

## DUM č. 10 v sadě

### 28. Inf-4 Jednoduchá hra Had ve Flashi (ActionScript)

Autor: Robert Havlásek

Datum: 07.06.2013

Ročník: 5AV

Anotace DUMu: Flash - příklad: Had s tělem délky 5 dílků (zavedený statickým polem).\nPrvotní nakreslení těla pomocí duplikování objektů.\nPohyb hada (cyklus posouvající postupně všechny části těla).\nTestování, zda hlava nenarazila do těla.

Materiály jsou určeny pro bezplatné používání pro potřeby výuky a vzdělávání na všech typech škol a školských zařízení. Jakékoliv další využití podléhá autorskému zákonu.



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

## Had s tělem délky 5 dílků

Přistoupíme-li na podmínku důsledného rozlišování „který kód logicky patří hadovi a který celé ploše“, budeme muset hadovo tělo nejprve inicializovat (nastavit pole, vyrobit objekty). Principiálně nelze začít psát do Actions hlavy jen tak příkazy, podobně jako píšeme příkazy do Actions plochy – volně ležící příkazy v Actions plochy se po spušení animace provedou, zatímco v Actions hlavy skončí Flash chybovým hlášením:

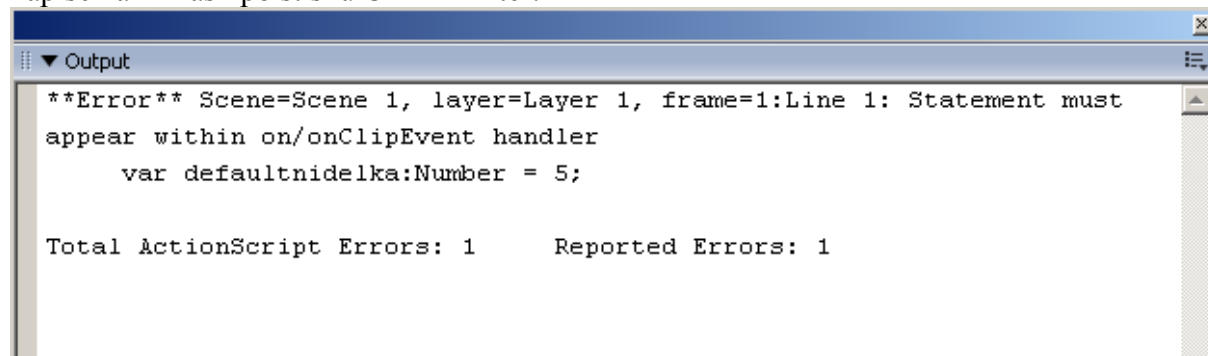
Například, zkusíme-li:

v objektu hlava:

```
var defaultnidelka:Number = 5;
on (keyPress "<Left>") {smer=2;}
on (keyPress "<Right>") {smer=0;}
on (keyPress "<Up>") {smer=1;}
on (keyPress "<Down>") {smer=3;}

onClipEvent (enterFrame)
{
  ...
}
```

napiše nám Flash po stisku CTRL+Enter:



Je to logické, kód zařazený v objektech se provádí až na základě nějaké události, na niž je třeba reagovat (pomocí on).


Naštěstí existuje událost onClipEvent (load), která se uděje jednou, na začátku, hned poté, co se klip naložuje (=nahraje do paměti a začne pracovat).

Její kód budeme psát dolů pod funkci onClipEvent.

## Inicializace polí, duplikace článků těla

Zavedeme pole jménem telox a teloy, v nichž si budeme pamatovat pozice jednotlivých článků těla. Dále zavedeme proměnnou defaultnidelka, v níž budiž uvedeno, jak velký je had na začátku hry. Pro konkrétní délku hada v průběhu hry postačí dotaz telox.length, na to zvláštní proměnnou nepotřebujeme.

```
onClipEvent (load) {
  var telox:Array = Array();
  var teloy:Array = Array();
  var defaultnidelka:Number = 5;
}
```

Kdekoliv v ploše nakreslíme (nástrojem ) novou modrou kuličku o rozměrech 10×10, převedeme ji do symbolu typu MovieClip, třída nechť má libovolné jméno, instanci pojmenujeme telovzor. Její postupnou duplikací bude vznikat tělo hada.

Otázka pro studenty: Na kterém místě byste vyrobili tělo hada, který stojí v základní pozici? Názory se mohou různit, někteří by očekávali, že had bude mít viditelné tělo jdoucí zleva (resp. zprava), někteří by tělo nakreslili „pod hlavu“, tedy všechny jeho články na pozici [150, 100]. Lze též nakreslit tělo mimo obrazovku (např. na pozici [600, 100]).

