

---

# Algoritmų parinkimas.

## Rekursija

# Tikslai

- Suprasti rekursijos sąvoką ir principus
- Išmokti rašyti rekursines funkcijas programavimo kalboje
- Suvokti rekursijos taikymo sritis ir privalumus/trūkumus

# Kas yra rekursija

*Rekursija – tai procesas, kai funkcija kviečia pati save spręsti mažesnę problemos versiją*

# Pavyzdys iš realaus gyvenimo

„Veidrodžių kambarys“  
– stovint tarp dviejų  
veidrodžių matome  
begalinį vaizdo  
atspindį, panašiai  
veikia ir rekursija



# Rekursijos pavyzdys (Faktorialo skaičiavimas)

```
def faktorialas(n):  
    if n == 0:  
        return 1  
    else:  
        return n * faktorialas(n-1)
```

Python

```
long long fakt(int n) {  
    if(n == 0) return 1;  
    return n*fakt(n-1);  
}
```

C++

# Praktika

## ■ Apskaičiuoti Fibonačio seką

***Fibonacci seka (Fibonāčio sekà), skaičių seka, kurios kiekvienas paskesnis narys lygus dviejų prieš jį esančių narių sumai, t. y.  $F_{i+2} = F_{i+1} + F_i$ ,  $i = 1, 2, \dots$ ;  $F_1 = F_2 = 1$ . Pirmieji sekos nariai yra 1, 1, 2, 3, 5, 8, 13, 21, ....***

1, 1, 2, 3, 5, 8, 13, 21, 34...

# Kur naudojama rekursija?

- Algoritmai: binarės paieškos medis, rūšiavimas, tokie kaip greitojo rūšiavimo algoritmas.
- Matematinės problemos: skaičių sekos, kombinatorika.
- Žaidimų kūrimas: algoritmai, sprendžiantys galvosūkius, maršrutų paieška

# Rekursijos privalumai / trūkumai

## Privalumai

- Lengvesnis problemos supratimas ir sprendimas / švaresnis ir elegantiškesnis kodas
- Tinka tam tikrų algoritmų įgyvendinimui
- Naudojimas, kai nežinomas tikslus iteracijų kiekis

## Trūkumai

- Didelis atminties naudojimas / lėtesnis veikimas
- Rizika įstrigti begalinėje rekursijoje
- Platformos apribojimai



# Užduotis namuose

- Perrašyk savo programą, kuri naudoja ciklą, taip, kad ji naudotų rekursiją.

# Daugiau

- [https://inf-knyga.nmakademija.lt/lt/latest/04\\_rekursija.html#f12](https://inf-knyga.nmakademija.lt/lt/latest/04_rekursija.html#f12)
- <https://blog.skillfactory.ru/glossary/rekursiya/>
- <https://practicum.yandex.ru/blog/rekursiya-v-programmirovanii/>



Pabaiga