

```

/*****
        hlavicky.h - description
        -----
begin          : So kvì 15 2004
copyright     : (C) 2004 by Petr Kavanek
email        : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

#ifndef HLAVICKY_H
#define HLAVICKY_H

// tento hlavickovy soubor slouzi k zjednoduseni prace s includovanim
// hlavicek ClanLibu

#include <ClanLib/core.h>
#include <ClanLib/application.h>
#include <ClanLib/gl.h>
#include <ClanLib/display.h>

// vytvorime ukazatel na SpravceZdroju ve jmennem prostoru Konfigurace,
// aby k nemu bylo mozne pristupovat odkudkoliv, coz bude treba
namespace Konfigurace {

    // ukazatel na SpravceZdroju
    // bude nutne jej vytvorit po inicializaci knihovny
    extern CL_ResourceManager* SpravceZdroju;

};

#endif

/*****
        t_aplikace.h - description
        -----
begin          : So kv? 15 2004
copyright     : (C) 2004 by Petr Kavanek
email        : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

#ifndef T_APLIKACE_H
#define T_APLIKACE_H

// hlavickove souhory ClanLibu
#include "hlavicky.h"

/*****
*
*   T_Aplikace
*
*****/

```

```

*****/

/**Potomek CL_ClanApplication nutny pro fungovani knihovny ClanLib.
  Bude nutne vytvorit jednu globalni instanci teto tridy!!!
  Nikde v programu nebudeme deklarovat klasiclou funkci main()
  nebo WinMain()!!!
  *@author Petr Kavanek
  */

class T_Aplikace : public CL_ClanApplication {

// verejne metody:
public:

    // Zastupuje klasickou funkci main. Prebira parametry z prikazoveho radku.
    virtual int main(int argc, char** argv);

}; // T_Aplikace *****

#endif

/*****
                t_aplikace.cpp - description
                -----
begin           : So kv? 15 2004
copyright      : (C) 2004 by Petr Kavanek
email         : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*
*****/

#include "t_aplikace.h"
#include "t_hra.h"

#include <iostream>
#include <string>

// nezbytna globalni instance teto tridy
T_Aplikace Aplikace;

// inicializujeme SpravceZroju na nulu
CL_ResourceManager* Konfigurace::SpravceZdroju = 0;

//-----
// main()
//-----
int T_Aplikace::main(int argc, char** argv) {

    using namespace std;
    using namespace Konfigurace;

    // vytvorime okno konzole kvuli textovemu vystupu jako
    // jsou zpravy o chybach
    CL_ConsoleWindow Console("Konzole");

```



```

*   This program is free software; you can redistribute it and/or modify   *
*   it under the terms of the GNU General Public License as published by   *
*   the Free Software Foundation; either version 2 of the License, or     *
*   (at your option) any later version.                                   *
*                                                                 *
*****/

#ifndef T_HRA_H
#define T_HRA_H

#include "hlavicky.h"
#include "t_neplatnymodexception.h"
#include "t_modhry.h"

#include <string>
#include <list>

/*****
*
*                               T_Hra
*
*****/

/**Metoda Run() teto tridy bude obstaravat vlastni
chod nasi hry. Jeji instance bude vytvorena
v T_Aplikace::main() odkud bude volana i metoda Run()

    *@author Petr Kavanek
    */

class T_Hra {

// typy:
public:

    // typ SeznamuModu
    typedef std::list<std::string> TSeznamModu;

// konstrukce a destrukce:
public:

    T_Hra();

    ~T_Hra();

// verejne metody:
public:

    // vlastni beh nasi hry
    void Run();

    // nastaveni noveho modu
    void NastavMod(std::string Jmeno);

// pristupove funkce
public:

    // vrati jmeno modu, který je nastaven jako aktivni
    std::string VratJmenoAktModu() const;

    // vrati konstantni referenci na seznam modu
    const TSeznamModu& VratSeznamModu() const;

// implementace:
private:

    // vytvori seznam vseh pripustnych modu
    void NactiSeznamModu();

