Options
CIRCLE.BAS
SCREEN 13
CIRCLE (10, 10), 10, 40
SLEEP
```



# Circle Pintado

- ❑ Não é possível utilizar a técnica do LINE para preencher o conteúdo do círculo com cor sólida
- ❑ Temos de utilizar a instrução PAINT para conseguirmos preencher o círculo
- ❑ Sintaxe

PAINT(X,Y)<sub>dentro\_da\_forma</sub>, cor de preenchimento, até tocar na cor



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: QB
File Edit View Search Run Debug Calls Options
CIRCLE.BAS
SCREEN 13
CIRCLE (10, 10), 10, 40
PAINT (10, 10), 14, 40
SLEEP
Press any key to continue
```

# Utilizar repetição

- Podemos utilizar os ciclos para conseguirmos coisas espetaculares:

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: QB
File Edit View Search Run Debug Calls Options Help
SCREEN.BAS
DIM x AS INTEGER
SCREEN 13
CLS
FOR x = 0 TO 159 STEP 4
    LINE (159 - x, 99 - x)-(160 + x, 100 + x), x, B
NEXT x
SLEEP
```



# Para hoje

Realizar os guiões no Website.

**CREDITS:** This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik** and illustrations by **Stories**