#include <iostream>

using namespace std;

//Koristene skracenice u komentarima

// dflt. = default

// user-def. = user-defined (korisnicki-definirano)

// ctor = constructor (konstruktor)

// copy ctor = copy constructor (konstruktor kopije)

// move ctor = move constructor (konstruktor premjestanja)

//Z1.0

char\* AlocirajIKopiraj(const char\* tekst);

int Min(int a, int b);

int Max(int a, int b);

class Datum

{

private:

int\* \_dan;

int\* \_mjesec;

int\* \_godina;

public:

//Z1.1 Dflt. ctor [Postaviti na dflt. vrijednosti]

Datum();

//Z1.2 User-def. ctor

Datum(int d, int m, int g);

//Z1.3 Copy ctor

Datum(const Datum& obj);

//Z1.4 Move ctor

Datum(Datum&& obj);

//Z1.5

int GetDan() const;

int GetMjesec() const;

int GetGodina() const;

//Z1.6

void SetDan(int dan);

void SetMjesec(int mjesec);

void SetGodina(int godina);

//Z1.7

void Ispis();

//Z1.8

~Datum();

};

class Glumac

{

private:

char\* \_ime;

char\* \_prezime;

char\* \_zemljaPorijekla;

Datum\* \_datumRodjenja;

bool\* \_spol; //1-Muski, 0-Zenski

public:

//Z2.0 Dflt. ctor

Glumac();

//Z2.1 User-def. ctor

Glumac(const char\* ime, const char\* prez, const char\* zemlja, int d, int m, int g, bool spol);

//Z2.2 Copy ctor

Glumac(const Glumac& obj);

//Z2.3 Move ctor

Glumac(Glumac&& obj);

//Z2.4

char\* GetIme() const;

char\* GetPrezime() const;

char\* GetZemljaPorijekla() const;

Datum GetDatumRodjenja() const;

bool GetSpol() const;

//Z2.5

void SetIme(const char\* ime);

void SetPrezime(const char\* prezime);

void SetZemljaPorijekla(const char\* zemlja);

void SetDatumRodjenja(Datum datumRodjenja);

void SetSpol(bool spol);

//Z2.6

void Ispis();

//Z2.7

~Glumac();

};

class Epizoda {

private:

char\* \_naziv;

int\* \_trajanje; //u minutama

char \_kratakSadrzaj[100];

Datum \_datumPremijere;

int \_maxBrojOcjena; // velicina niza

int \_trenutnoOcjena; //brojac

int\* \_ocjene;

public:

//Z3.0

//Settovati vrijednosti na defaultne

Epizoda();

//Z3.1

Epizoda(const char\* naziv, int trajanje, const char\* kratakOpis, Datum datum, int ukupnoOcjena);

//Z3.2

Epizoda(const Epizoda& obj);

//Z3.3

Epizoda(Epizoda&& obj);

//Z3.4

char\* GetNaziv() const;

int GetTrajanje() const;

const char\* GetKratakSadrzaj() const;

Datum GetDatumPremijere() const;

int GetTrenutnoOcjena() const;

int GetMaxBrojOcjena() const;

//Z3.5

//Vratiti vrijednost na lokaciji specificiranoj ulaznim parametrom 'index'

//Ukoliko je proslijedjena nevalidna vrijednost, potrebno je vratiti element niza sa najblizim validnim indeksom

//Npr. Ako je index = -5, vratiti vrijednost \_ocjene na indexu 0; Isto vazi i za indekse vece od vrijednosti brojaca '\_trenutnoOcjena'.

int GetOcjena(int index) const;

//Z3.6

void SetNaziv(const char\* naziv);

void SetTrajanje(int trajanje);

void SetKratakSadrzaj(const char\* kratakSadrzaj);

void SetDatumPremijere(Datum datum);

//Z3.7

//Prosiriti niza \_ocjene na sljedeci nacin:

// \*Konstruisati novi niz velicine [\_maxBrojOcjena + prosiriZa]

// \*Kopirati vrijednosti iz starog niza u novi niz

// \*Dealocirati stari niz

void ProsiriNizOcjena(int prosiriZa);

//Z3.8

//Ukoliko je brojac dosao do kraja (jednak velicini niza), uraditi prosirivanje niza za 10 elemenata;

void DodajOcjenu(int ocjena);

//Z3.9

bool UkloniZadnjuOcjenu();

//Z3.10

float GetProsjecnaOcjena();

//Z3.11

void Ispis();

//Z3.12

~Epizoda();

};

class Uloga {

private:

Glumac\* \_glumac;

char\* \_opis;

char\* \_tipUloge; //Glavna, sporedna, epizodna, statista, gostujuca zvijezda, cameo ...

public:

//Z4.0

Uloga();

//Z4.1

Uloga(Glumac& glumac, const char\* opis, const char\* tip);

//Z4.2

Uloga(const Uloga& obj);

//Z4.3

Uloga(Uloga&& obj);

//Z4.4

Glumac GetGlumac() const;

char\* GetOpis() const;

char\* GetTipUloge() const;

