

Programação Multimédia

Interacção

Conceitos Avançados

- **Interacção (Rato e Teclado)**
- Classes e Objectos
- Imagem, 3D e Captura de Vídeo
- Movimentos Reais
- Comunicação OSC
- DiABlu

Rato - Exemplo #1

```
void setup() {  
    size(400, 400);  
  
}  
  
void draw() {  
    background(0);  
  
    // desenhar um rectangulo na posicao do ponteiro do rato  
    rect(mouseX, mouseY, 10, 10);  
}
```

Posição do ponteiro do rato

- O Processing mantém duas variáveis internas (não precisam de ser declaradas) com a posição actual do ponteiro
 - `mouseX` e `mouseY`

Rato - Exercício #1

- Modifique o Exemplo #1 para desenhar um rectângulo cuja altura depende da posição X do ponteiro do rato

Rato - Exemplo #2

```
void setup() {
    size(400, 400);
}

void draw() {
    background(0);

    if (mousePressed) {
        fill(0, 0, 255); //azul
    } else {
        fill(255); //branco
    }

    // desenhar um rectangulo na posicao do ponteiro do rato
    rect(mouseX, mouseY, 10, mouseX);
}
```

Determinar se um botão está pressionado

- A variável interna `mousePressed` (booleana) indica se há algum botão pressionado
 - Não indica qual o botão

Rato - Exercício #2

- Modifique o Exemplo #2 para desenhar um círculo centrado na posição do rato, se algum botão estiver pressionado

Rato - Exemplo #3

```
void setup() {
  size(400, 400);
}

void draw() {
  background(0);

  if (mousePressed) {
    if (mouseButton == LEFT) {
      fill(255, 0, 0);          // vermelho
    } else if (mouseButton == CENTER) {
      fill(0, 255, 0);        // verde
    } else if (mouseButton == RIGHT) {
      fill(0, 0, 255);        // azul
    }
  } else {
    fill(255);                //branco
  }

  // desenhar um rectangulo na posicao do ponteiro do rato
  rect(mouseX, mouseY, 50, 50);
}
```

Determinar o botão pressionado

- A variável `mouseButton` (int) indica o botão pressionado através do seu valor
 - `LEFT`, `CENTER` ou `RIGHT` (variáveis internas que funcionam como constantes durante o programa)
- É necessário primeiro verificar se há algum botão pressionado!

Rato - Exercício #3

- Modifique o Exemplo #3 para contar quantas vezes o botão direito foi pressionado
 - A altura do rectângulo deve ser igual à contagem actual
 - Quando a contagem chegar a 50 deve ser reiniciada (zero)

Resumo da utilização do rato

- O Processing mantém variáveis globais que podemos utilizar para saber qual a posição do rato ou qual o botão pressionado
- **mouseX, mouseY**
 - Posição do ponteiro
- **mousePressed**
 - Indica se há algum botão pressionado (mas não qual)
- **mouseButton (LEFT, CENTER, RIGHT)**
 - Qual o botão pressionado

Teclado – Exemplo #4

```
void setup() {  
    size(400, 400);  
}  
  
void draw() {  
    if (keyPressed) {  
        background(200, 0, 0);  
    } else {  
        background(0);  
    }  
}
```

Determinar se uma tecla está pressionada

- Variável interna **keyPressed** (booleana)
 - Funcionamento idêntico ao **mousePressed**

Teclado – Exemplo #5

```
int x, y;

void setup() {
  size(400, 400);

  x = 0;
  y = 0;
}

void draw() {
  if (keyPressed) {
    if (key == 'a') {
      x = x - 1;
    } else if (key == 'w') {
      y = y - 1;
    } else if (key == 'd') {
      x = x + 1;
    } else if (key == 'x') {
      y = y + 1;
    }
  }
  rect(x, y, 40, 40);
}
```

Teclado – Exercício #4

- Modifique o Exemplo #5 de forma a que pressionando as teclas
 - ‘1’ : o rectângulo mude a cor da borda para azul
 - ‘2’ : o rectângulo mude a cor da borda para branco
 - ‘3’ : o rectângulo mude a cor da borda para preto

Determinar a tecla pressionada

- A variável interna `key` indica qual a tecla pressionada
 - Apenas funciona para teclas com caracteres associados!

