

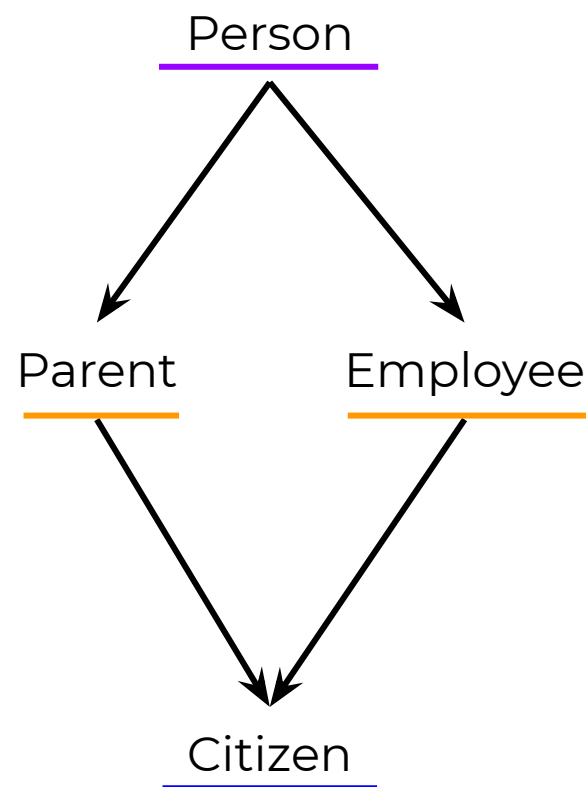
Skillbox

# **Множественное наследование: MRO**

**Роман Булгаков**

Спикер

# Diamond problem

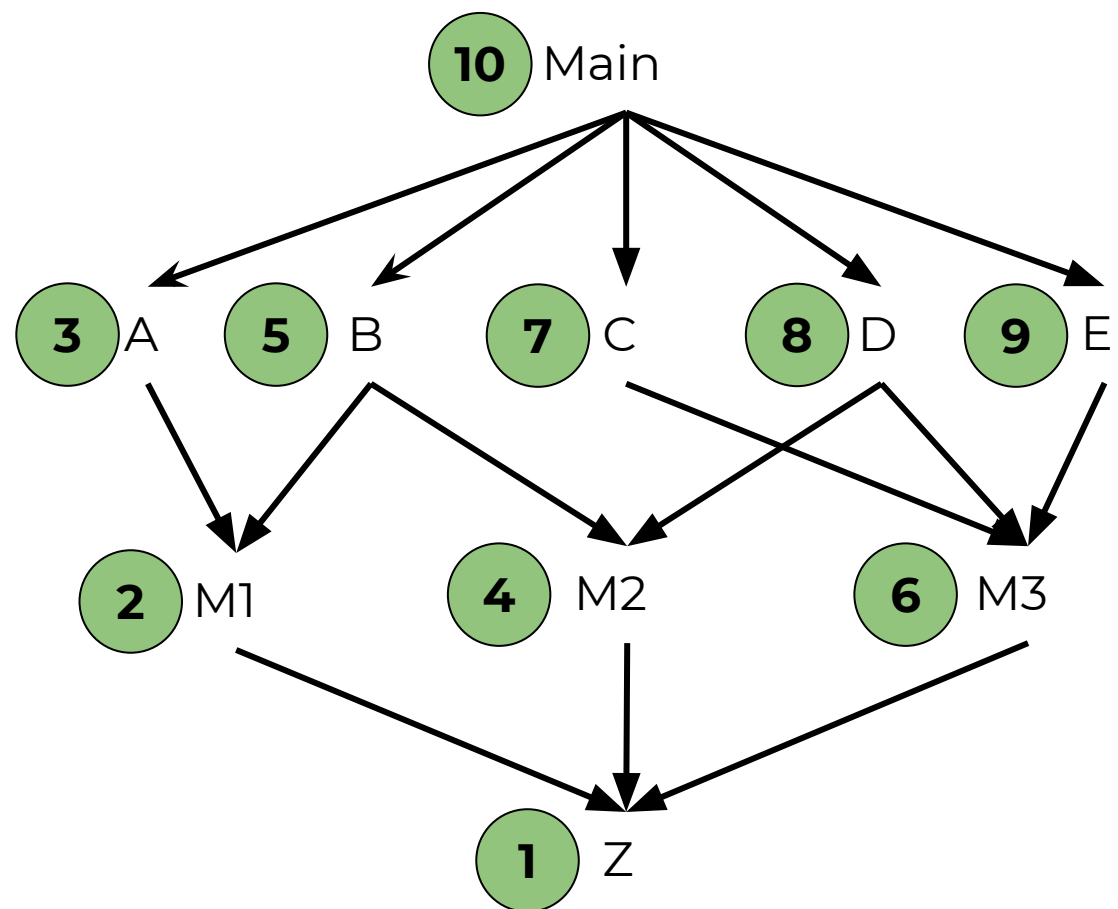


**Method Resolution Order (MRO)** —  
порядок разрешения методов.

1. Citizen
2. Parent
3. Employee
4. Person

# Diamond problem

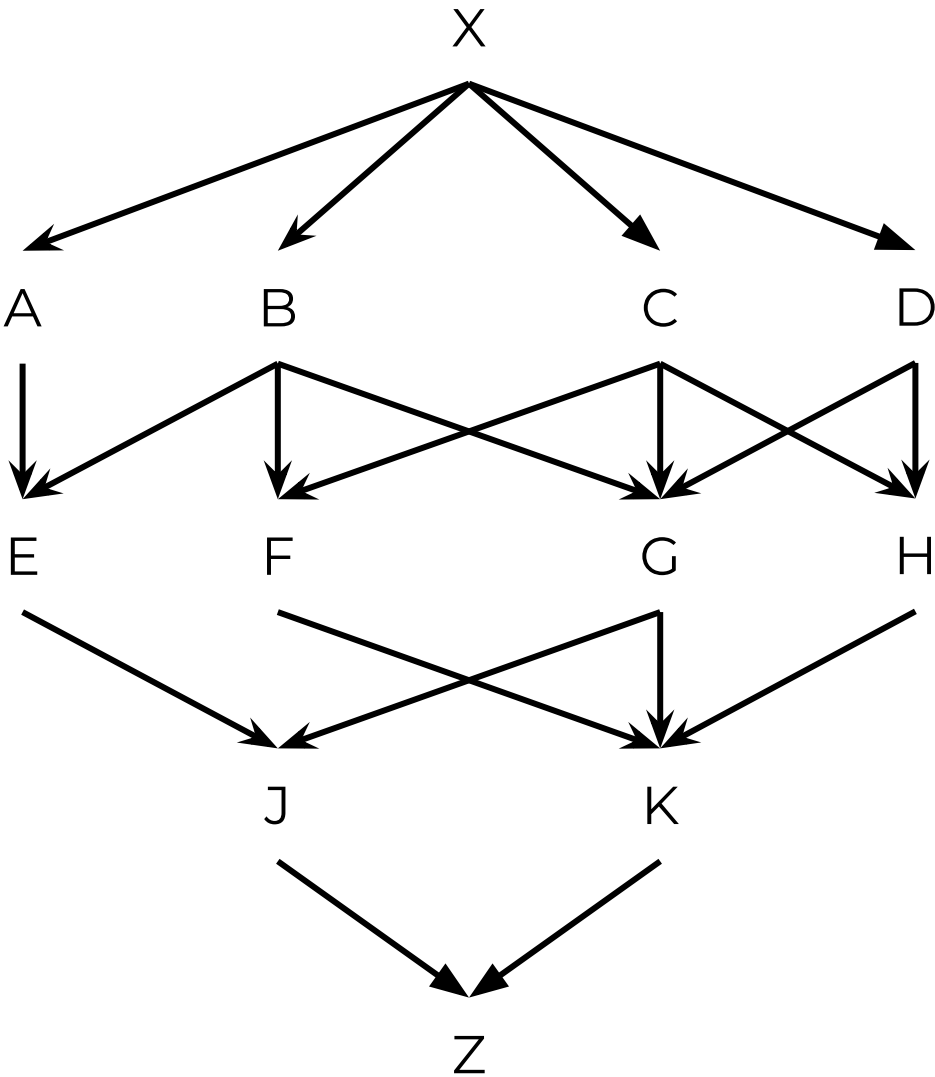
```
+class Main:...\n+class A(Main):...\n+class B(Main):...\n+class C(Main):...\n+class D(Main):...\n+class E(Main):...\n+class M1(A, B):...\n+class M2(B, D):...\n+class M3(C, D, E):...\n+class Z(M1, M2, M3):...
```



~~MRO: Z -> M1 -> M2 -> M3 -> A -> B -> C -> ...~~

MRO: Z -> M1 -> A -> M2 -> B -> M3 -> C -> D -> E -> Main

# Для задачи



# Множественное наследование

Одна из самых спорных вещей в ООП: стоит ли это использовать?

**Плюс:** позволяет сокращать затраты на разработку класса и избегать повторного использования кода.

**Минус:** повышает сложность создания и модификации системы классов. Увеличивает связь между классами, а значит, изменения в базовом классе могут повлечь серьёзные проблемы в дочерних.

**Итого:** обычно не используют, но есть ситуации, когда оно необходимо.

Skillbox

**Спасибо  
за внимание!**