```

```

// soukrome atributy
private:

// seznam vseh pripustnych modu
TSeznamModu SeznamModu;

// iterator na mod nastaveny jako aktivni v SeznamModu
typedef TSeznamModu::iterator TSeznamModuIt;
TSeznamModuIt AktMod;

// vlastni aktivni mod
T_ModHry AktivniMod;
}; /*****

#endif

/*****
                t_hra.cpp - description
                -----
begin           : So kv? 15 2004
copyright      : (C) 2004 by Petr Kavanek
email         : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

#include "t_hra.h"

#include <algorithm>

//-----
// konstruktor
//-----
T_Hra::T_Hra(){

    using namespace std;
    using namespace Konfigurace;

    // vytvorime seznam pripustnych modu
    NactiSeznamModu();

    // zjistime si jaky mod ma byt jako prvn? nastaven jako aktivni
    string UvodniMod = CL_String::load("T_Hra/UvodniMod", SpravceZdroju);

    // nastavime UvodniMod jako aktivni
    NastavMod(UvodniMod);

} // Konstruktor -----

//-----
// destruktork
//-----

```

```

T_Hra::~T_Hra(){

} // destruktor -----

//-----
// NactiSeznamModu()
//-----
void T_Hra::NactiSeznamModu() {

    using namespace std;
    using namespace Konfigurace;

    // nejprve smazeme aktualni seznam
    SeznamModu.clear();

    // vytvorime seznam vseh jmen polozek typu ModHry
    // v sekci T_Hra ve zdroje.xml
    typedef list<string> TTmpList;
    TTmpList TmpList = SpravceZdroju->get_resources_of_type("ModHry", "T_Hra");

    // vytvorime seznam jmen pripustnych modu
    typedef TTmpList::iterator TTmpListIt;

    TTmpListIt eIt = TmpList.end();
    for (TTmpListIt It = TmpList.begin(); It != eIt; ++It) {

        // vytvorime mod prislusneho nazvu
        CL_Resource ModHry = SpravceZdroju->get_resource(*It);

        // nahrajeme jeho data
        ModHry.load();

        // nyní jiz muzeme pristupovat k jeho slozkam pres CL_DomElement
        CL_DomElement Element = ModHry.get_element();

        // zjistime tedy jeho jmeno
        string JmenoModu = Element.get_attribute("name");

        // toto jmeno pridame do vytvareneho seznamu
        SeznamModu.push_back(JmenoModu);

    }

} // NactiSeznamModu() -----

//-----
// NastavMod()
//-----
void T_Hra::NastavMod(std::string Jmeno) {

    using namespace std;

    TSeznamModuIt Pozice = find(SeznamModu.begin(), SeznamModu.end(), Jmeno);

    if (Pozice == SeznamModu.end()) {

        throw T_NeplatnyModException();
    }
}

```

```

    return;
}

AktMod = Pozice;

// zde se nemusime starat o uvolnovani pameti
AktivniMod = T_ModHry(*AktMod);
} // NastavMod() -----

//-----
// VratJmenoAktModu()
//-----
std::string T_Hra::VratJmenoAktModu() const {

    return *AktMod;
} // VratJmenoAktModu() -----

//-----
// VratSeznamModu()
//-----
const T_Hra::TSeznamModu& T_Hra::VratSeznamModu() const {

    // vytvorime referenci a tu pak vratime
    const TSeznamModu& Navrat = SeznamModu;

    return Navrat;
} // VratSeznamModu() -----

//-----
// Run()
//-----
void T_Hra::Run() {

    // pockame si na stistení klavesy escape
    while (! CL_Keyboard::get_keycode(CL_KEY_ESCAPE)) {

        // invariant: Dosud nebyl stisten escape pri get_keycode()

        // aktualizujeme logiku aktivního modu
        AktivniMod.AktualizujSe();

        // vykreslime aktivni mod
        AktivniMod.VykresliSe();

        // zobrazime nakreslene zmeny (prehození predního a zadního bufferu)
        CL_Display::flip();

        // uspime aplikaci na deset milisekund, abychom zbytecne
        // neblokovali procesor

