

Programação com a linguagem Céu

`code/await` - linha

Anny Caroline

`annycarolinegnr@gmail.com`

Francisco Sant'Anna

`francisco@ime.uerj.br`

Nosso exercício do asterisco

```
par/and do
  var int i;

  loop i in [-25 -> 25] do
    emit GRAPHICS_DRAW_PIXEL(i,i);
    await 150ms;
  end
with
  var int i;

  loop i in [-25 -> 25] do
    emit GRAPHICS_DRAW_PIXEL(-i,i);
    await 100ms;
  end
with

//...
```

```
//...

  var int i;

  loop i in [-25 -> 25] do
    emit GRAPHICS_DRAW_PIXEL(i,0);
    await 50ms;
  end
with
  var int i;

  loop i in [-25 -> 25] do
    emit GRAPHICS_DRAW_PIXEL(0,i);
    await 50ms;
  end
end
```

Melhorando

- Poderíamos usar uma abstração de código para melhorar a qualidade do código

code/await

- subprograma
- pode manipular eventos
- usar composições paralelas
- pode sobreviver a múltiplas reações

Melhorando

- Vamos começar desenhando uma linha

code/await - desenhar uma linha

```
code/await Line(var int interval) -> none do
  var int i;

  loop i in [-25 -> 25] do
    emit GRAPHICS_DRAW_PIXEL(i, i);
    await (interval)ms;
  end
end

await Line(100);

emit GRAPHICS_SET_COLOR_NAME(COLOR_RED);
emit GRAPHICS_DRAW_PIXEL(0,0);
```

await

- A instrução **await** aguarda o término do **code/await Line**
- Só então o pixel vermelho pode ser desenhado

```
await Line(100);
```

```
emit GRAPHICS_SET_COLOR_NAME(COLOR_RED);
```

```
emit GRAPHICS_DRAW_PIXEL(0,0);
```

2-linha.ceu

Exercício

- Modifique o exemplo anterior 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 - Solução - X

```
code/await Line(var int interval, var int mult_x, var int mult_y) -> none do
  var int i;
  loop i in [-25 -> 25] do
    emit GRAPHICS_DRAW_PIXEL(mult_x*i, mult_y*i);
    await (interval)ms;
  end
end
```

```
par/and do
  await Line(100,1,1);
with
  await Line(100,-1,1);
end
```

X

```
emit GRAPHICS_SET_COLOR_NAME(COLOR_RED);
emit GRAPHICS_DRAW_PIXEL(0,0);
```

Exercício - Solução

```
code/await Line(var int interval, var int mult_x, var int mult_y) -> none do
    //...
end

par/and do
    await Line(100,1,1);
with
    await Line(100,-1,1);
with
    await Line(100,1,0);
with
    await Line(100,0,1);
end

emit GRAPHICS_SET_COLOR_NAME(COLOR_RED);
emit GRAPHICS_DRAW_PIXEL(0,0);
```