



Algoritmų parinkimas.

Sveikieji ir realieji skaičiai

Praėjusios pamokos santrauka

- Supažinome su paprogramių (angl. functions) sąvoka.
- Išmokome kurti ir naudoti paprogrames su parametrais.
- Supratome kaip veikia parametrai, perduodami į paprogrames, bei skirtumus tarp vertės ir nuorodos perdavimo.
- Atlikome praktinius užsiėmimus su paprogramėmis Python ir C++ kalbose.

Ką išmoksime šiandien

- Išmoksime naudoti sveikuosius ir realiuosius skaičius Python ir C++ programose
- Suprasime skaičių tipų skirtumus ir su jais atliekamas operacijas skirtingose programavimo kalbose.

Python: Skaičių tipai

- Sveikieji skaičiai (**int**):
 - Naudojami teigiamiems, neigiamiems skaičiams ir nuliui be dešimtainės dalies.
 - Pavyzdys: **5**, **-10**, **0**.
- Realieji skaičiai (**float**):
 - Naudojami skaičiams su dešimtainėmis dalimis.
 - Pavyzdys: **3.14**, **-2.5**, **0.001**, **2.0**, **0.0**.

C++: Skaičių tipai

- Sveikieji skaičiai (**int**, **long**):
 - **int**: Naudojami mažesniems sveikiesiems skaičiams (paprastai iki ~2 milijardų).
 - **long**: Naudojami didesniems sveikiesiems skaičiams.
 - Pavyzdys: **int x = 10; long y = 1000000000;**
- Realieji skaičiai (**float**, **double**):
 - **float**: Naudojamas realiesiems skaičiams su mažesniu tikslumu (apie 7 skaitmenys po kablelio).
 - **double**: Naudojamas didesniajam tikslumui (apie 15 skaitmenų po kablelio).
 - Pavyzdys: **float a = 3.14f; double b = 3.141592653589793;**

Skaičių tipų skirtumai tarp Python ir C++

- Python naudoja **dinaminę tipų sistemą**, todėl jums nereikia aiškiai nurodyti tipo, Python automatiškai priskiria tipą (pvz., `int` arba `float`) pagal reikšmę.
- C++ reikalauja **statinio tipo priskyrimo**, todėl reikia iš anksto nurodyti, kokio tipo duomenis naudosite (`int`, `float`, `double` ir kt.), kas užtikrina didesnę kontrolės lygį ir efektyvumą.

Maksimalios sveikųjų ir realiųjų skaičių reikšmės (Python)

int (Sveikasis skaičius):

- Maksimalios reikšmės:

- Python **neturi fiksuoto int dydžio limitu** – skaičiai gali būti itin dideli (tik ribojami atminties dydžio).
- Pavyzdys: **123456789012345678901234567890**

float (Realiai skaičius):

- Maksimali reikšmė:

- Apytiksliai **1.8×10^{308}**
- Tikslumas: 15-17 skaitmenų po kablelio.
- Pavyzdys: **1.7976931348623157e+308**

Maksimalios sveikųjų ir realiųjų skaičių reikšmės (C++)

int:

- Maksimali reikšmė:
 - ❑ **2,147,483,647** (32 bitų sistemoje)
 - ❑ Pavyzdys: **2147483647**

long:

- ❑ Maksimali reikšmė:
- ❑ **2³¹-1** (32 bitų – 2,147,483,647)
- ❑ **2⁶³-1** (64 bitų – 9,223,372,036,854,775,807)
- ❑ Pavyzdys: **9223372036854775807L**

float:

- Maksimali reikšmė:
 - ❑ Apytiksliai 3.4×10^{38}
 - ❑ Tikslumas: Apie 7-8 skaitmenys po kablelio.
 - ❑ Pavyzdys: **3.4028235e+38**

double:

- Maksimali reikšmė:
 - ❑ Apytiksliai 1.8×10^{308}
 - ❑ Tikslumas: Apie 15-16 skaitmenų po kablelio.
 - ❑ Pavyzdys: **1.7976931348623157e+308**

Maksimalių reikšmių skirtumai tarp Python ir C++

- Python `int` nėra ribojamas, tuo tarpu C++ `int` ir `long` turi fiksuotas maksimalias reikšmes.
- Abiejose kalbose `float` ir `double` turi panašias maksimalias reikšmes, tačiau `double` suteikia didesnį tikslumą.