```

```

const int DobaObnoveni = 10;
CL_System::sleep(DobaObnoveni);

// probudime aplikaci resp. knihovnu aby byla aktualni
CL_System::keep_alive();

} // while

} // Run() -----

/*****
                t_modhry.h - description
                -----
begin           : Po kv? 17 2004
copyright      : (C) 2004 by Petr Kavanek
email         : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

#ifndef T_MODHRY_H
#define T_MODHRY_H

#include <string>

#include "hlavicky.h"
#include "t_modhryzaklad.h"

/*****
*
*                               T_ModHry
*****/

/**Zakladni abstraktni trida predstavujici rozhrani jednotlivych
modu hry.

    *@author Petr Kavanek
    */

class T_ModHry {

// konstrukce a destrukce:
public:

    // konstruktor - vytvori prazdny mod
    T_ModHry();

    // konstruktor - parametrem nazev konkretniho modu
    T_ModHry(std::string JmenoModu);

// verejne metody
public:

    void AktualizujSe();
    void VykresliSe();

// implementace:

```



```

private:

    // ukazatel na implementacni tridu
    CL_SharedPtr<T_ModHryZaklad> Impl;

}; /*****

#endif

/*****
        t_modhry.cpp - description
        -----
begin          : Po kv? 17 2004
copyright      : (C) 2004 by Petr Kavanek
email          : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

#include "t_modhry.h"

#include "t_menu.h"
#include "t_bitva.h"
#include "t_neplatnymodexception.h"

//-----
// konstruktor
//-----
T_ModHry::T_ModHry()
: Impl(0) {

} // konstruktor -----

//-----
// konstruktor
//-----
T_ModHry::T_ModHry(std::string JmenoModu)
{
    // na zaklade jmena se rozhodneme, jaky konkretni mod vytvorime
    if (JmenoModu == "Menu") {

        Impl = new T_Menu();

        return;

    }

    if (JmenoModu == "Bitva") {

        Impl = new T_Bitva();

        return;

    }
}

```

```

// pokud dojdeme az sem, znamena to, ze Jmeno nezname
// vyyvolame proto vyjimku
throw T_NeplatnyModException();

} // konstruktor -----

//-----
// AktualizujSe()
//-----
void T_ModHry::AktualizujSe() {

    if (Impl) {

        Impl->AktualizujSe();

    }
    else {

        throw T_NeplatnyModException();

    }

} // AktualizujSe() -----

//-----
// VykresliSe()
//-----
void T_ModHry::VykresliSe() {

    if (Impl) {

        Impl->VykresliSe();

    }
    else {

        throw T_NeplatnyModException();

    }

} // VykresliSe() -----

/*****
                                t_modhryzaklad.h - description
                                -----
begin                          : ?t kv? 20 2004
copyright                       : (C) 2004 by Petr Kavanek
email                           : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

#endif T_MODHRYZAKLAD_H

```

```

#define T_MODHRYZAKLAD_H

#include <string>

#include "hlavicky.h"

/*****
*
*                               T_ModHryZaklad                               *
*****/

/**Implementacni abstraktni trida modu hry.

    *@author Petr Kavanek
    */

class T_ModHryZaklad {

    friend class T_Menu;
    friend class T_Bitva;
    friend class T_ModHry;

// destrukce
public:

    virtual ~T_ModHryZaklad() {};

// konstrukce
private:

    // konstruktor
    T_ModHryZaklad() {};

// rozhrani:
private:

    virtual void AktualizujSe() = 0;
    virtual void VykresliSe() = 0;

}; /*****/

#endif

/*****
*                               t_modhryzaklad.cpp - description
*                               -----
*
* begin                          : ?t kv? 20 2004
* copyright                       : (C) 2004 by Petr Kavanek
* email                           : pietrotk@seznam.cz
*****/

/*****
*
* This program is free software; you can redistribute it and/or modify
* it under the terms of the GNU General Public License as published by
* the Free Software Foundation; either version 2 of the License, or
* (at your option) any later version.
*
* *****/

// #include "t_modhryzaklad.h"

//-----
// konstruktor
//-----
// T_ModHryZaklad::T_ModHryZaklad() {

