

## ZADANIE 1. ADAPTER, FASADA

TERMIN WYKONANIA: 29 PAŹDZIERNIKA

1a. Wyobraźmy sobie, że mamy bardzo prosty system do obsługi zakupu instrumentów bankowych. Zakładamy, że klient może kupić/sprzedać dowolną ilość danego instrumentu finansowego (pod warunkiem, że instrument jest zarejestrowany w systemie). W systemie istnieją następujące klasy (z metodami). `ManagementSystem`(`addClient(...)`, `removeClient(...)`, `printClientsInstruments(...)`), `Client`(`buyInstrument(...)`, `sellInstrument(...)`), `Instrument`(`register(...)`, `unregister(...)`).

Proszę zaprojektować wewnętrzną strukturę tych klas oraz dodać odpowiednie parametry do metod. Następnie proszę stworzyć klasę API, która będzie udostępniała wszystkie powyższe metody "na zewnątrz".

1b. Proszę zaimplementować klasę `LegacyCar`, która zawiera informacje o swoim położeniu (X, Y) i posiada metodę `drive(X1, Y1)` – zmieniająca położenie samochodu na mapie o wektor (X1, Y1). Następnie stworzyć klasę `Racetrack`, która zawierać będzie listę obiektów `Vehicle` (klasa `Vehicle` posiada metodę `moveTo(X2, Y2)`, która przesuwa pojazd do podanego położenia). Proszę w implementacji klasy `Vehicle` wykorzystać klasę `LegacyCar` w taki sposób, żeby uniknąć duplikacji kodu.