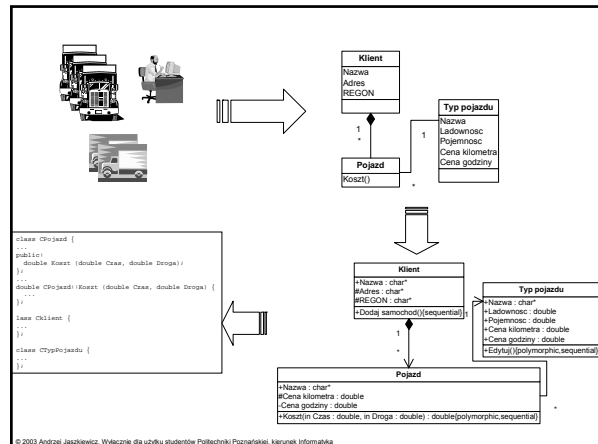


Projektowanie

- Czy oprogramowanie jest zawsze projektowane?
- Znaczenie projektowania
- Projektowanie a wyniki fazy analizy

© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka



© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Znaczenie notacji graficznych w projektowaniu

- Ułatwienie projektowania
- Komunikacja
- Podstawa (zautomatyzowanej) implementacji

© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Kryteria jakości projektu

- Modularność
 - Silne (spójne) składowe
 - Niezależne (luźno powiązane) składowe
- Przejrzystość
 - Modularność
 - Nazewnictwo
 - Złożoność składowych

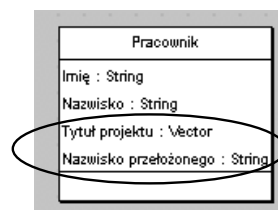
© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Spójność obiektowa

- Pola klasy służą do opisu pojedynczego bytu, a metody klasy operują na (wykorzystują, modyfikują) polach tej klasy

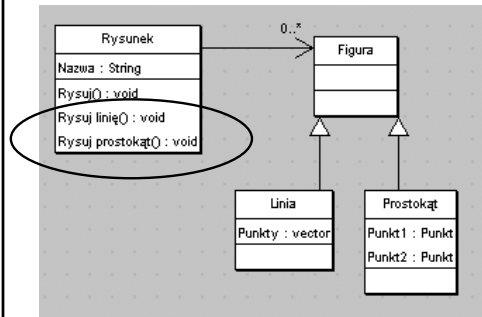
© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Brak spójności obiektowej



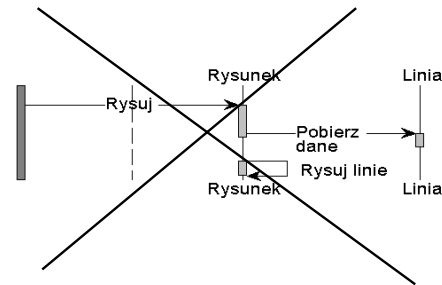
© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Brak spójności obiektowej



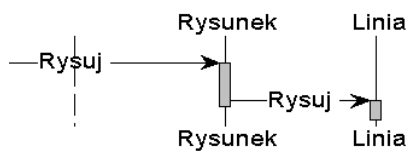
© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Dostęp do danych zwiększa zależność składowych



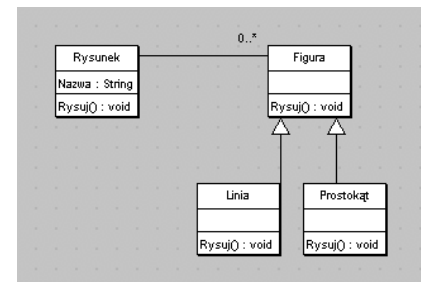
© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Lepiej



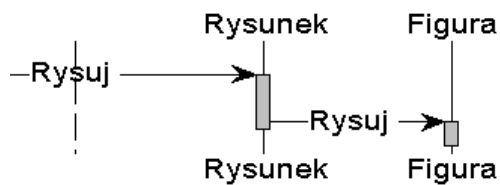
© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Uniezależnienie klasy Rysunek od figur



© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Uniezależnienie klasy Rysunek od figur



© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Złożoność klasy

- Suma złożoności metod
- Miary złożoności metod:
 - Liczba linii kodu
 - Liczba warunków
 - Złożoność cyklomatyczna - liczba niezależnych ścieżek w grafie przepływu sterowania programu

© 2003 Andrzej Jaskiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Zadania w fazie projektowania

- Uszczegółowienie wyników analizy
- Projektowanie technicznych składowych oprogramowania - innych niż składowa dziedzina problemu
- Poprawa efektywności systemu
- Określenie fizycznej struktury systemu

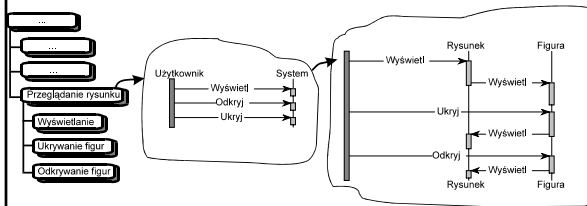
© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Uszczegółowienie wyników analizy

- Opracowanie diagramów sekwencji
- Wprowadzenie związków skierowanych
- Określenie sposobu implementacji związków
- Dobór typów danych
- Opracowanie nagłówków operacji
- Opracowanie algorytmów metod

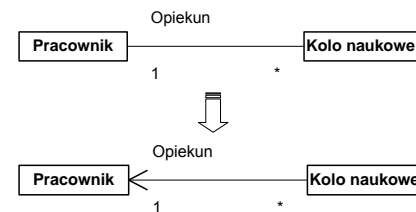
© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Opracowanie diagramów sekwencji



© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Wprowadzenie związków skierowanych



© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Wprowadzenie związków skierowanych

- Czynniki brane pod uwagę
 - Nadmiarowość danych w przypadku związków nieskierowanych (obustronnie nawigowalnych)
 - Efektywność
 - Dostęp do pól i metod powiązanych obiektów

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Określenie sposobu implementacji związków

- Wybór spośród omówionych wcześniej możliwości
- Najlepszym podejściem jest przyjęcie standardu dla projektu/firmy

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Dobór typów danych

- Typy danych dla:
 - Pól
 - Parametrów
 - Metod
- Wybór typu spełniającego wymagania (np. zakres, dokładność, liczba możliwych wartości) określonych na etapie analizy
- Jeżeli żaden z typów standardowych i bibliotecznych nie spełnia naszych wymagań to definiowanie własnych typów:
 - Złożonych
 - Obiektowych (klas) – inne wykorzystanie programowania obiektowego

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Opracowanie nagłówków operacji

- Zamiana danych wejściowych i wyjściowych na parametry i wyniki o odpowiednich typach
- Np. operacja obliczania kosztu przewozu
 - Dane wejściowe Czas, Droga
 - Dany wyjściowe Koszt przewozu
 - double Koszt (double Czas, double Droga)
 - lub
 - Koszt (double Czas, double Droga, double &Koszt)

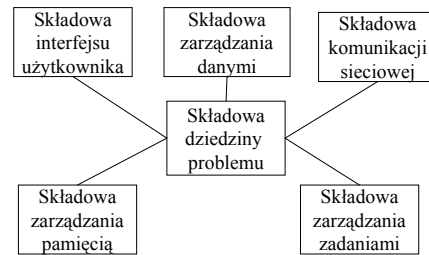
© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Opracowanie algorytmów metod

- ~90% metod to metody algorytmicznie proste
- Algorytmy ~10% metod mają istotnie znaczenie dla systemu (np. efektywność, przejrzystość)

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Projektowanie technicznych składowych oprogramowania



© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Projektowanie technicznych składowych

- W ogólności składowe techniczne mogą być projektowane tak, jak składowa dziedziny problemu - Rzeczywistość -> Model -> Projekt
- W praktyce (choć nie zawsze) projektowanie tych składowych ma charakter powtarzalny, wykorzystujemy więc:
 - Gotowe biblioteki
 - Generatory kodu
 - Wzorce projektowe

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Składowa interfejsu użytkownika