Teclado – Exemplo #6

```
int x, y;

void setup() {
  size(400, 400);

  x = 0;
  y = 0;
}

void draw() {
  if (keyPressed) {
    if (keyCode == LEFT) {
      x = x - 1;
    } else if (keyCode == UP) {
      y = y - 1;
    } else if (keyCode == RIGHT) {
      x = x + 1;
    } else if (keyCode == DOWN) {
      y = y + 1;
    } else if (key == '1') {
      stroke(0, 0, 255);
    } else if (key == '2') {
      stroke(255);
    } else if (key == '3') {
      stroke(0);
    }
  }
  rect(x, y, 40, 40);
}
```

Determinar a tecla pressionada

- A variável interna `keyCode` indica qual a tecla pressionada
 - No caso das teclas especiais
 - Alguns valores:
 - LEFT, RIGHT, UP, DOWN (teclas de direcção)
 - SHIFT, CONTROL, ALT

Resumo da utilização do teclado

- O Processing mantém variáveis globais que podemos utilizar para saber se há alguma tecla pressionada e qual
- **keyPressed**
 - Indica se há alguma tecla pressionada (mas não qual)
- **key**
 - Indica a tecla pressionada ('a', 'b', '1', etc)
- **keyCode**
 - Indica a tecla pressionada (LEFT, SHIFT, CONTROL)

Interacção Rato/Teclado

- Existem duas formas de obtermos o input
 1. Num determinado ponto do programa verificamos o estado do rato/teclado
 - O que fizemos até agora
 2. Podemos pedir ao Processing para interromper o nosso programa quando uma tecla/botão for carregado

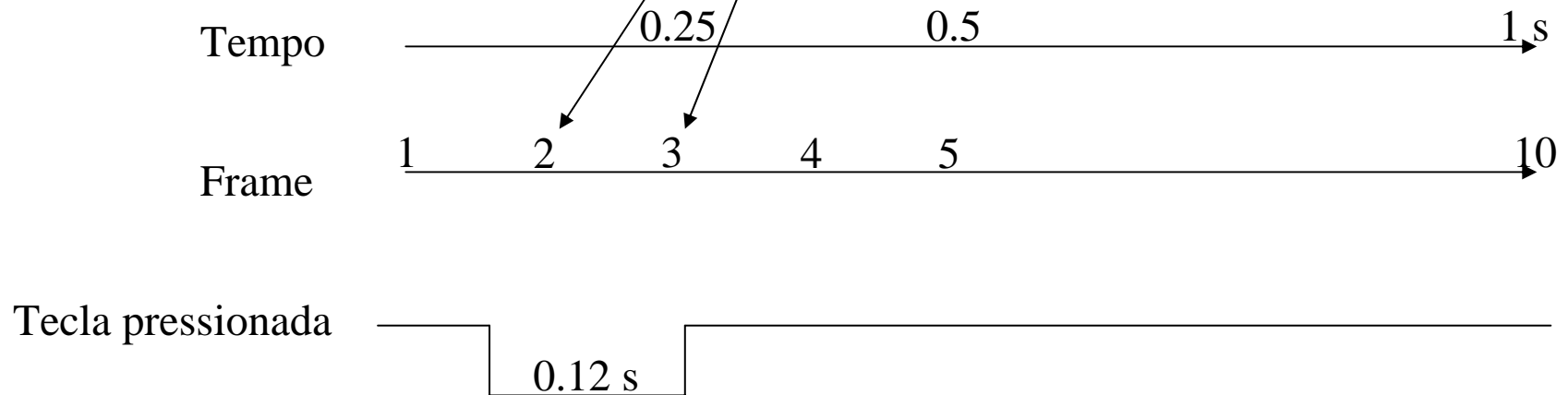
Interacção Rato/Teclado

1. No primeiro caso há pouco controlo sobre os eventos
 - Se o utilizador ficar a pressionar a tecla durante alguns instantes, o programa conta várias vezes

Primeiro Caso

- 10 FPS
 - 0.10 segundos entre frames
 - Neste caso o pressionar de tecla é detectado em 2 frames!

