

Programação com a linguagem Céu

Aula 18/04

Anny Caroline

`annycarolinegnr@gmail.com`

Francisco Sant'Anna

`francisco@ime.uerj.br`

Objetivos

- **FRAME_REDRAW**
- Treinar o que já foi apresentado
- Exercícios
 - Já apresentados
 - Novos

Última aula

- Vimos como movimentar um pixel na tela
 - <https://youtu.be/OAZMHm01IUk>
 - <https://youtu.be/HAYTXWoZszg>

FRAMES_REDRAW

- limpar a última posição do pixel
- **FRAMES_SET**
 - Habilita ou desabilita a geração periódica de eventos **FRAMES_UPDATE** e **FRAMES_REDRAW** para a aplicação
- `emit FRAMES_SET(yes);`

FRAMES_REDRAW

- limpar a última posição do pixel
- **FRAMES_SET**
 - Habilita ou desabilita a geração periódica de eventos **FRAMES_UPDATE** e **FRAMES_REDRAW** para a aplicação

```
every FRAMES_REDRAW do  
  //...  
end
```

```
emit FRAMES_SET(yes);

code/await Player(none) -> NEVER do
    var Point pt = val Point(0,0);

    par do
        var int key;
        every key in KEY_PRESS do
            //atualiza pt
        end
    with
        every FRAMES_REDRAW do
            emit GRAPHICS_SET_COLOR_NAME(COLOR_WHITE);
            emit GRAPHICS_DRAW_PIXEL(pt.x, pt.y);
        end
    end
end

end
```

```
emit FRAMES_SET(yes);
```

```
code/await Player(none) -> NEVER do
```

```
  var Point pt = val Point(0,0);
```

```
  par do
```

```
    var int key;
```

```
    every key in KEY_PRESS do
```

```
      //atualiza pt
```

```
    end
```

```
  with
```

```
    every FRAMES_REDRAW do
```

```
      emit GRAPHICS_SET_COLOR_NAME(COLOR_WHITE);
```

```
      emit GRAPHICS_DRAW_PIXEL(pt.x, pt.y);
```

```
    end
```

```
  end
```

```
end
```

```
if key == KEY_LEFT then
```

```
  pt.x = pt.x - 1;
```

```
else/if key == KEY_RIGHT then
```

```
  pt.x = pt.x + 1;
```

```
else/if key == KEY_UP then
```

```
  pt.y = pt.y + 1;
```

```
else/if key == KEY_DOWN then
```

```
  pt.y = pt.y - 1;
```

```
end
```

Objetivos

- ~~FRAME_REDRAW~~
- Treinar o que já foi apresentado
- Exercícios
 - Já apresentados
 - Novos

Exercício

- Baseando-se no exemplo [1-desenharPixel.ceu](#)
- Crie um desenho utilizando o **GRAPHICS_DRAW_PIXEL**
- Tente utilizar o evento **GRAPHICS_SET_COLOR_NAME**

Exercício

- Modifique o exemplo [3-linha.ceu](#) para desenhar um x
 - Utilize quantos loops achar necessário

Exercício

- Modifique o exemplo [6-parAnd-intervalo.ceu](#) para desenhar um asterisco (*)
- Utilize `par/and`

Exercício

- Modifique o exemplo [8-parOr.ceu](#) para que a tela seja apagada com o clique do mouse
- Vamos usar dois novos eventos:
 - **WINDOW_CLEAR**
 - **MOUSE_CLICK**

Exercício

- Utilize os eventos `WINDOW_SET_CLEAR_COLOR_NAME` ou `WINDOW_SET_CLEAR_COLOR_RGB` para definir a cor com que a tela deverá ser limpa

Exercício

- Modifique o exemplo [2-linha.ceu](#) para desenhar um asterisco na tela utilizando o `code/await`
- Dicas:
 - No exemplo do asterisco, identifique o que há de diferente entre os 4 blocos paralelos
 - adicione novos parâmetros ao `code/await`
 - comece tentando desenhar um X e só depois evolua para um asterisco

Exercício

- Utilizando o **every** modifique o exemplo [10-every.ceu](#) para desenhar um pixel a cada clique do mouse

Exercício

- Com um clique do mouse, parar a execução de todos os pixels
 - (baseando-se no exemplo 12-poolSemLimite.ceu)

Exercício

- Modifique o exercício anterior para limpar a tela após o clique do mouse

Exercício

- Modifique o exemplo [4-everyKeyUp.ceu](#) para movimentar o pixel ao longo da tela
- Utilize as teclas
 - KEY_UP
 - KEY_RIGHT
 - KEY_DOWN
 - KEY_LEFT
- [Comportamento esperado](#)

```
if Exp then
    //Block
else/if Exp then
    //Block
else
    //Block
end
```

Exercício

- Modifique o exercício anterior para limpar a última posição do pixel

Objetivos

- ~~FRAME_REDRAW~~
- Treinar o que já foi apresentado
- Exercícios
 - ~~Já apresentados~~
 - Novos

Ideias

- Carrinhos
- Birds
- Snake