

Fundamentos da Computação

Francisco Sant'Anna

`francisco@ime.uerj.br`

`http://github.com/fsantanna-uerj/Fundamentos`

Programação Estruturada

- Entrada e Saída
- Atribuição
- Estruturas de Controle
 - Sequência
 - Condicional
 - Repetição
 - *Paralelismo (lógico)*
- Abstrações
 - Código
 - Dados

```

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
end

```

```

code Pix (var integer x, y,
  var integer r, g, b,
  var integer key_left, key_right,
  key_up, key_down)
  -> FOREVER
do
  emit GRAPHICS_SET_RGB(r,g,b);
  emit GRAPHICS_DRAW_PIXEL(x,y);
  loop do
    var integer key = await KEY_PRESS;
    emit GRAPHICS_SET_RGB(0x00,0x00,0x00);
    emit GRAPHICS_DRAW_PIXEL(x,y);
    if key == key_left then
      x = x - 1;
    else/if <...> then
      <...>
    end
    emit GRAPHICS_SET_RGB(r,g,b);
    emit GRAPHICS_DRAW_PIXEL(x,y);
  end
end

par do
  await Pix(25,25,
    0xFF,0x00,0x00,
    KEY_LEFT,KEY_RIGHT,
    KEY_UP,KEY_DOWN);
with
  await Pix(24,24,
    0x00,0xFF,0x00,
    KEY_a,KEY_d,
    KEY_w,KEY_s);
end

```

```

data Posicao with
    var integer x;
    var integer y;
end

```

```

data Cor with
    var integer r;
    var integer g;
    var integer b;
end

```

```

data Controle with
    var integer left;
    var integer right;
    var integer up;
    var integer down;
end

```

```

code/await Pix (var Posicao posicao,
                 var Cor cor,
                 var Controle controle)
    -> FOREVER

```

```

do
    <...>
end

```

```

par do
    await Pix(Posicao(25,25),
              Cor(0xFF,0x00,0x00),
              Controle(KEY_LEFT,<...>));

```

```

with
    await Pix(Posicao(24,24),
              Cor(0x00,0xFF,0x00),
              Controle(KEY_a,<...>));

```

```

end

```

```

code/await Pix (var Posicao pos,
                 var Cor cor,
                 var Controle controle)
    -> FOREVER

```

```

do

```

```

    emit GRAPHICS_SET_RGB(cor.r, cor.g, cor.b);
    emit GRAPHICS_DRAW_PIXEL(pos.x, pos.y);

```

```

loop do

```

```

    var integer key = await KEY_PRESS;
    emit GRAPHICS_SET_RGB(0x00,0x00,0x00);
    emit GRAPHICS_DRAW_PIXEL(pos.x, pos.y);
    if key == controle.left then

```

```

        pos.x = pos.x - 1;

```

```

    else/if key == controle.right then

```

```

        pos.x = pos.x + 1;

```

```

    else/if key == controle.up then

```

```

        pos.y = pos.y - 1;

```

```

    else/if key == controle.down then

```

```

        pos.y = pos.y + 1;

```

```

    end

```

```

    emit GRAPHICS_SET_RGB(cor.r, cor.g, cor.b);

```

```

    emit GRAPHICS_DRAW_PIXEL(pos.x, pos.y);

```

```

end

```

```

end

```

Trabalho 2

- Baseado no Trabalho 1
 - Aplicar sugestões do professor
 - Adotar abstrações de código e dados

Trabalho 2

- Avaliações individualizadas
- Dificulta cola
- Sugestões tiram o aluno da zona de conforto