

Programowanie obiektowe 1

Gabriel Malanowski

3 kwietnia 2025

Ten dokument zawiera zestaw zadań do nauki programowania obiektowego. Każdy projekt został zaprojektowany tak, aby rozwijać kluczowe umiejętności OOP, takie jak dziedziczenie, polimorfizm oraz zarządzanie danymi. Projekty symulują rzeczywiste systemy, co pozwala na naukę poprzez praktykę.

1 Biblioteka

1.1 Cel

Stworzenie prostego systemu wypożyczania i zwracania książek w bibliotece.

1.2 Wymagania

- Stwórz klasy `Book`, `Member`, `Librarian` oraz `Library`.
- Każda książka ma tytuł, autora i status (dostępna/wypożyczona).
- Członkowie mogą wypożyczać i zwracać książki.
- Bibliotekarz zarządza biblioteką, może dodawać nowe książki.
- Parametry (np. lista książek i członków) wczytywane z pliku lub podawane przez użytkownika (opcjonalnie w pliku bazodanowym).
- Zapisywanie historii wypożyczeń do pliku.

2 Sieć komputerowa w firmie

2.1 Cel

Zasymulowanie przesyłu danych w sieci firmowej.

2.2 Wymagania

- Stwórz klasy `Computer`, `Server`, `Router`, `Network`.
- Każdy komputer może wysyłać i odbierać dane.
- Serwer przechowuje pliki i odpowiada na żądania.
- Router zarządza ruchem w sieci.
- Możliwość symulacji awarii routera lub przeciążenia sieci.
- Parametry (np. liczba komputerów, pakiety danych) wczytywane z pliku lub podawane przez użytkownika.
- Zapisywanie logów aktywności do pliku.

3 Magazyn

3.1 Cel

Stworzenie symulacji zarządzania zapasami w magazynie.

3.2 Wymagania

- Stwórz klasy `Product`, `Warehouse`, `Order`
- Produkty mają nazwę, ilość i cenę.
- Magazyn przechowuje produkty i aktualizuje ich stan.
- Zamówienia zmniejszają ilość produktów w magazynie.
- Symulacja dostaw i odbiorów towaru.
- Parametry (np. lista produktów, zamówienia) wczytywane z pliku lub podawane przez użytkownika.
- Zapisywanie stanu magazynu do pliku.

4 Komunikacja miejska

4.1 Cel

Stworzenie symulacji działania miejskiej komunikacji.

4.2 Wymagania

- Stwórz klasy `Bus`, `Tram`, `Route`, `Passenger`.
- Pojazdy poruszają się po trasach i zabierają pasażerów.
- Pasażerowie mają miejsce początkowe i docelowe.
- Możliwość dodawania nowych tras i pojazdów.
- Parametry (np. liczba pojazdów, przystanki) wczytywane z pliku lub podawane przez użytkownika.
- Zapisywanie statystyk podróży do pliku.

5 Pomoce

5.1 Artykuły w Internecie

- <https://technikprogramista.pl/kurs/cpp/lekcja/cpp-klasy-i-objekty/>
- <https://technikprogramista.pl/kurs/cpp/lekcja/cpp-konstruktor-i-destruktor/>
- <https://miroslawzelent.pl/kurs-objektowy-c++/podejscie-objektowe-objekt-klasa/>
- <https://sii.pl/blog/programowanie-objektowe-na-przykladzie-python/>

5.2 Literatura

- <https://helion.pl/ksiazki/python-zaczynaj-programowac-rob-miles,pytzap.htm>
- <https://helion.pl/ksiazki/jezyk-c-szkola-programowania-wydanie-vi-stephen-prata,cppr6v.htm>

5.3 Przykładowe repozytoria

- <https://github.com/gmalanowski/Warehouse-simulator>
- <https://github.com/mnurzyns/BakcylProgramowania2021>
- <https://github.com/gmalanowski/PWR-ITE>