

### 1. Definicja i podstawowe pojęcia:

- Co to jest metoda wirtualna w C++?
- Co to jest klasa abstrakcyjna w C++?
- Jakie są różnice między zwykłą metodą a metodą wirtualną?

### 2. Implementacja i użycie:

- Jak zadeklarować metodę wirtualną w C++? Podaj przykład.
- Jak zadeklarować klasę abstrakcyjną w C++? Podaj przykład.
- Jak jest znaczenie słowa kluczowego **virtual** w deklaracji metody?

### 3. Polimorfizm:

- Co to jest polimorfizm i jak metody wirtualne go wspierają?
- Jakie korzyści przynosi używanie metod wirtualnych w programowaniu obiektowym?
- Jak działa mechanizm wirtualny w czasie wykonywania programu?

### 4. Mechanizmy i szczegóły implementacyjne:

- Co to jest wirtualna tabela (vtable) i jak działa w C++?
- Jakie są konsekwencje używania metod wirtualnych w kontekście wydajności?

### 5. Przykłady i scenariusze:

- Podaj przykład klasy bazowej z metodą wirtualną i klasy pochodnej, która ją nadpisuje.
- Jakie błędy można popełnić przy używaniu metod wirtualnych i jak ich unikać?
- Jakie są przypadki, w których użycie klas abstrakcyjnych jest zalecane?

### 6. Słowo kluczowe **override** i **final**:

- Jak jest znaczenie słowa kluczowego **override** w C++?
- Jak jest znaczenie słowa kluczowego **final** w kontekście metod wirtualnych?

### 7. Destruktory wirtualne:

- Dlaczego ważne jest, aby destruktory były wirtualne w klasach bazowych?
- Co się stanie, jeśli destruktory klasy bazowej nie jest wirtualny?

## 8. Porównanie z innymi językami:

- Jak metody wirtualne i klasy abstrakcyjne w C++ różnią się od ich odpowiedników w innych językach programowania, takich jak Java lub C#?

## 9. Przykładowe problemy:

- Stwórz program, który demonstruje działanie metod wirtualnych w hierarchii klas.
- Napisz program, który wykorzystuje klasy abstrakcyjne do zdefiniowania interfejsu i implementuje go w kilku klasach pochodnych.