//Z4.5

void SetGlumac(Glumac glumac);

void SetOpis(const char\* opis);

void SetTipUloge(const char\* tipUloge);

//Z4.6

void Ispis();

//Z4.7

~Uloga();

};

class Serija {

private:

char\* \_naziv;

int \_trenutnoUloga;

Uloga\* \_uloge[50] = { nullptr }; //Svi elementi (pokazivaci) se postavljaju na NULL

int \_maxBrojEpizoda;

int \_trenutnoEpizoda;

Epizoda\* \_epizode;

public:

//Z5.0

Serija();

//Z5.1

Serija(const char\* naziv, int maxBrojEpizoda);

//Z5.2

Serija(const Serija& obj);

//Z5.3

bool DodajUlogu(Uloga& uloga);

//Z5.4

bool DodajEpizodu(Epizoda& ep);

//Z5.5

Epizoda\* GetNajboljeOcijenjenaEpizoda();

//Z5.6 :: Pored ostalih atributa ispisati i sve uloge i sve epizode

void Ispis();

//Z5.7

~Serija();

};

void Zadatak1() {

cout << "Testiranje klase 'Datum'\n\n";

Datum novaGodina; //Def. ctor

novaGodina.SetDan(1);

novaGodina.SetMjesec(1);

novaGodina.SetGodina(2021);

novaGodina.Ispis();

cout << endl;

//

Datum prviFebruar(novaGodina.GetDan(), novaGodina.GetMjesec() + 1, novaGodina.GetGodina());

prviFebruar.Ispis();

cout << endl;

Datum prviMart(1, 3, 2021); //User-def. ctor

prviMart.Ispis();

cout << endl;

Datum danSale(prviMart); //Copy ctor

danSale.SetMjesec(4);

danSale.Ispis();

cout << endl;

Datum praznikRada(move(danSale)); //Move ctor

praznikRada.SetMjesec(5);

praznikRada.Ispis();

cout << endl;

cout << "Dealokacija ..." << endl;

}

void Zadatak2() {

cout << "Testiranje klase 'Glumac'\n\n";

Glumac ryanGosling; //Def. ctor

ryanGosling.SetIme("Ryan");

ryanGosling.SetPrezime("Gosling");

ryanGosling.SetSpol(1);

ryanGosling.SetDatumRodjenja(Datum(1, 1, 1980));

ryanGosling.SetZemljaPorijekla("Kanada");

ryanGosling.Ispis();

cout << endl;

//

Glumac harrisonFord("Harrison", "Ford", "SAD", 2, 2, 1955, 1); //User-def. ctor

Glumac michellePfeifer("Michelle", "Pfeiffer", "SAD", 3, 3, 1966, 0); //User-def. ctor

harrisonFord.Ispis();

cout << endl;

michellePfeifer.Ispis();

cout << endl;

Glumac jackNicholson(harrisonFord); // copy ctor

jackNicholson.SetIme("Jack");

jackNicholson.SetPrezime("Nicholson");

jackNicholson.SetDatumRodjenja(Datum(1, 4, 1945));

jackNicholson.Ispis();

cout << endl;

Glumac heathLedger(move(jackNicholson)); //move ctor

heathLedger.SetIme("Heath");

heathLedger.SetPrezime("Ledger");

heathLedger.SetDatumRodjenja(Datum(5, 3, 1983));

heathLedger.SetZemljaPorijekla("Australija");

heathLedger.Ispis();

cout << endl;

cout << "Dealokacija ..." << endl;

}

void Zadatak3() {

cout << "Testiranje klase 'Epizoda'\n\n";

Epizoda e1;

e1.SetNaziv("What's Cooking?");

e1.SetTrajanje(21);

e1.SetKratakSadrzaj("Bender decides to become a chef so ...");

e1.SetDatumPremijere(Datum(5, 5, 2021));

e1.Ispis();

cout << endl;

Epizoda e2("This Mission is Trash", 22, "Fry, Leela, and Bender travel to the garbage meteor and discover loads of discarded junk.", Datum(13, 5, 2021), 10);

e2.Ispis();

cout << endl;

Epizoda e3(e2);

e3.SetNaziv("Smell-o-Scope");

e3.SetTrajanje(20);

e3.SetKratakSadrzaj("Using Professor Farnsworth's Smell-o-Scope, Fry locates the stinkiest object in the universe.");

e3.SetDatumPremijere(Datum(21, 5, 2021));

e3.Ispis();

cout << endl;

Epizoda e4(move(e3));

e4.SetNaziv("Electric Drug");

e4.SetTrajanje(24);

e4.SetKratakSadrzaj("Bender's electricity addiction puts the Planet Express crew in danger");

e4.SetDatumPremijere(Datum(29, 5, 2021));

for (size\_t i = 0; i < 15; i++)

e4.DodajOcjenu(rand() % 10 + 1);

e4.UkloniZadnjuOcjenu();

e4.UkloniZadnjuOcjenu(); //Brisemo zadnje dvije ocjene

cout << endl;

e4.Ispis();

cout << "Dealokacija ..." << endl;

}

void Zadatak4() {

cout << "Testiranje klase 'Uloga'\n\n";

