Ishod 4

41.

#include <iostream>

#include <string>

using namespace std;

struct Biljka {

string naziv, latinskiNaziv;

bool jestivo;

double cijena;

void ispis() {

cout << "Naziv: " << naziv << endl;

cout << "Latinski naziv: " << latinskiNaziv << endl;

cout << "Je li jestivo?: ";

if (jestivo) {

cout << "Da." << endl;

}

else {

cout << "Ne." << endl;

}

cout << "Cijena: " << cijena << endl;

}

};

int main() {

Biljka breza, brusnica;

breza.naziv = "Breza";

breza.latinskiNaziv = "Betula pendula";

breza.jestivo = false;

breza.cijena = 99.99;

brusnica.naziv = "Brusnica";

brusnica.latinskiNaziv = "Vaccinium macrocarpa";

brusnica.jestivo = true;

brusnica.cijena = 19.99;

breza.ispis();

cout << endl;

brusnica.ispis();

return 0;

}

42.

#include <iostream>

#include <string>

using namespace std;

struct Drzava {

string naziv;

int brojStanovnika;

void ispis() {

cout << naziv << " ima " << brojStanovnika << " stanovnika." << endl;

}

};

Drzava kreirajDrzavu() {

Drzava d;

cout << "Upisite naziv drzave: ";

getline(cin, d.naziv);

cout << "Upisite koliko stanovnika ima drzava: ";

cin >> d.brojStanovnika;

cin.ignore();

return d;

}

int main() {

Drzava drzave[3];

for (int i = 0; i < 3; i++) {

Drzava d = kreirajDrzavu();

drzave[i] = d;

}

cout << endl;

for (int i = 0; i < 3; i++) {

Drzava drz = drzave[i];

drz.ispis();

}

return 0;

}

.

51.

#include <iostream>

using namespace std;

int main() {

int len;

cout << "Koliko brojeva zelite ucitati?: ";

cin >> len;

int\* brojevi = new int[len];

for (int i = 0; i < len; i++) {

cout << "Unesite broj: ";

cin >> brojevi[i];

}

int min = brojevi[0];

for (int i = 0; i < len; i++) {

int n = brojevi[i];

if (n < min) min = n;

}

cout << "Najmanji uneseni broj je " << min << endl;

delete[] brojevi;

return 0;

}

43.

#include <iostream>

#include <vector>

#include <string>

using namespace std;

bool provjeraMjeseca(string nazivPlanete, vector<string> mjeseci, string& naziv) {

for (int i = 0; i < mjeseci.size(); i++) {

string m = mjeseci[i];

if (m == naziv) {

cout << "Mjesec " << naziv << " priprada planeti " << nazivPlanete << endl;

return true;

}

}

return false;

}

int main() {

vector<string> zemljiniMjeseci;

zemljiniMjeseci.push\_back("Mjesec");

vector<string> marsoviMjeseci;

marsoviMjeseci.push\_back("Phobos");

marsoviMjeseci.push\_back("Deimos");

vector<string> jupiteroviMjeseci;

jupiteroviMjeseci.push\_back("Io");

jupiteroviMjeseci.push\_back("Europa");

jupiteroviMjeseci.push\_back("Callisto");

jupiteroviMjeseci.push\_back("Elara");

jupiteroviMjeseci.push\_back("Himalia");

string unosMjeseca;

cout << "Unesite naziv nekog mjeseca: ";

getline(cin, unosMjeseca);

bool postoji = false;

if (provjeraMjeseca("Zemlja", zemljiniMjeseci, unosMjeseca)) return 0;

else if (provjeraMjeseca("Mars", marsoviMjeseci, unosMjeseca)) return 0;

else if (provjeraMjeseca("Jupiter", jupiteroviMjeseci, unosMjeseca)) return 0;

else cout << "Takav mjesec ne postoji!" << endl;

return 0;

}

52.

#include <iostream>

#include <vector>

using namespace std;

const int VELICINA = 5;

void kopirajBrojeve(int\* prvo, int\* drugo, vector<int>& brojevi) {

for (int i = 0; i < VELICINA; i++) {

int b = prvo[i];

bool postoji = false;

for (int j = 0; j < VELICINA; j++) {

if (b == drugo[j]) {

postoji = true;

break;

}

}

if (postoji) {

// Zatim provjeravamo postoji li taj broj vec u vektoru

bool vecUneseno = false;

for (int k = 0; k < brojevi.size(); k++) {

if (b == brojevi[k]) {

vecUneseno = true;

break;

}

}

if (!vecUneseno) {

brojevi.push\_back(b);

}

}

}

}

int main() {

int prvoPolje[VELICINA] = { 1, 2, 3, 4, 5 };

int drugoPolje[VELICINA] = { 4, 4, 5, 6, 7 };

vector<int> brojevi;

kopirajBrojeve(prvoPolje, drugoPolje, brojevi);

cout << "Brojevi u novom vektoru: " << endl;

for (int i = 0; i < brojevi.size(); i++) {

cout << brojevi[i] << " ";

}

return 0;

}

53.

#include <iostream>

#include <string>

using namespace std;

int main() {

const int len = 5;

string imena[len];

int dynamicLen = 0;

for (int i = 0; i < len; i++) {

string ime;

cout << "Unesite neko ime: ";

getline(cin, ime);

imena[i] = ime;

if (ime.size() <= 4) {

dynamicLen++;

}

}

string\* malaImena = new string[dynamicLen];

for (int i = 0, j = 0; i < len; i++) {

string ime = imena[i];

if (ime.size() <= 4) {

malaImena[j++] = ime;

}

}

cout << endl << "Sva imena koja imaju manje od 4 slova u sebi: " << endl;

for (int i = 0; i < dynamicLen; i++) {

cout << malaImena[i] << endl;

}

delete[] malaImena;

return 0;

}

Ishod 6

61.

#include <iostream>

#include <fstream>

using namespace std;

int main() {

ifstream brojevi("brojevi.txt");

if (!brojevi) {

cout << "Greska pri otvaranju datoteke!" << endl;

return 1;

}

ofstream neparni("neparni.txt");

int n;

while (brojevi >> n) {

if (n % 2 == 1) {

neparni << n << endl;

}

}

brojevi.close();

neparni.close();

return 0;

}

62.

#include <iostream>

#include <fstream>

#include <vector>

using namespace std;

struct Pravokutnik {

int a, b;

};

int main() {

vector<Pravokutnik> pravokutnici;

ifstream dat("pravokutnici.txt");

if (!dat) {

cout << "Greska pri otvaranju datoteke!" << endl;

return 1;

}

int a;

while (dat>> a) {

int b;

dat >> b;

Pravokutnik p;

p.a = a;

p.b = b;

pravokutnici.push\_back(p);

}

for (int i = 0; i < pravokutnici.size(); i++) {

Pravokutnik p = pravokutnici[i];

if (p.a == p.b) {

cout << "Kvadrat sa stranicama " << p.a << " i " << p.b << endl;

}

}

return 0;