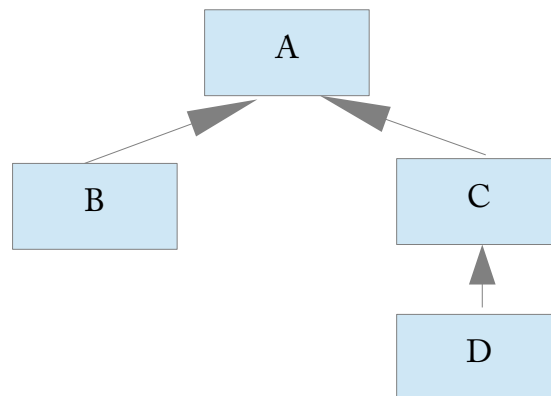


Celem projektu jest rozszerzenie możliwości obiektu stworzonego w pierwszym projekcie z wykorzystaniem dziedziczenia.

1. Zaproponować, zaprojektować i zrealizować dla klasy głównej z projektu 1 następujący schemat dziedziczenia:



Obiekt z projektu 1 może być dowolnym obiektem z powyższych (B, C lub D). Klasa A powinna być abstrakcyjna. Proszę zaprojektować i zdefiniować wybrane, sensowne metody wirtualne dla klasy bazowej i pochodnych.

2. Zaimplementować operatory strumieniowe dla wszystkich utworzonych klas

3. Rygorystycznie potraktować deklaratory zakresu - private, protected i public, stosując zasadę najmniejszych możliwych przywilejów.

4. Wszystkie klasy powinny zawierać metody zapamiętywania stanu **całego** obiektu w pliku i odtwarzania jego stanu z dysku. Zapis i odczyt powinien być wykonany kaskadowo/hierarchicznie tzn. tak, aby dana metoda danej klasy zajmowała się odczytem i zapisem jedynie "swoich" pól.

5. Wykorzystać właściwości polimorfizmu do zademonstrowania działania metod wirtualnych mając do dyspozycji tablicę wskaźników na obiekty klasy podstawowej.

6. Napisać aplikację z konsolowym interfejsem. Program główny powinien testować wybrane działania obiektu/obiektów głównych. Dla klas pochodnych powinny działać metody rzutu i odczytu stanu obiektu do/z pliku dyskowego z wykorzystaniem przeciążonych operatorów strumieniowych. Program główny powinien demonstrować zastosowanie wcześniej zaprojektowanych metod wirtualnych. Interfejs powinien być niewrażliwy na błędy użytkownika.

7. Utworzyć dokumentację projektu (PDF).

Jeżeli jest wybór pomiędzy stosowaniem mechanizmów, funkcji, instrukcji typowych dla języka C i C++ proszę stosować odpowiednie konstrukcje właściwe dla C++ np. char* - string, FILE* - iostream, itp. Jeden obiekt – 2 pliki: obiekt.h, obiekt.cpp.