Není možné tělo nekreslit vůbec, objekty potřebujeme překreslovat (mazat a znovu vyrábět), i když se had momentálně nehýbe.

Asi nejrozumnější je nakreslit články těla pod hlavu; na konci tohoto DUMu budeme vědět, že (bude-li hlava stát na místě) by tam stejně po 5 framech animace skončily.

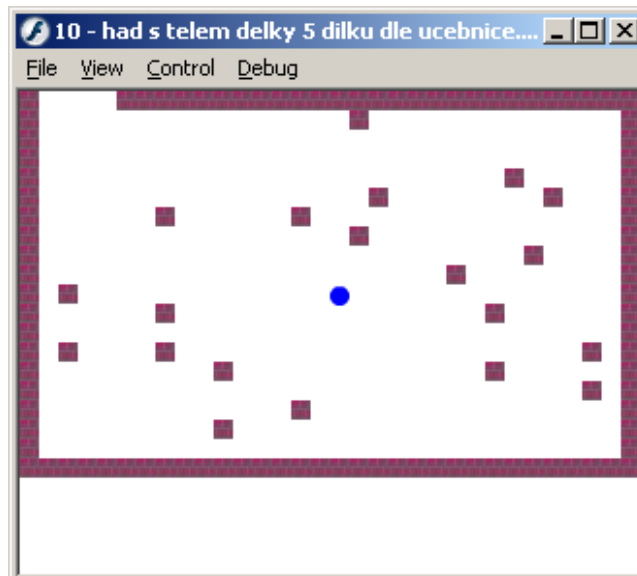
Tedy:

```
onClipEvent(load) {
    var telox:Array = Array();
    var teloy:Array = Array();
    var defaultnidelka:Number = 5;
    _root.telovzor._x=_root.hlava._x; //posuneme vzor do pozice hlavy
    _root.telovzor._y=_root.hlava._y;
    for (var i=0; i<defaultnidelka; i++) //zduplikujeme (defaultnidelka)-krat
    {telox.push(_root.hlava._x);
      teloy.push(_root.hlava._y);
      duplicateMovieClip(_root.telovzor,"klontelo["+i+"]",i)
    } //konec cyklu for
} // konec onClipEvent(load)
```

Povšimněte si zásadní změny v posledním argumentu funkce `duplicateMovieClip` – nové duplikáty nevyrábíme do `getNextHighestDepth()`, ale do hloubky `i`, tedy počínaje nejnižší hloubkou 0 až po hloubku 5.

To je zásadní změna. Pokud bychom tělo vyráběli do nejvyšší možné hloubky, riskujeme, že ve Flashi ona hloubka přesáhne „únosnou mez“ a Flash nám posléze odmítne objekty mazat – viz bug zmiňovaný v DUMu č. 8 v kapitole „Výmaz klonu“ (str. 2, ř. 16).

V této souvislosti musíme udělat ještě jednu změnu – zdí se nesmějí tvořit od hloubky 0, ale od vyšší hloubky, třeba 50. Pokud bychom je nechali tak, jak jsou, při prvním výmazu těla hada by se pět prvních zdí smazalo s ním:



Je to nepříjemné, co vše kvůli bugu musí programátor v kódu změnit. Upravíme v Actions plochy v tvorbě zdí v části // FYZICKE KLONOVANI ZDI OKOLO (řádek 13) duplikaci


```
Z duplicateMovieClip(_root.vzorzdi,"klonzdi["+k+"]",getNextHighestDepth());}
```

```
na duplicateMovieClip(_root.vzorzdi,"klonzdi["+k+"]",50+getNextHighestDepth());}
```

*Pozn.: Není to úplně „čistý způsob“, první zed' vyrobíme v hloubce 50, druhou v hloubce 100, třetí v hloubce 150, čistší by bylo vyrobit první zed' zvlášť a další v `getNextHighestDepth()`.*

## Posouvání hada – výpočet nového těla

*Pedagogická poznámka: Následuje dle mého soudu logicky nejtěžší část celého kódu hry – posun těla hada. Obvykle nejdřív celého pětičlankového hada nakreslím na tabuli (např. jdoucího zprava doleva. Hlava má pozici [telox[0], teloy[0]], jednotlivé články pak [telox[1], teloy[1]], atd., až do indexu 4. Mimo jiné to rozbije „opisovací způsob hodiny“, studenti se uvolní, odpočnou, zamyslí...*

Rozebereme způsob, jak s hadem pohnout. Jedině tak studenti zjistí, že nelze hadem hýbat od hlavy – posuneme-li hlavou o 10 doleva, ztratíme informaci, kde původně ta hlava byla. („Ale pane profesore, vždyť to je jasný, to se dá dopočítat, byla o 10 vpravo!“ „Dobře, a když s hlavou posunete o 10 nahoru, tak byla předtím kde – níž nebo vpravo?  To už se neví...“)

Jediná informace, kterou jsme ochotni při posouvání ztratit, je pozice bývalého ocásku – v další fázi hry nás nezajímá, odkud jsme přijeli. Cyklus tedy musí poslednímu článku předat pozici předposledního, pak předposlednímu předat pozici předpředposledního, atd. Přiřazení budou (v našem případě) čtyři. Nultý článek dostane (až po skončení cyklu) pozici zvlášť od objektu `_root.hlava`, jenž jsme na začátku `onClipEvent(enterFrame)` posunuli podle směru pohybu.

