

# *Programação Reativa para Educação*

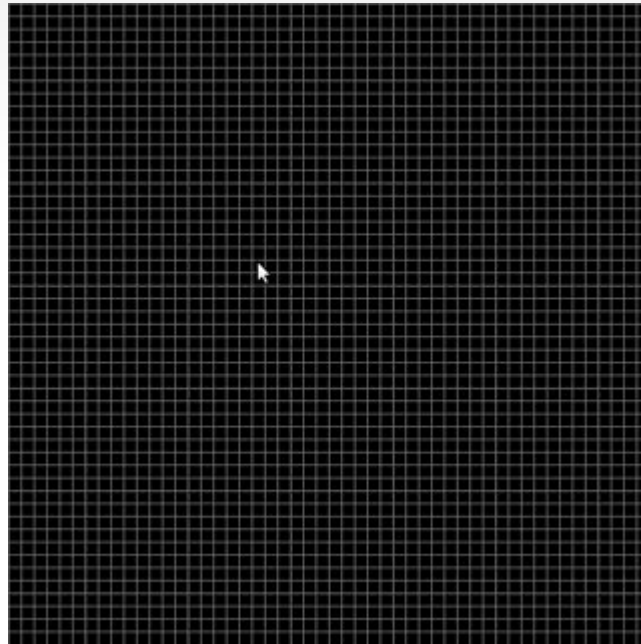
**Francisco Sant'Anna**

`francisco@ime.uerj.br`

`http://github.com/fsantanna-no/101`

# Exercício 1.1

- Desenhar um **X** na tela:



# Exercício 1.1

```
var integer p = 0;  
loop do  
  emit GRAPHICS_DRAW_PIXEL(p,p);  
emit GRAPHICS_DRAW_PIXEL(49-p,p);  
  await 100ms;  
  p = p + 1;  
  if p == 50 then  
    break;  
  end  
end
```

par do

```
var integer p1 = 0;  
loop do  
  emit GRAPHICS_DRAW_PIXEL(p1,p1);  
  await 100ms;  
  p1 = p1 + 1;  
  if p1 == 50 then  
    break;  
  end  
end
```

with

```
var integer p2 = 0;  
loop do  
  emit GRAPHICS_DRAW_PIXEL(49-p2,p2);  
  await 100ms;  
  p2 = p2 + 1;  
  if p2 == 50 then  
    break;  
  end  
end
```

end

# Exercício 2.2

- Movimentar dois pixels ao mesmo tempo
- Usar conjunto de teclas diferentes
  - como se fossem dois jogadores
- <https://fsantanna.github.io/pico-ceu/out/manual/v0.30/input/>

# Exercício 3.1

- Resolver Exercício 1.1 usando par

```
var integer x1 = 25;
var integer y1 = 25;
emit GRAPHICS_SET_RGB(0xFF,0x00,0x00);
emit GRAPHICS_DRAW_PIXEL(x1,y1);
```

```
var integer x2 = 24;
var integer y2 = 24;
emit GRAPHICS_SET_RGB(0x00,0xFF,0x00);
emit GRAPHICS_DRAW_PIXEL(x2,y2);
```

```
loop do
  var integer key = await KEY_PRESS;
  emit GRAPHICS_SET_RGB(0x00,0x00,0x00);
  emit GRAPHICS_DRAW_PIXEL(x1,y1);
  emit GRAPHICS_DRAW_PIXEL(x2,y2);
```

```
  if key == KEY_LEFT then
    x1 = x1 - 1;
  else/if key == KEY_RIGHT then
    x1 = x1 + 1;
  else/if key == KEY_UP then
    y1 = y1 - 1;
  else/if key == KEY_DOWN then
    y1 = y1 + 1;
  end
```

```
  if key == KEY_a then
    x2 = x2 - 1;
  else/if key == KEY_d then
    x2 = x2 + 1;
  else/if key == KEY_w then
    y2 = y2 - 1;
  else/if key == KEY_s then
    y2 = y2 + 1;
  end
```

```
  emit GRAPHICS_SET_RGB(0xFF,0x00,0x00);
  emit GRAPHICS_DRAW_PIXEL(x1,y1);
  emit GRAPHICS_SET_RGB(0x00,0xFF,0x00);
  emit GRAPHICS_DRAW_PIXEL(x2,y2);
```

```
end
```

par do

```
  var integer x1 = 25;
  var integer y1 = 25;
  emit GRAPHICS_SET_RGB(0xFF,0x00,0x00);
  emit GRAPHICS_DRAW_PIXEL(x1,y1);
  loop do
    var integer key = await KEY_PRESS;
    emit GRAPHICS_SET_RGB(0x00,0x00,0x00);
    emit GRAPHICS_DRAW_PIXEL(x1,y1);
    if key == KEY_LEFT then
      x1 = x1 - 1;
    else/if <...> then
      <...>
    end
    emit GRAPHICS_SET_RGB(0xFF,0x00,0x00);
    emit GRAPHICS_DRAW_PIXEL(x1,y1);
  end
```

with

```
  var integer x2 = 24;
  var integer y2 = 24;
  emit GRAPHICS_SET_RGB(0x00,0xFF,0x00);
  emit GRAPHICS_DRAW_PIXEL(x2,y2);
  loop do
    var integer key = await KEY_PRESS;
    emit GRAPHICS_SET_RGB(0x00,0x00,0x00);
    emit GRAPHICS_DRAW_PIXEL(x2,y2);
    if key == KEY_a then
      x2 = x2 - 1;
    else/if <...> then
      <...>
    end
    emit GRAPHICS_SET_RGB(0x00,0xFF,0x00);
    emit GRAPHICS_DRAW_PIXEL(x2,y2);
  end
```

end

# Paralelismo Lógico

- Distinção entre concorrência e paralelismo
- Modelo de execução síncrono
  - sem locks
  - determinístico
- `par` , `par/and` , `par/or`

# Trabalho 1

- Criar um jogo ou aplicação
  - partir do exercício 3.1



# Trabalho 1

- Demonstrações
- Criatividade
- “Competição” entre ideias e não entre implementações