```

```

//} // konstruktor -----

//-----
// destruktor
//-----
//T_ModHryZaklad::~T_ModHryZaklad(){

//} // destruktor -----

/*****
                t_neplatnymodexception.h - description
                -----
begin           : Ne kv? 16 2004
copyright      : (C) 2004 by Petr Kavanek
email         : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

#ifndef T_NEPLATNYMODEXCEPTION_H
#define T_NEPLATNYMODEXCEPTION_H

#include <stdexcept>

/*****
*
*                               T_NeplatnyModException
*****/

/**Vyjimka vyvolana pokusem o nastaveni neplatneho modu.

    *@author Petr Kavanek
    */

class T_NeplatnyModException : public std::runtime_error {
public:

    T_NeplatnyModException() : runtime_error("Neplatny mod hry!!!") {};
}; /*****/

#endif

/*****
                t_menu.h - description
                -----
begin           : ?t kv? 20 2004
copyright      : (C) 2004 by Petr Kavanek
email         : pietrotk@seznam.cz
*****/

/*****
*
*   This program is free software; you can redistribute it and/or modify
*
*****/

```



```

*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

#include "t_menu.h"

#include <iostream>

//-----
// konstruktor
//-----
T_Menu::T_Menu()

// konstrukce pozadi
: Pozadi("Obrazky/Pozadi", Konfigurace::SpravceZdroju) {

} // destruktor -----

//-----
// destruktor
//-----
T_Menu::~T_Menu(){

// std::cout << "T_Menu destruktor..." << std::endl;

} // destruktor -----

//-----
// AktualizujSe()
//-----
void T_Menu::AktualizujSe() {

} // AktualizujSe() -----

//-----
// VykresliSe()
//-----
void T_Menu::VykresliSe() {

// vykreslime pozadi
Pozadi.draw();

} // VykresliSe() -----

/*****
t_bitva.h - description
-----

```

```

begin          : ?t kv? 20 2004
copyright     : (C) 2004 by Petr Kavanek
email        : pietrotk@seznam.cz
*****

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****

#ifndef T_BITVA_H
#define T_BITVA_H

#include <t_modhryzaklad.h>

/*****
*
*                               T_Bitva
*****

/**Vlastni imlementace bitvy.

    *@author Petr Kavanek
    */

class T_Bitva : public T_ModHryZaklad {

    friend class T_ModHryZaklad;
    friend class T_ModHry;

// konstrukce a destrukce
private:
    T_Bitva();
    ~T_Bitva();

// metody:
private:

    virtual void AktualizujSe() {};
    virtual void VykresliSe() {};

}; /*****

#endif

/*****
*                               t_bitva.cpp - description
*                               -----
begin          : ?t kv? 20 2004
copyright     : (C) 2004 by Petr Kavanek
email        : pietrotk@seznam.cz
*****

/*****
*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****

```

```

#include "t_bitva.h"

#include <iostream>

T_Bitva::T_Bitva(){

//  std::cout << "T_Bitva konstruktor..." << std::endl;

}

T_Bitva::~T_Bitva(){

//  std::cout << "T_Bitva destruktork..." << std::endl;

}

*****zdroje/zdroje.xml*****
<resources>

  <section name="Obrazky">

    <surface      name="Pozadi"      file="obrazky/pozadi/pozadi.png"/>

  </section>

  <section name="T_Hra">

    <string      name="UvodniMod"    value="Menu"/>

    <ModHry      name="Menu"/>
    <ModHry      name="Bitva"/>

  </section>

  <section name="HlavniOkno">

    <string      name="Titulek"      value="Onion Bomberman"/>
    <integer     name="SirkaOkna"    value="660"/>
    <integer     name="VyskaOkna"    value="540"/>
    <boolean     name="FullScreen"   value="false"/>

  </section>

</resources>

```