

Laboratorium nr 14

Temat: Złożenia: obiekty jako składowe klas.

Zakres laboratorium:

- obiekty jako składowe klas
- odwieczny problem: być czy mieć?
- konstrukcja obiektu, którego składnikiem jest obiekt innej klasy
- przykład złożenia obiektów
- zadanie laboratoryjne

Obiekty jako składowe klas

- **Złożenie (agregacja, zawieranie)** (ang. *has a relationship*) polega na tym, że jedna klasa zawiera jako składowe obiekty innych klas. Mówiąc inaczej, obiekt będący całością składa się z określonej liczby obiektów-składników.

Odwieczny problem: być czy mieć?

- Rozróżnienie pomiędzy **dziedziczeniem** a **zawieraniem** może czasami nastęrczać pewnych trudności.
- Jeżeli relację pomiędzy dwoma obiektami lepiej opisuje określenie „**ma**” („zawiera”, „składa się” itp.), to należy zastosować **agregację**. Kiedy natomiast klasy są naturalnie połączone poprzez stwierdzenie „**jest**”, wtedy odpowiedniejszym rozwiązaniem jest **dziedziczenie**.

Przykład:

Klasa1 **jest** rodzajem Klasa2
Klasa1 **zawiera obiekt** typu Klasa2



Kwadrat **jest** rodzajem Figury
Samochód **zawiera obiekt** typu Koło

Konstrukcja obiektu, którego składnikiem jest obiekt innej klasy

- Konstruktory obiektów składowych wykonują się najpierw, a dopiero potem rusza do akcji konstruktor głównego obiektu.
- Obiekt jakiejś klasy będący składnikiem innej klasy może być inicjalizowany tylko za pomocą listy inicjalizacyjnej!!!
- Nie można próbować uruchomić konstruktora obiektu składowego z wnętrza konstruktora obiektu głównego.
- Jeśli obiekt składowy nie ma konstruktora, to na liście inicjalizacyjnej nie umieszcza się tego obiektu.
- Jeśli klasa obiektu składowego ma konstruktor wywoływany bez żadnych argumentów (czyli **konstruktor domniemany**), i ten konstruktor chcemy użyć, to można tę pozycję na liście inicjalizacyjnej pominąć.
- Jeśli klasa obiektu składowego ma konstruktory (wszystkie z jakimiś argumentami), to pominięcie wywołania na liście inicjalizacyjnej spowoduje błąd kompilacji.
- Jeżeli jedną ze składowych klasy jest obiekt innej klasy, zadeklarowanie go jako składowej publicznej nie koliduje z hermetyzacją i ukrywaniem informacji dotyczącej prywatnych danych tego obiektu.
- Destruktry wywoływane są w kolejności odwrotnej niż konstruktory. Najpierw rusza do pracy destruktor głównego obiektu (klasy zawierającej obiekty), a dopiero potem wykonywane są destruktry obiektów składowych.

Przykład złożenia obiektów

```
class Volkswagen
{
private:
    Silnik diesel; //obiekt klasy Silnik
    Kolo zapasowe; //obiekt klasy Kolo
    char *nazwa;
    float cena;
public:
    Volkswagen(char *, float, int, int, char *, char *); //konstr. obiektu głównego
    ...
};
/////////////////////////////////////////////////////////////////
class Silnik
{
private:
    int moc;
    int cylindry;
public:
    Silnik(int=0, int=4);
    ...
};
/////////////////////////////////////////////////////////////////
class Kolo
{
private:
    char *felga;
    char *opona;
public:
    Kolo(char="brak", char="brak");
    ...
};
/////////////////////////////////////////////////////////////////
Volkswagen::Volkswagen(char *model, float ce, int kW, int cyl, char *fel, char *op) //lista inicjalizacyjna
:diesel(kW, cyl), zapasowe(fel, op)
{
    nazwa=model;
    cena=ce;
}
/////////////////////////////////////////////////////////////////
main()
{
    Silnik s1;
    Kolo k1;
    Volkswagen Golf("Golf V", 65499, 86, 4, "aluminiowa", "letnia"); //tworzymy obiekt złożony
    Volkswagen Golf3("Golf III", 20000, s1, k1); //lub w inny sposób
}
```

Zadanie laboratoryjne