Em cada frame verificamos o estado das teclas



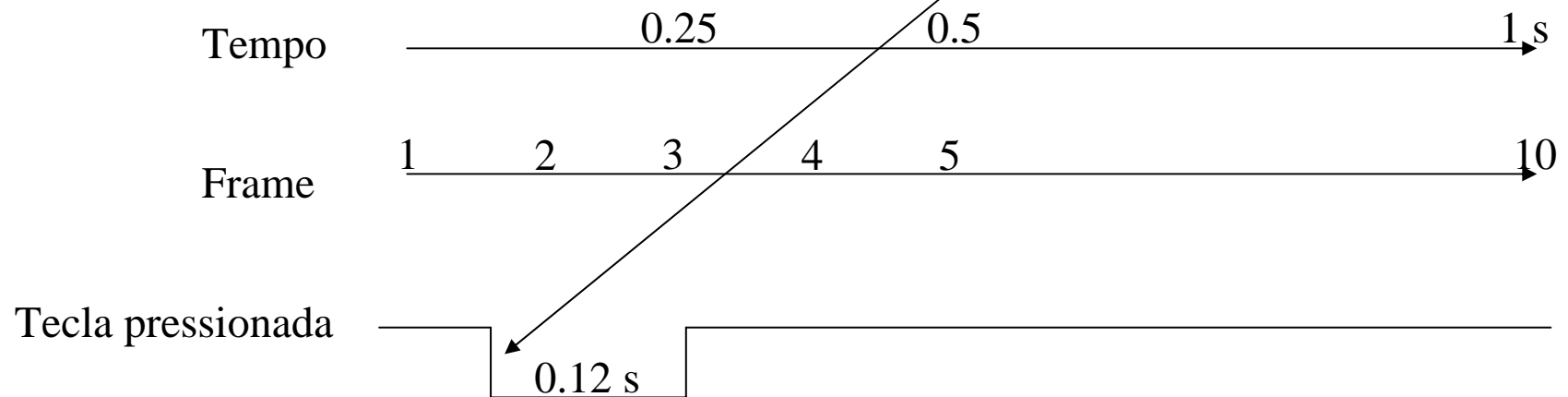
Exemplo #7

```
void setup() {  
    size(400, 400);  
}  
  
void draw() {  
    background(0);  
  
    if (keyPressed) {  
        rect(100, 100, 50, 50);  
        println(frameCount);  
    }  
}
```

Segundo Caso

- 10 FPS
 - 0.10 segundos entre frames

Um evento é lançado para avisar o programa que uma tecla foi pressionada



Exemplo #8

```
void setup() {  
    size(400, 400);  
}  
  
void draw() {  
    background(0);  
}  
  
void keyPressed() {  
    rect(100, 100, 40, 40);  
    println(frameCount);  
}
```

Eventos

- O método `keyPressed` é chamado pelo Processing quando um evento de tecla pressionada é gerado pelo utilizador
 - O programador, se quiser escutar estes eventos, implementa o código deste método

Rato

- **Eventos**
 - **mouseDragged()** – chamado quando o rato é arrastado
 - **mouseMoved()** – chamado quando o rato é movido
 - **mouseReleased()** – chamado quando o botão do rato é largado
 - **mousePressed()** – chamado quando o botão do rato é pressionado

Teclado

- **Eventos**
 - `keyReleased()` – chamado quando largamos largamos uma tecla
 - `keyPressed()` – chamado quando pressionamos uma tecla

Exercício #5

- Modifique o Exercício #4 para utilizar os eventos

Projecto Semanal

- Crie um programa que permita ao utilizador configurar uma cara:
 - Deve ser possível mover cada elemento da face