Celý kód:

```
// VYPOCET NOVEHO TELA:
for (m=telox.length-2;m>=0;m--){telox[m+1]=telox[m];
                                teloy[m+1]=teloy[m];}

telox[0]=_root.hlava._x;
teloy[0]=_root.hlava._y;
```

přidáme na konec `onClipEvent(enterFrame)`.

## Znovunakreslení těla

V polích `telox` a `teloy` máme nově vypočítané pozice, pojdme tedy staré objekty smazat a nové naklonovat. Tento postup není nutný, ve většině případů by stačilo pozice objektů prostě posunout, ale na druhou stranu je univerzálnější vzhledem k možnosti plynulých změn délky těla (prodlužování při sněžení jablka, zkrácení při smrti, atp.)

```
// KRESLENI TELA:
for (var m=0;m<telox.length;m++)
{removeMovieClip("klontelo["+m+"]");
 _root.telovzor._x=telox[m];
 _root.telovzor._y=teloy[m];
 duplicateMovieClip(_root.telovzor,"klontelo["+m+"]",m)
}
```

Kód přidáme rovněž na konec `onClipEvent(enterFrame)`.

Přichází krásná chvíle: Studenti, vyzkoušejte si, jestli se had pohybuje a tělo tahá správně za sebou. ☺ Vyzkoušejte všechny varianty, včetně otočky do protipohybu a naražení do zdi (to je poněkud komické, had do zdi „doběhne“ v dalších čtyřech framech...)

## Smrt ve zdi – úprava

Aby had při naražení do zdi nedobíhal další čtyři framy, než se jeho tělo zformuje pod stojící hlavu, musíme v Actions plochy na konec funkce `smrt()` přidat cyklus, který všechny části těla nastaví pod hlavu:

```
for (var m=0;m<_root.hlava.telox.length;m++)
{ _root.hlava.telox[m]=_root.hlava._x;
  _root.hlava.teloy[m]=_root.hlava._y;
}
```

## Nekousnul jsem se?

Test, zda se had nekousne do vlastního těla, je nevhodnější napsat přesně do míst, kde testujeme naražení do zdi – mezi `var obsazeno=0` a `if (obsazeno==1)`... – proměnnou `obsazeno` lze s výhodou použít, v principu je jedno, narazíme-li do zdi nebo sami do sebe...

Porovnáváme přitom `_root.hlava._x` a `_root.hlava._y` s prvky polí `telox` a `teloy` (které ovšem patří hlavě, musíme jej tedy adresovat `_root.hlava.telox` a `_root.hlava.teloy`).

Samotné porovnávání:

```
// A I KONTROLA NARAZENI DO SVEHO TELA (JEN POKUD SE HYBEME):
if (smer<4) for (m=0; m<_root.hlava.telox.length; m=m+1)
    if ((_root.hlava._x==_root.hlava.telox[m])&&(_root.hlava._y==_root.hlava.teloy[m]))
        {obsazeno=1;break;}
```

Studenti, otestujte kód. Pro kvalitní test, jestli se opravdu kousne až do ocásku, si pokusně zvyšte proměnnou `defaultnidelka` na cca 14. Jak vidíte, nelze ocáskem „na poslední chvíli uhnout“ – pokud budeme chtít mít možnost na poslední chvíli uhnout, nebudeme pole testovat až do `m<_root.hlava.telox.length`, ale jen do `m<_root.hlava.telox.length-1`.

