TUTORIAIS

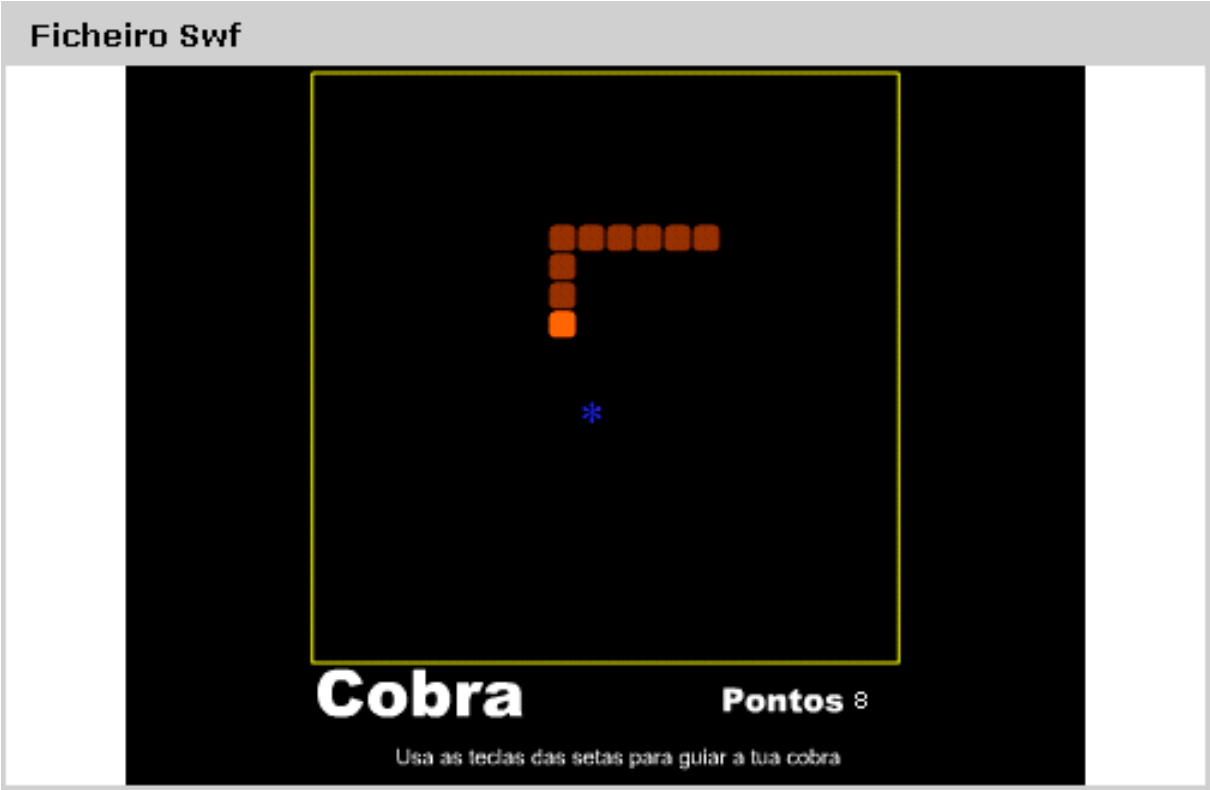
FLASHWEBTRAINING.COM

VISUALWORK®

JOGO DA COBRA

Introdução

Neste tutorial não vamos ensinar a fazer este jogo passo a passo mas sim analisar os vários scripts usados na sua concepção, por isso aconselhamos a ler o tutorial acompanhando com a análise da source que está em **downloads**.



Tutorial

Esta é a imagem da timeline principal que contém as actionscripts que vamos analisar em detalhe:

VISUALWORK © 2002 VISUALWORK Produções Multimedia, Lda. Todos os direitos reservados. Qualquer reprodução, incluindo fotocópia, só pode ser feita com autorização expressa da VISUALWORK. FLASHWEBTRAINING.com, truquesedicas.com e respectivos logos são marcas registadas da VISUALWORK Produções Multimedia, Lda.

Vamos então explicar o que se passa em cada frame:

Frame 2 - Layer random actions

Em primeiro lugar vamos analisar esta frame para que seja entendido mais facilmente o resto do código.

Este código controla as peças que a cobra vai comer bem como o som que lançado nesse momento e aumenta a cobra.

```
if (getproperty("star",_x) == getproperty("master",_x) && getproperty("star",_y) ==
getproperty("master",_y)) {
tellTarget ("sound") {
play ();
}
score = number(score) +1;
Rxpos = random(20);
Rypos = random(20);
setProperty ("star", _y, Rypos * number(20) + 20);
setProperty ("star", _x, Rxpos * number(20) + 20);
if (clones < clonelimit) {
clone = number(clone)+1;
duplicateMovieClip ("master", "slave" add clone, clone);
tellTarget ("master") {
gotoAndStop (35);
}
}
}
```

Este script detecta quando a posição x,y do Movie Clip "master" (representa a cobra) é precisamente a mesma do Movie Clip "star" (representa o objecto a ser comido pela cobra), então faz tocar um som [linha 3 - 5]. Depois disto, é adicionado um ponto à pontuação e calculado uma nova posição para o Movie Clip "star".