- Zasady projektowania IU
 - Spójność
 - Skróty dla doświadczonych użytkowników
 - Potwierdzanie przyjęcia polecenia użytkownika
 - Prosta obsługa błędów
 - Odwoływanie akcji
 - Wrażenie kontroli nad systemem
 - Nieobciążanie pamięci krótkotrwałej
 - Grupowanie powiązanych operacji
 - Przestrzeganie zasady 7±2
 - Komunikaty o błędach:
 - Jakie są efekty błędów
 - Co użytkownik może zrobić

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Projektowanie składowej zarządzania danymi

- Dokumenty vs. na bieżąco wybierane dane
- Bazy danych - pliki
- Relacja/plik na klasę - wszystkie obiekty w jednym pliku/relacji
- Odczyt/zapis na bieżąco - na żądanie
- Projektowanie baz danych - normalizacja, indeksy

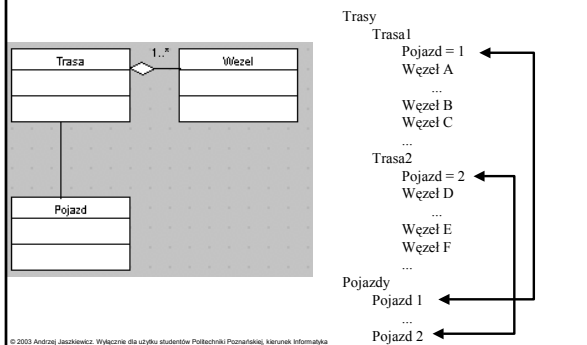
© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

XML jako narzędzie realizacji zarządzania danymi

- Plik XML - zbiór danych - obiektów zawierających się i wzajemnie powiązanych
- DTD, X Schematy - sposób definiowania struktury dokumentu XML
- Narzędzia dla XML - np. parsery, translatorzy, współpraca z bazami danych

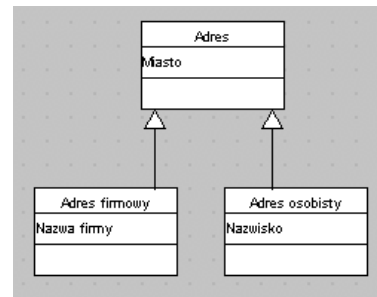
© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Model klas a format XML



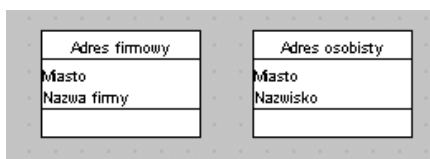
© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Problem dziedziczenia



© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Propagacja pól w dół



Adres firmowy

Poznań	ABC S.A.
Kielce	XYZ o.o.

Adres osobisty

Warszawa	Jan Kowalski
Wólka	Zenon Nowak

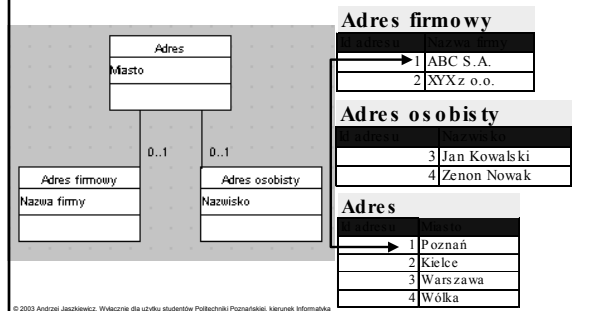
© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Propagacja pól w górę



© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunku Informatyka

Transformacja do zwykłego związku



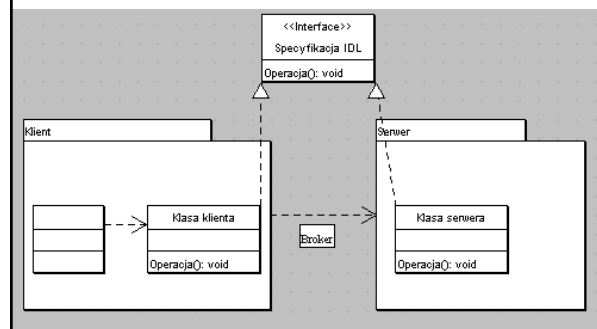
Sposoby implementacji składowej komunikacji sieciowej

