



Duomenų struktūrų
naudojimas.

Vienmatis masyvas

Duomenų struktūros

- Suprasti pagrindines duomenų struktūrų sąvokas
 - masyvai
 - susietieji sąrašai
 - stekai / deklai (dekas)
 - eilės
- Įgyvendinimas naudojant Python ir C++
- Suprasti įvairių duomenų struktūrų privalumus ir trūkumus

Ką išmoksime šiandien

- Suprasime vienmačio masyvo sąvoką
 - Kas yra masyvas ir kaip jis veikia.
- Išmoksime naudoti vienmačius masyvus Python ir C++ kalbose
 - Sukurti masyvą, pasiekti jo elementus, keisti vertes.
- Išmoksime atlikti operacijas su masyvais
 - Sumuoti elementus, rasti didžiausią arba mažiausią reikšmę.
- Suprasime masyvų taikymą realiuose pavyzdžiuose
 - Realūs pavyzdžiai, kurie padės įsisavinti masyvų naudojimą programuojant.

Kas yra vienmatis masyvas

- Tai duomenų struktūra, kuri saugo elementų seką vienoje eilėje.
- Kiekvienas elementas turi savo indeksą (vietą) masyve, pradedant nuo 0.
- Masyvą naudojame, kai reikia saugoti daug elementų to paties tipo (pvz., skaičių ar tekstų)

Vienmačio masyvo naudojimas (Python)

- Masyvo (sąrašo) sukūrimas:

- Sąrašas masyvas turi 5 elementus.
- Indeksavimas prasideda nuo 0.

```
masyvas = [1, 2, 3, 4, 5]
```

- Elementų pasiekimas:

```
print(masyvas[0])  
# Rezultatas: 1
```

- Elemento keitimas:

```
masyvas[1] = 10  
print(masyvas)  
# Rezultatas: [1, 10, 3, 4, 5]
```

Vienmačio masyvo naudojimas (C++)

- Masyvo (sąrašo) sukūrimas:

- Sąrašas masyvas turi 5 elementus.
- Indeksavimas prasideda nuo 0.

```
int masyvas[5] = {1, 2, 3, 4, 5};
```

- Elementų pasiekimas:

```
cout << masyvas[0];  
// Rezultatas: 1
```

- Elemento keitimas:

```
masyvas[1] = 10;  
cout << masyvas[1];  
// Rezultatas: 10
```

Dažniausios klaidos

■ Indeksų klaidos:

- ❑ Bandymas pasiekti neegzistuojantį indeksą:
 - Python: `IndexError`
 - C++: neapibrėžtas elgesys.

■ Netinkamas masyvo dydis:

- ❑ Python masyvo dydis gali keistis, bet C++ masyvas turi fiksuotą dydį.

Išvados

- Vienmačiai masyvai leidžia lengvai valdyti duomenis toje pačioje duomenų struktūroje.
- Python ir C++ masyvų naudojimo sintaksė skiriasi, tačiau principai panašūs: indeksavimas nuo 0, elementų pasiekimas ir keitimas pagal jų indeksą

Užduotis

1. Sukurkite vienmatį masyvą, kuris saugo 10 sveikųjų skaičių.
2. Raskite didžiausią masyvo elementą.
3. Apskaičiuokite visų masyvo elementų vidurkį.

Masyvų taikymo pavyzdžiai

Masyvas studentų pažymiams saugoti

- Užduotis: Sukurti masyvą, kuris saugo 5 studentų pažymius

```
pazymiai = [8, 9, 7, 10, 6]
vidurkis = sum(pazymiai) / len(pazymiai)
print("Vidutinis pažymys:", vidurkis) # Rezultatas: 8.0
```

```
int pazymiai[5] = {8, 9, 7, 10, 6};
int suma = 0;
for (int i = 0; i < 5; i++) {
    suma += pazymiai[i];
}
cout << "Vidutinis pažymys: " << (suma / 5) << endl;
// Rezultatas: 8
```

Masyvų taikymo pavyzdžiai (tęs)

Masyvas prekių kainoms saugoti

- Užduotis: Sukurti masyvą, kuris saugo 5 prekių kainas

```
kainos = [10.5, 20.0, 5.99, 15.0, 8.75]
brangiausia = max(kainos)
print("Brangiausia prekė kainuoja:", bragiausia)
# Rezultatas: 20.0
```

```
float kainos[5] = {10.5, 20.0, 5.99, 15.0, 8.75};
float bragiausia = kainos[0];
for (int i = 1; i < 5; i++) {
    if (kainos[i] > bragiausia) {
        bragiausia = kainos[i];
    }
}
cout << "Brangiausia prekė kainuoja: " << bragiausia << endl;
// Rezultatas: 20.0
```

Masyvų taikymo pavyzdžiai (tęs)

Masyvas temperatūrų duomenims saugoti

- Užduotis: Sukurti masyvą, kuris saugo 7 dienų temperatūrų duomenis

```
temp = [20, 22, 19, 21, 23, 20, 18]
temp_min = min(temp)
temp_max = max(temp)
print("Mažiausia temperatūra:", temp_min)
# Rezultatas: 18
print("Didžiausia temperatūra:", temp_max)
# Rezultatas: 23
```

```
int temp[7] = {20, 22, 19, 21, 23, 20, 18};
int temp_min = temp[0], temp_max = temp[0];
for (int i = 1; i < 7; i++) {
    if (temp[i] < temp_min) temp_min = temp[i];
    if (temp[i] > temp_max) temp_max = temp[i];
}
cout << "Mažiausia temperatūra: " << temp_min << endl;
// Rezultatas: 18
cout << "Didžiausia temperatūra: " << temp_max << endl;
// Rezultatas: 23
```

Apibendrinimas

- Vienmačio masyvo sąvoka
 - Supratome, kas yra vienmatis masyvas ir kokia jo struktūra.
- Masyvų naudojimas Python ir C++
 - Išmokome, kaip sukurti, pasiekti ir keisti masyvo elementus abiejose kalbose.
- Operacijos su masyvais
 - Išmokome atlikti praktines operacijas:
 - Elementų sumavimas.
 - Didžiausio ir mažiausio elemento radimas.
- Praktiniai pavyzdžiai
 - Pateikėme realius pavyzdžius, kurie iliustravo, kaip naudoti vienmačius masyvus programavimo užduotims spręsti.



Pabaiga