Como podes ver, há um controlo de limite de numero de clones (clonelimit == 300). Se clones < clonelimit então incrementa 1 valor à variável clones e duplica o Movie Clip "master" com o nome slave concatenado com o valor da variável clone .

VISUALWORK® © 2002 VISUALWORK Produções Multimedia, Lda. Todos os direitos reservados. Qualquer reprodução, incluindo fotocópia, só pode ser feita com autorização expressa da VISUALWORK.
FLASHWEBTRAINING.com, truquesedicas.com e respectivos logos são marcas registadas da VISUALWORK Produções Multimedia, Lda.

Frame 1 - Layer actions da cobra

```
setProperty ("master", _y, "20");
setProperty ("master", _x, "20");
setProperty ("master", _xscale, "40");
setProperty ("master", _yscale, "40");
hit = "0";
dir = "";
Rxpos = random(19);
Rypos = random(19);
setProperty ("star", _y, Rypos * number(20) + 20);
setProperty ("star", _x, Rxpos * number(20) + 20);
setProperty ("star", _xscale, "40");
setProperty ("star", _yscale, "40");
setProperty ("grid", _y, 210);
setProperty ("grid", _x, 210);
clone = "0";
score = "0";
clonelimit = "300";
n = "0";
tellTarget ("sound") {
play ();
}
```

clonelimit = "300";

Esta linha é a única que merece uma explicação mais detalhada. A variável "clonelimit" tem como objectivo guardar o valor que irá limitar, neste caso a 300, o número de duplicações do Movie Clip que constitui a cobra , condição esta feita na frame #2 da "layer random actions" como vimos anteriormente. Isto significa que o jogador terá sucesso quando conseguir comer 300 estrelas, depois disto não haverá mais duplicação das estrelas. Podes inserir o valor que quiseres, este valor funciona como constante.
O resto do código serve para posicionar e redimensionar os vários elementos no inicio do jogo, através da action Setproperty e com as propriedades size/width/position.

VISUALWORK © 2002 VISUALWORK Produções Multimedia, Lda. Todos os direitos reservados. Qualquer reprodução, incluindo fotocópia, só pode ser feita com autorização expressa da VISUALWORK.
FLASHWEBTRAINING.com, truquesedicas.com e respectivos logos são marcas registadas da VISUALWORK Produções Multimedia, Lda.

Frame 2 - Layer actions da cobra

Este código é usado basicamente para movimentar a serpente (só o Movie Clip "master") e detectar os limites onde a serpente se pode movimentar (o limite é o Movie Clip "grid").

```
xpos = getProperty("master", _x);
ypos = getProperty("master", _y);
if (hit == 0 ) {
if (xpos < 10 or xpos > 410 or ypos < 5 or ypos > 400) {
hit = "1";
}
}
```

É aqui que a colisão é detectada, se a condição mais interna for verdadeira, ou seja, se o "master" tiver colidido com os limites a variável hit passa a ter o valor 1.

```
if (hit == 1) {
hit = "2";
n = "0";
tellTarget ("master") {
gotoAndPlay (11);
}
gotoAndPlay (4);
}
```

Quando o hit é 1, ou seja colidiu com os limites, o "master" explode e é atribuído o valor 2 á variável "hit"

```
if (hit == 0) {
if (dir eq "right") {
setProperty ("master", _x, xpos+Number(20));
}
if (dir eq "left") {
setProperty ("master", _x, xpos-Number(20));
}
if (dir eq "up") {
setProperty ("master", _y, ypos-Number(20));
}
if (dir eq "down") {
setProperty ("master", _y, ypos+Number(20));
}
}
```

VISUALWORK © 2002 VISUALWORK Produções Multimedia, Lda. Todos os direitos reservados. Qualquer reprodução, incluindo fotocópia, só pode ser feita com autorização expressa da VISUALWORK. FLASHWEBTRAINING.com, truquesedicas.com e respectivos logos são marcas registadas da VISUALWORK Produções Multimedia, Lda.

Se esta condição for verdadeira o "master" será movimentado 20 pixeis num sentido que depende do valor da variável "dir". O valor desta variável é atribuído pelo script contido no botão colocado na layer "botão das teclas".