Glumac seanConnery("Sean", "Connery", "Velika Britanija", 25, 8, 1930, 1);

Glumac danielCraig("Daniel", "Craig", "Velika Britanija", 2, 3, 1968, 1);

Uloga jamesBond(seanConnery, "MI6 Detective James Bond ....", "Main role");

jamesBond.SetGlumac(danielCraig);

jamesBond.SetOpis("After earning 00 status and a licence to kill, Secret Agent James Bond sets out on his first mission as 007.");

jamesBond.SetTipUloge("Main role");

Uloga bond25(jamesBond);

Uloga bond26(move(bond25));

bond26.Ispis();

cout << "Dealokacija ..." << endl;

}

void Zadatak5() {

cout << "Testiranje klase 'Serija'\n\n";

Serija teorijaVelikogPraska("The Big Bang Theory", 200);

Glumac jimParsons("Jim", "Parsons", "SAD", 17, 7, 1967, 1);

Glumac johnnyGalecki("Johnny", "Galecki", "SAD", 15, 3, 1975, 1);

Glumac kaleyCuoco("Kaley", "Cuoco", "SAD", 13, 4, 1985, 0);

Uloga sheldonCooper(jimParsons, "Dr. Sheldon Cooper, a theoretical physicist at Caltech", "Series regular");

Uloga leonardHofstadter(johnnyGalecki, "Dr. Leonard Hofstadter, a experimental physicist at Caltech", "Series regular");

Uloga penny(kaleyCuoco, "Penny, a waitress at Cheesecake factory", "Series regular");

//Serija::Dodavanje uloga

teorijaVelikogPraska.DodajUlogu(sheldonCooper);

teorijaVelikogPraska.DodajUlogu(leonardHofstadter);

teorijaVelikogPraska.DodajUlogu(penny);

Epizoda E1("The Big Bran Hypothesis", 22, "When Sheldon and Leonard drop off a box of flat pack furniture...", Datum(1, 6, 2021), 100);

Epizoda E2("The Luminous Fish Effect", 21, "Sheldon is fired from his job as a physicist after insulting his new boss...", Datum(8, 6, 2021), 100);

Epizoda E3("The Bat Jar Conjecture", 22, "The guys decide to compete in a university quiz called physics bowl...", Datum(15, 6, 2021), 100);

Epizoda E4("The Nerdvana Annihilation", 21, "In an online auction, Leonard buys a full-sized replica of the time machine...", Datum(22, 6, 2021), 100);

//Epizoda::DodajOcjenu

int ocjene1[] = { 5,7,8 }, ocjene2[] = { 10,5,7,10,9 }, ocjene3[] = { 9,8,9,9 }, ocjene4[] = { 10,5,3,7,6,6 };

for (size\_t i = 0; i < size(ocjene1); i++)

E1.DodajOcjenu(ocjene1[i]);

for (size\_t i = 0; i < size(ocjene2); i++)

E2.DodajOcjenu(ocjene2[i]);

for (size\_t i = 0; i < size(ocjene3); i++)

E3.DodajOcjenu(ocjene3[i]);

for (size\_t i = 0; i < size(ocjene4); i++)

E4.DodajOcjenu(ocjene4[i]);

//Serija::DodajEpizodu

teorijaVelikogPraska.DodajEpizodu(E1);

teorijaVelikogPraska.DodajEpizodu(E2);

teorijaVelikogPraska.DodajEpizodu(E3);

teorijaVelikogPraska.DodajEpizodu(E4);

Serija bigbangTheory1(teorijaVelikogPraska);

bigbangTheory1.Ispis();

Epizoda\* ep = bigbangTheory1.GetNajboljeOcijenjenaEpizoda();

cout << "Najbolje ocijenjena epizoda: " << ep->GetNaziv() << endl;

cout << "Ocjena: " << ep->GetProsjecnaOcjena() << endl;

cout << "Dealokacija ..." << endl;

}

void main() {

int nastaviDalje = 1;

while (nastaviDalje == 1) {

int izbor = 0;

do {

system("cls");

cout << "::Zadaci::" << endl;

cout << "(1) Zadatak 1" << endl;

cout << "(2) Zadatak 2" << endl;

cout << "(3) Zadatak 3" << endl;

cout << "(4) Zadatak 4" << endl;

cout << "(5) Zadatak 5" << endl;

cout << "Unesite odgovarajuci broj zadatka za testiranje: -->: ";

cin >> izbor;

cout << endl;

} while (izbor < 1 || izbor > 5);

switch (izbor) {

case 1: Zadatak1(); cout << "Zadatak 1. Done." << endl; break;

case 2: Zadatak2(); cout << "Zadatak 2. Done." << endl; break;

case 3: Zadatak3(); cout << "Zadatak 3. Done." << endl; break;

case 4: Zadatak4(); cout << "Zadatak 4. Done." << endl; break;

case 5: Zadatak5(); cout << "Zadatak 5. Done." << endl; break;

default:break;

}

do {

cout << "DA LI ZELITE NASTAVITI DALJE? (1/0): ";

cin >> nastaviDalje;

} while (nastaviDalje != 0 && nastaviDalje != 1);

}

}