**1 UŽDUOTIS**

| #include <iostream> // biblioteka atsakinga už output/input  #include <string> // biblioteka atsakinga už string  using namespace std; // trumpina std  void naujasZodis(const string& zodis); // kintamasis  int main() { // Pagrindinė programos funkcija  setlocale(LC\_ALL, "Lithuanian"); // Nustatome lietuvių kalbą  apsuptasZodis("Kefyras");  apsuptasZodis("Lapas");  apsuptasZodis("Ugnikalnis");  apsuptasZodis("Tualetas");  apsuptasZodis("Korta"); // generuojam naujus žodžius  return 0;  }  void naujasZodis(const string& zodis) // Sudės brūkšnelius iš šonų  {  cout << "-- " << zodis << " --" << endl; // // Sudės kablelius iš šonų  } |
| --- |

**3 UŽDUOTIS**

| #include <iostream>  using namespace std;  double skaiciuoti(double pirmas, double antras, char veiksmas); // KINTAMIEJI  int main() { // Pagrindinė programos funkcija  setlocale(LC\_ALL, "Lithuanian"); // Nustatome lietuvių kalbą  cout << skaiciuoti(5, 7, '+') << endl;  cout << skaiciuoti(10, 3, '-') << endl;  cout << skaiciuoti(4, 6, '\*') << endl;  cout << skaiciuoti(8, 2, '/') << endl; //išveda rezultatus su cout  return 0;  }  double skaiciuoti(double pirmas, double antras, char veiksmas) { Ši eilutė pradeda skaiciuoti funkcijos apibrėžimą.  double atsakymas; //kintamasis  switch (veiksmas) {  case '+':  atsakymas = pirmas + antras;  break;  case '-':  atsakymas = pirmas - antras;  break;  case '\*':  atsakymas = pirmas \* antras;  break;  case '/':  if (antras != 0) {  atsakymas = pirmas / antras;  } else {  cout << "Dalyba iš nulio negalima." << endl;  return 0;  }  break;  default:  cout << "Neatpažintas veiksmas." << endl;  return 0; //atrenkami tinkami veiksmai su case pagalba  }  return atsakymas; // grąžina atsakymą  } |
| --- |

**5 UŽDUOTIS**

| #include <iostream> // biblioteka atsakinga už output/input  #include <string> // biblioteka atsakinga už string  using namespace std;  void informacija(string vardas, string pavarde, string grupesKodas);  double vidurkis(int pazymys1, int pazymys2, int pazymys3);  void apjungtiInfoIrVidurki(string vardas, string pavarde, string grupesKodas, int pazymys1, int pazymys2, int pazymys3); // parodomos funkcijos  int main() // Pagrindinė programos funkcija  {  setlocale(LC\_ALL, "Lithuanian"); // Nustatome lietuvių kalbą  sujungtiinfo("Jonas", "Jonaitis", "AF3", 8, 9, 7);  sujungtiinfo("Laura", "Laurutė", "BB2", 10, 9, 10); // apjungia visas funkcijas  return 0;  }  void informacija(string vardas, string pavarde, string grupesKodas) // išveda gautą info  {  cout << "Vardas: " << vardas << endl;  cout << "Pavardė: " << pavarde << endl;  cout << "Grupės kodas: " << grupesKodas << endl;  }  double vidurkis(int pazymys1, int pazymys2, int pazymys3) // išveda vidurkį  {  double vidurkis = (pazymys1 + pazymys2 + pazymys3) / 3.0; // Naudojame 3.0, kad gauti tikslų dalinamąjį  cout << "Pažymių vidurkis: " << vidurkis << endl;  return vidurkis; // vidurkis grąžinamas  }  void sujungtiinfo(string vardas, string pavarde, string grupesKodas, int pazymys1, int pazymys2, int pazymys3) // apjungia visą info + vidurkį  {  informacija(vardas, pavarde, grupesKodas);  vidurkis(pazymys1, pazymys2, pazymys3);  } |
| --- |

**7 UŽDUOTIS**

| #include <iostream> // biblioteka atsakinga už output/input  #include <cstdlib> // biblioteka atsakinga už funkcijų kolekciją  #include <ctime> // biblioteka, atsakinga už laiko ir datos kontroliavimą  using namespace std;  void GeneruotiSkaiciu(int kiekis, int minimumas, int maximumas); // sudaroma funkciją  int main() {  setlocale(LC\_ALL, "Lithuanian"); // nustatome lietuvių kalbą  int kiekis, minimumas, maximumas; // kintamieji  cout << "Įveskite kiekį, kiek sugeneruoti skaičių: ";  cin >> kiekis; // Surenkam atsakymus iš vartotojo  cout << "Įveskite mažiausią generuojamų skaičių reikšmę: ";  cin >> minimumas; // Surenkam atsakymus iš vartotojo  cout << "Įveskite didžiausią generuojamų skaičių reikšmę: ";  cin >> maximumas; // Surenkam atsakymus iš vartotojo  GeneruotiSkaiciu(kiekis, minimumas, maximumas);  cout << endl;  return 0;  }  void GeneruotiSkaiciu(int kiekis, minimumas, maximumas) { // Funkcija  srand(time(NULL));  cout << "Sugeneruoti skaičiai: "; // Išvedame atsakymą  for (int i = 0; i < count; i++) {  int generuotassk = rand() % (maximumas - minimumas + 1) + minimumas;  cout << generuotassk; // Išvedame sugeneruotus random skaičius  if (i != kiekis - 1) {  cout << ", ";  } else {  cout << ". "; // Dedame kablelius  }  }  } |
| --- |