Frame 2 - layer rabo

Este código cola o último duplicado do "master" na posição actual do "master" simulando assim o movimento.

```
Txpos = getProperty("master", _x);
Typos = getProperty("master", _y);
if (n < clone) {
n = number(n)+1;
if (dir eq "right" or dir eq "left" or dir eq "up" or dir eq "down") {
setProperty ("slave" add n , _x, Txpos);
setProperty ("slave" add n , _y, Typos);
}
setProperty ("slave" add n, _alpha, "50");
tellTarget ("slave" add n) {
gotoAndStop (25);
}
}
if (clone == n) {
n = 0;
}
```

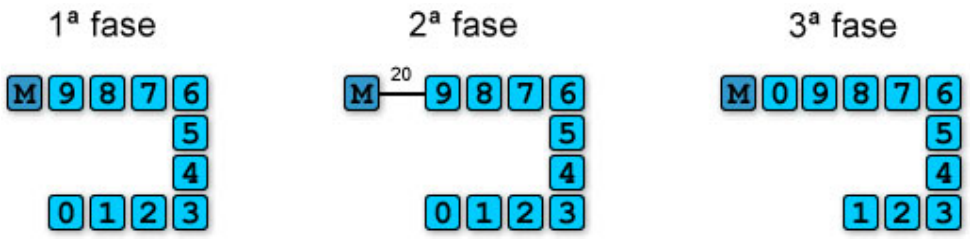
Quando o script é iniciado são guardados nas variáveis Txpos e Typos os valores da posição do "master" nesse momento. A seguir o duplicado "slave" de índice n é posicionado através dos valores reservados nas variáveis Txpos e Typos.

Chegando ao fim do script o filme avança para a frame seguinte que tem instruções para voltar a esta frame.

Quando o script é novamente lido o duplicado que é posicionado é o de índice seguinte. Quando a variável n é igual ao número de duplicados (valor da variável clone) , o n é actualizado a 0 e assim quando o script for novamente lido tudo começa de novo.

Como isto é um pouco complicado de explicar recorremos a um esquema para auxiliar a explicação, neste caso n=0 e clone=9

VISUALWORK © 2002 VISUALWORK Produções Multimedia, Lda. Todos os direitos reservados. Qualquer reprodução, incluindo fotocópia, só pode ser feita com autorização expressa da VISUALWORK. FLASHWEBTRAINING.com, truquesedicas.com e respectivos logos são marcas registadas da VISUALWORK Produções Multimedia, Lda.



Frame 2 - Layer auto colisão

```
t = clone;
```

Quando o "master" é duplicado, na frame #2 da Layer "random actions", o resultado do nome do duplicado é "slave" concatenado com o valor da variável "clone". O resultado é "slave5" ou "slave3" conforme o valor da variável "clone". Assim a variável "t" assume o numero de clones que existe.

Frame 3 - Layer auto colisão

Aqui é detectado se a cobra bate em si própria.

```
while (t > 0) {
slaveX = getproperty("slave" add t,_x);
slaveY = getproperty("slave" add t,_y);
masterX = getproperty("master",_x);
masterY = getproperty("master",_y);
t = number(t) - 1;
if (slaveX == masterX && slaveY == masterY) {
hit = "1";
gotoAndPlay (2);
}
}
if (t == 0) {
gotoAndPlay (2);
}
```

O código vai verificar se a posição do "master" coincide com algum dos duplicados que o seguem. Isto é calculado através de um ciclo while com a condição t>0, em que t é o numero de duplicados que existem, como vimos anteriormente. Em cada ciclo é guardado em variáveis o valor da posição do "master" bem como do slave referente ao valor do t, ou seja "slave10", "slave9", "slave8", "slave7", até ao valor ser de -1. Assim, é garantido que a posição do "master" é testado com todos os duplicados.

VISUALWORK © 2002 VISUALWORK Produções Multimedia, Lda. Todos os direitos reservados. Qualquer reprodução, incluindo fotocópia, só pode ser feita com autorização expressa da VISUALWORK. FLASHWEBTRAINING.com, truquesedicas.com e respectivos logos são marcas registadas da VISUALWORK Produções Multimedia, Lda.

Quando as posições coincidem a variável hit assume o valor 1 e o filme avança para a frame #2 e explode.

Frame 4 - Layer apagar duplicados

O objectivo deste script é apagar todos os duplicados quando o a cobra explode.

```
if (n < clone) {
n = number(n)+1;
removeMovieClip ("slave" add n);
}
if (clone == n) {
n = 0;
gotoAndPlay (1);
}
```

Os duplicados são todos corridos através da concatenação da string slave com a variável n que vai sendo decrementada. Não existe nenhum ciclo porque no fim do script o filme vai para a frame #5 e está tem uma action que o obriga a ir para a frame #4. Quando a variável "clone" é igual à "n" o filme vai para a frame #1 começando o jogo novamente.

VISUALWORK® © 2002 VISUALWORK Produções Multimedia, Lda. Todos os direitos reservados. Qualquer reprodução, incluindo fotocópia, só pode ser feita com autorização expressa da VISUALWORK.
FLASHWEBTRAINING.com, truquesedicas.com e respectivos logos são marcas registadas da VISUALWORK Produções Multimedia, Lda.