Kompletní výpis kódu na závěr:

```
var zdix:Array = Array();
var zdiy:Array = Array();
var klonzdi:Array = Array();
// TVORBA POLE ZDI OKOLO:
for (i=0; i<=310; i=i+10) {zdix.push(i);zdiy.push(0);} // horni rada
for (i=0; i<=310; i=i+10) {zdix.push(i);zdiy.push(190);} // spodni rada
for (i=0; i<=190; i=i+10) {zdix.push(0);zdiy.push(i);} // leva rada
for (i=0; i<=190; i=i+10) {zdix.push(310);zdiy.push(i);} // prava rada
// FYZICKE KLONOVANI ZDI OKOLO:
for (k=0; k<zdix.length; k++){
    _root.vzorzdi._x=zdix[k];
    _root.vzorzdi._y=zdiy[k];
    duplicateMovieClip(_root.vzorzdi,"klonzdi["+k+"]",50+getNextHighestDepth());}
// NAHODNE ZDI A JEJICH KONTROLA:
for (k=0; k<20; k=k+1){
    _root.vzorzdi._x=10+10*random(30);
    _root.vzorzdi._y=10+10*random(18);
    // KONTROLA OBSAZENOSTI:
    var obsazeno=0;
    for (m=0; m<zdix.length; m=m+1)
        if ((_root.vzorzdi._x==zdix[m])&&(_root.vzorzdi._y==zdiy[m]))
            {obsazeno=1;break;}
    if ( (obsazeno==1) or
        ((_root.vzorzdi._x==160)&&(_root.vzorzdi._y==100)) ) {k=k-1}
    else {zdix.push(_root.vzorzdi._x);
        zdiy.push(_root.vzorzdi._y);
        duplicateMovieClip(_root.vzorzdi,"klonzdi["+zdix.length-1+"]",getNextHighestDepth());}
}

function smrt() {
    _root.hlava._x=160;
    _root.hlava._y=100;
    _root.hlava.smer=4;
    for (var m=0;m<_root.hlava.telox.length;m++)
        {_root.hlava.telox[m]=_root.hlava._x;
        _root.hlava.teloy[m]=_root.hlava._y;
        }
}
```

v ploše:

v objektu hlava:

```
on (keyPress "<Left>") {smer=2;}
on (keyPress "<Right>") {smer=0;}
on (keyPress "<Up>") {smer=1;}
on (keyPress "<Down>") {smer=3;}

onClipEvent(enterFrame)
{ if (smer==0) {_root.hlava._x = _root.hlava._x + 10;}
  if (smer==1) {_root.hlava._y = _root.hlava._y - 10;}
  if (smer==2) {_root.hlava._x = _root.hlava._x - 10;}
  if (smer==3) {_root.hlava._y = _root.hlava._y + 10;}
  if (_root.hlava._x>=320) {_root.hlava._x=0;}
  if (_root.hlava._y>=200) {_root.hlava._y=0;}
  if (_root.hlava._x<0) {_root.hlava._x=310;}
  if (_root.hlava._y<0) {_root.hlava._y=190;}
  // KONTROLA NARAZENI HLAVY DO ZDI:
  var obsazeno=0;
  for (m=0; m<_root.zdix.length; m=m+1)
    if ((_root.hlava._x==_root.zdix[m])&&(_root.hlava._y==_root.zdiy[m]))
      {obsazeno=1;break;}
  // A I KONTROLA NARAZENI DO SVEHO TELA (JEN POKUD SE HYBEME):
  if (smer<4) for (m=0; m<_root.hlava.telox.length; m=m+1)
    if
  ((_root.hlava._x==_root.hlava.telox[m])&&(_root.hlava._y==_root.hlava.teloy
  [m]))
    {obsazeno=1;break;}
  if (obsazeno==1) _root.smrt();
/*   if ((_root.hlava._x>=320) || (_root.hlava._y>=200) ||
      (_root.hlava._x<0) || (_root.hlava._y<0)) _root.smrt();*/
  // VYPOCET NOVEHO TELA:
  for (m=telox.length-2;m>=0;m--) {telox[m+1]=telox[m];
                                  teloy[m+1]=teloy[m];}

  telox[0]=_root.hlava._x;
  teloy[0]=_root.hlava._y;
  // KRESLENI TELA:
  for (var m=0;m<telox.length;m++)
    {removeMovieClip("klontelo["+m+"]");
      _root.telovzor._x=telox[m];
      _root.telovzor._y=teloy[m];
      duplicateMovieClip(_root.telovzor,"klontelo["+m+"]",m)
    }
} // konec onClipEvent(enterFrame)

onClipEvent(load) {
  var telox:Array = Array();
  var teloy:Array = Array();
  var defaultnidelka:Number = 14;
  _root.telovzor._x=_root.hlava._x; //posuneme vzor do pozice hlavy
  _root.telovzor._y=_root.hlava._y;
  for (var i=0; i<defaultnidelka; i++) //zduplikujeme (defaultnidelka)-krat
    {telox.push(_root.hlava._x);
      teloy.push(_root.hlava._y);
      duplicateMovieClip(_root.telovzor,"klontelo"+i+"",i)
    } //konec cyklu for
} // konec onClipEvent(load)
```

*Pedagogická poznámka: Tento DUM obvykle za jednu vyučovací jednotku nestihneme.*