- Funkcje niskopoziomowe
- CORBA - (*Common Object Request Broker Architecture*)
- Web services - np. SOAP (*Simple Object Access Protocol*)

CORBA jako narzędzie implementacji składowej komunikacji sieciowej

- Możliwość komunikacji rozproszonych obiektów napisanych w różnych językach oprogramowania
- Specyfikacja klas w języku IDL
- Automatyczna transformacja IDL do kodu komunikacji sieciowej w odpowiednim języku

Struktura CORBY



Poprawa efektywności systemu

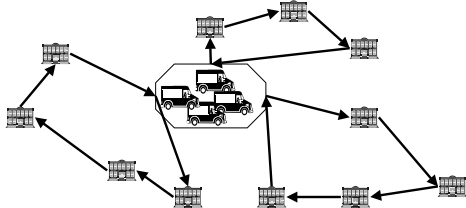
- Kompromis pomiędzy efektywnością a jakością projektu
- Poprawa efektywności na poziomie programistycznym i projektowym
 - Dobór struktur danych
 - Dobór algorytmów

Typowe sposoby poprawy efektywności

- Przechowywanie danych pośrednich
- Indeksowanie

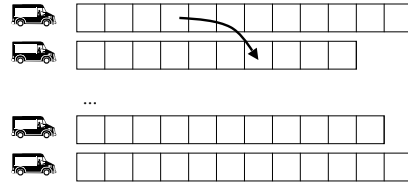
Problem marszrutyzacji pojazdów - (*vehicle routing problem VRP*)

- Należy odwiedzić wszystkie wierzchołki korzystając ze zbioru pojazdów



© 2003 Andrzej Jasiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Rozwiązanie i lokalny ruch/modyfikacja rozwiązania



© 2003 Andrzej Jasiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Koncepcja sąsiedztwa

- Sąsiedztwo** $\mathcal{N}(x)$ - zbiór rozwiązań, które można uzyskać lokalnie modyfikując x
- Ruch** m - lokalna modyfikacja x do y , $y = m(x)$
- $\mathcal{M}(x)$ - zbiór ruchów, które można zastosować do x
- $\mathcal{N}(x) = \{y | \exists m \in \mathcal{M}(x) | y = m(x)\}$
- Ruchy powinny modyfikować rozwiązania lokalnie w przestrzeni rozwiązań i funkcji celu
- Rozmiar $\mathcal{M}(x)$ ($\mathcal{N}(x)$) powinien być dużo mniejszy od całej przestrzeni rozwiązań

© 2003 Andrzej Jasiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Lokalne przeszukiwanie w wersji strömej

Wygeneruj rozwiązanie x

powtarzaj

znajdź najlepsze rozwiązanie $y \in \mathcal{N}(x)$

jeżeli $f(y) > f(x)$ to

$x := y$ (zaakceptuj y)

dopóki nie znaleziono lepszego rozwiązania po przejrzaniu całego sąsiedztwa

© 2003 Andrzej Jasiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Złożoność jednej iteracji lokalnego przeszukiwania

- W - liczba wierzchołków
- T - liczba pojazdów/tras
- Liczba ruchów $|\mathcal{M}(x)| = W(T-1)$
- Nakład na stworzenie rozwiązania sąsiedniego - T
- Złożoność jednej iteracji - $T W(T-1)$
- Zakładamy, że utworzenie trasy od podstaw jest tak samo czasochłonne jak jej modyfikacja - traktowane jako operacja elementarna

© 2003 Andrzej Jasiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Efektywne ocenianie ruchów

- Większość ruchów jest tylko oceniana, a nie wykonywana
- Ocena ruchu wymaga zmodyfikowania dwóch tras
- Nakład na ocenienie ruchu - 2
- Złożoność jednej iteracji - $2 W(T-1)$

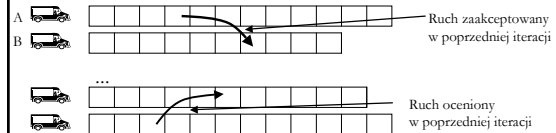
© 2003 Andrzej Jasiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Wykorzystanie ocen ruchów z poprzednich iteracji