Konvertavimas tarp tipų (Python)

- Automatinis konvertavimas:
 - Python dažnai automatiškai konvertuoja reikšmes tarp skirtingų tipų (pvz., iš `int` į `float` dalybos metu).
- Aiškus konvertavimas:
 - Naudokite `int()`, `float()`, kad aiškiai nurodytumėte tipų konvertavimą.

```
# Konvertavimas iš  
# int į float  
a = int(5)  
b = float(a)
```

Konvertavimas tarp tipų (C++)

- Rankinis konvertavimas:
 - ▣ C++ reikalauja aiškos konvertavimo komandos (pvz., naudojant `static_cast`)

```
// Konvertavimas iš int į float  
int a = 5;  
float b = static_cast<float>(a);
```

Kodo pavyzdžiai su sveikaisiais ir realiaisiais skaičiais (Python)

```
1  # Sveikųjų skaičių pavyzdys
2  a = int(input("Įveskite pirmą sveikąjį skaičių: "))
3  b = int(input("Įveskite antrą sveikąjį skaičių: "))
4
5  suma = a + b
6  print("Suma:", suma)
7
8  # Realieji skaičiai (float)
9  x = float(input("Įveskite pirmą realųjį skaičių: "))
10 y = float(input("Įveskite antrą realųjį skaičių: "))
11
12 sandauga = x * y
13 print("Sandauga:", sandauga)
```

Kodo pavyzdžiai su sveikaisiais ir realiaisiais skaičiais (C++)

```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      // Sveikieji skaičiai
6      int a, b;
7      cout << "Įveskite pirmą sveikąjį skaičių: ";
8      cin >> a;
9      cout << "Įveskite antrą sveikąjį skaičių: ";
10     cin >> b;
11
12     int suma = a + b;
13     cout << "Suma: " << suma << endl;
14
15     // Realieji skaičiai (float)
16     float x, y;
17     cout << "Įveskite pirmą realųjį skaičių: ";
18     cin >> x;
19     cout << "Įveskite antrą realųjį skaičių: ";
20     cin >> y;
21
22     float sandauga = x * y;
23     cout << "Sandauga: " << sandauga << endl;
24
25     return 0;
26 }
```

Praktika

■ Sveikieji skaičiai:

- **Užduotis1:** Parašykite programą, kuri suskaičiuoja dviejų sveikųjų skaičių sumą, skirtumą ir sandaugą.

■ Realieji skaičiai:

- **Užduotis2:** Parašykite programą, kuri suskaičiuoja dviejų realiųjų skaičių sumą, skirtumą ir sandaugą, bei padalina pirmą skaičių iš antro.

Apibendrinimas (Skaičių tipų naudojimo skirtumai)

Python:

- `int` – sveikieji skaičiai.
- `float` – realieji skaičiai.
- Python automatiškai pritaiko atminties dydį pagal reikšmę.

C++:

- `int` – sveikieji skaičiai.
- `float` ir `double` – realieji skaičiai, skiriasi tikslumu:
 - `float` – mažesnis tikslumas (4 baitai).
 - `double` – didesnis tikslumas (8 baitai).
- C++ reikalauja konkretesnio skaičių tipo pasirinkimo.

Apibendrinimas (Konvertavimas tarp tipų)

- **Python:** automatinis konvertavimas (pvz., iš `int` į `float`), bet galima naudoti `int()` ir `float()`.
- **C++:** reikalauja aiškaus konvertavimo naudojant `static_cast<>` arba panašius būdus.

Užduotis namuose

Sukurti programą, kuri atlieka skaičių konvertavimą tarp sveikųjų ir realiųjų skaičių tipų abiejose kalbose (pvz., iš sveikojo skaičiaus į realųjį ir atvirkščiai)



Pabaiga