

## Grajek

Pewien bardzo bogaty koncern muzyczny, postanowił wynająć studenta informatyki z Wydziału MIM do zaprojektowania systemu Grajek. System ten służy do sprzedaży utworów muzycznych przez internet bezpośrednio od kompozytorów (oczywiście z odpowiednią prowizją dla koncertu).

Użytkownicy systemu dzielą się na dwie rozłączne grupy: *kompozytorów*, którzy mogą zamieszczać swoje utwory oraz klientów mogących je kupić. Utwory posiadają swój gatunek określany napisem (np., „discopolo”, „jazz”, „ludowa”). Każdy użytkownik posiada określoną ilość pieniędzy na koncie i może w je każdej chwili doładować. Klienci kupują za pieniądze, a kompozytorzy zarabiają na swoich utworach. Z każdą transakcją konto kupującego jest obciążane, a kompozytora jest zasilane gotówką pomniejszoną o prowizję koncertu. Ponadto, każdy kompozytor posiada licznik ściągnięć (*delta*), który określa ile razy ich utwory były kupowane.

Najlepsi kompozytorzy mogą otrzymać status wyłączności. Oznacza to, że kompozytor zamieszcza swoje utwory tylko w Grajku i nie może tego rozbić u konkurencji. Ten status oznacza także, że kompozytor dostaje wyższą prowizję od sprzedaży utworu (50% zamiast standardowych 30%). By uniknąć nadmiernego przeładowania systemu nie każda kompozycja może być w nim umieszczona. Najpierw musi być zaakceptowana przez specjalnego użytkownika tzw. *recenzenta*. Recenzentem może być tylko wyłączny kompozytor. Do zadań recenzentów należy również określenie ceny sprzedaży zaakceptowanego utworu. Dla każdego kompozytora określa się *współczynnik akceptacji (alfa)*. Jest to liczba z przedziału [0..1], która określa jaka część ze wszystkich utworów została zaakceptowana.

Kompozytor, którego  $\alpha > 0.5$  oraz  $\delta > 99$ , może zostać wyłącznym. Wówczas do kompozytora jest wysyłane zapytanie czy chce zostać wyłącznym i w przypadku pozytywnej odpowiedzi, system przekształca kompozytora w kompozytora wyłącznego. Na podobnej zasadzie działa przekształcanie w recenzenta, ale warunkami są  $\delta > 999$  oraz  $\alpha > 0.9$ .

Klient może kupić utwory w Grajku, wtedy gdy system wyśle do niego ofertę. Wówczas wysyła do systemu propozycję kupna z ceną, którą uzna za stosowną (uwaga klient nie wie jaka jest cena utworu z oferty). System tylko sprawdza, czy proponowana cena jest wyższa lub równa cenie utworu i tylko wtedy następuje transakcja. Występują dwa rodzaje klientów: *Skapiec*, który zawsze chce kupić za 1PLN oraz *Cierpliwy*, który ponawia wysyłanie propozycji kupna zwiększając cenę o 1PLN, aż do pomyślnego zakończenia.

Grajek działa w nieskończonej pętli. W każdym przebiegu najpierw pyta kompozytorów czy chcą coś przesłać, odbiera od nich kolekcję utworów (metoda *dajUtwory*), a następnie wysyła utwory do recenzji. Najpierw oceniane są utwory kompozytorów wyłącznych, a później pozostałych. Po zakończeniu, system sprawdza czy klienci chcą coś kupić. Wysyła zapytanie do klienta *oferta:kolekcjaUtworow od:grajek*, klient ewentualnie odpowiada systemowi serią propozycji *sprzedaj:utwor za:pieniadze* (wynikiem metody *sprzedaj:za:* jest true wtw gdy transakcja powiodła się). Po zakończeniu tej fazy system sprawdza, czy kompozytorzy mogą zmienić się w wyłącznych bądź recenzentów i ewentualnie wykonuje przekształcanie. Na koniec wypisywane są: ilość transakcji, zysk w tej iteracji, całkowity zysk.

1. Zaprojektuj w Smalltalku klasy służące do symulacji systemu Grajek, wskazując klasy z ich atrybutami i metodami oraz związki między klasami. Nie zapomnij o opisach.
2. Zaimplementuj
  - główną pętlę systemu,
  - metodę *oferta:kolekcjaUtworow od:grajek* dla klientów
  - metodę *sprzedaj:utwor za:pieniadze* dla Grajka
  - metodę *dajRecenzje:utwor* dla recenzentów (zaproponuj rozwiązanie np. zależne od gatunku utworu) oraz wszystkie niestandardowe metody potrzebne do zaimplementowania powyższych.