- Dla $y \in M(x)$ z reguły $M(x) \cap M(y) = \emptyset$ lub $|M(x) \cap M(y)| \ll |M(x)|$
- Ale $M(x) \cap M(y)$ może być liczny tzn. wiele ruchów pozostaje taka sama po wykonaniu ruchu

© 2003 Andrzej Jaszkiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Wykorzystanie ocen ruchów, z poprzednich iteracji



- Oceny wszystkich ruchów nie dotyczących tras A i B pozostają te same

© 2003 Andrzej Jaszkiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Wykorzystanie ocen ruchów z poprzednich iteracji dla VRP

- Każdy ruch modyfikuje tylko dwie trasy
- Wszystkie ruchy pomiędzy niezmodyfikowanymi trasami pozostają takie same
- Po zaakceptowaniu nowego rozwiązania, należy więc ocenić średnio tylko $2W(T-1) / T \approx 2W$ ruchów

© 2003 Andrzej Jaszkiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Efekt

- Poprawa efektywności (z wyjątkiem pierwszej iteracji) w stosunku do wersji podstawowej

$$TW(T-1) / 2 \quad W = T(T-1) / 2$$

© 2003 Andrzej Jaszkiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Przykład – lokalne przeszukiwanie z zakazem powrotu do poprzednich rozwiązań

- Problem komiwojażera – 100 miast
- Zapamiętanie jednego rozwiązania 100 B
- Milion rozwiązań ~100 MB
- Duży czas weryfikacji czy nowe rozwiązanie znajduje się na liście uprzednio wygenerowanych rozwiązań.

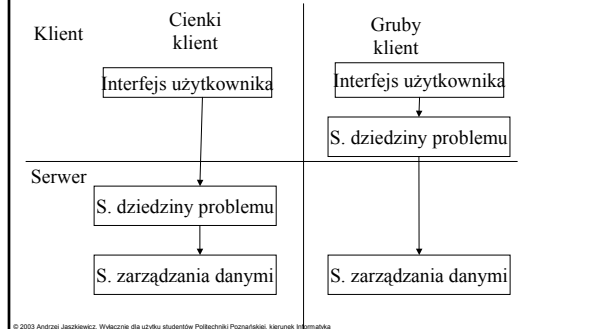
© 2003 Andrzej Jaszkiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Określenie fizycznej struktury systemu

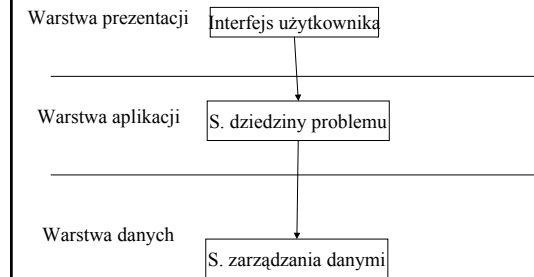
- Rozmieszczanie klas w modułach
- Podział na programy wykonywalne
- Projektowanie systemów rozproszonych

© 2003 Andrzej Jaszkiewicz. Wyłączone dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Architektura klient-serwer



Architektura trójwarstwowa



Metoda CRC cards

- *Class-Responsibility-Collaboration* - Klasa-Odpowiedzialność-Współpraca
- Może poprzedzać opracowanie diagramów sekwencji lub samodzielnie wspomagać wprowadzanie metod

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Przykład karty CRC

Prostokąt	
Odpowiedzialność	Współpraca
Rysowanie	Wzór linii Wzór wypełnienia
Edycja interaktywna	Manipulator prostokąta
Zaznaczanie się	

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka

Przykład karty CRC

Pojazd	
Odpowiedzialność	Współpraca
Obliczanie kosztów	Kategoria pojazdu
Sprawdzanie wykonalności zlecenia	Kategoria pojazdu

© 2003 Andrzej Jaszkiewicz. Wyłączenie dla użytku studentów Politechniki Poznańskiej, kierunek Informatyka