

**Pamiętaj o:**

- Stworzeniu osobnych plików (header.h, source.cpp, main.cpp) każdego projektu.
- Usuwanu dynamicznie tworzonych obiektów.

*Etykiety dostępu składników w każdej z klas ustaw jako **private**.*

**1. Modyfikacja programu z tematu „Konstruktor, destruktor...”**

**Stwórz:**

- Klasę TSilnik ze składnikami: moc, max\_moment\_obr.
- Klasę TSamochód ze składnikami: silnik (typu TSilnik), string model.
- Zaprzyjaźnij ze sobą klasy, tak żeby klasa TSamochód miała dostęp do prywatnych składników klasy TSilnik.
- Stwórz konstruktor klasy TSilnik z argumentami: moc, moment i zainicjalizuj zmienne w ciele konstruktora.
- Stwórz konstruktor klasy TSamochód z argumentami: MODEL, MOC, MOMENT. Inicjalizację przeprowadź przy użyciu listy inicjalizacyjnej.
- Stwórz funkcję składową klasy TSamochód do wypisywania informacji o modelu, mocy i maksymalnym momencie obrotowym.
- W funkcji głównej stwórz dynamicznie obiekt typu Samochód i wypisz informacje na jego temat na ekran.

2 pkt.

**2. Przyjaźń:**

- Napisz klasę TKlasa ze składnikiem typu int, nadaj mu wartość.
- Zadeklaruj przyjaźń klas, tak żeby TZaprzyjaźniona\_Klasa mogła modyfikować wartość prywatnego składnika TKlasa.
- Zadeklaruj funkcję f\_Zaprzyjaźniona z klasą TKlasa z jednym argumentem.
- W ciele klasy TZaprzyjaźniona\_Klasa zdefiniuj publiczną metodę, która:
  - ❖ wyświetli pierwotną wartość składnika klasy TKlasa,
  - ❖ nada nową wartość składnikowi klasy TKlasa i go wyświetli.
- Zdefiniuj funkcję f\_Zaprzyjaźniona, która zmodyfikuje wartość składnika klasy TKlasa i go wyświetli
- 

Uwaga: Metoda i funkcja przyjmują argument typu Klasa.

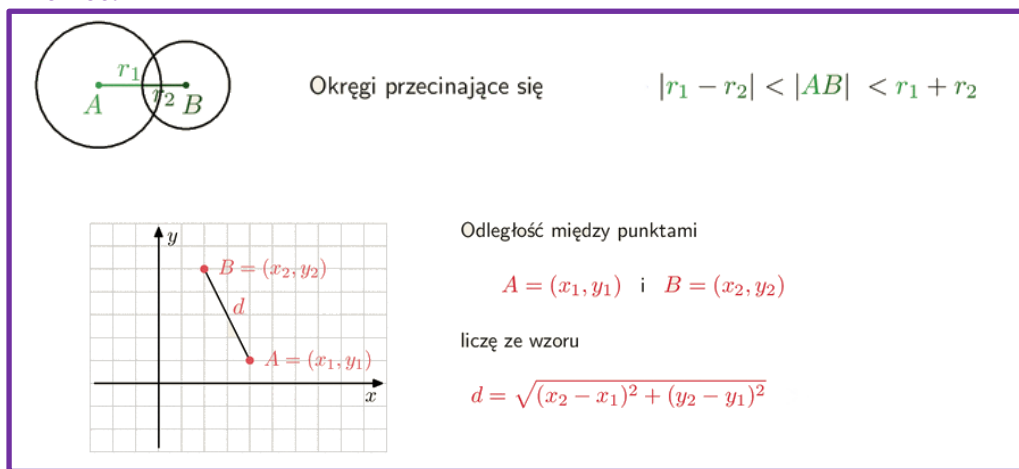
- W funkcji głównej wywołaj metodę (funkcję składową) oraz funkcję.

3 pkt.

### 3. Przyjaźń:

- Napisz klasę *TOkrąg* ze składnikami:
  - \* promień typu float;
  - \* środek typu struktura ze współzrędnymi x oraz y
  - \* Konstruktor z trzema argumentami:
    - a. promień
    - b. współzrędną środka x
    - c. współzrędną środka y
  - \* Zaprzyjaźniona funkcja dwuargumentowa *PrzecinająSię* typu bool;
- Zdefiniuj dwuargumentową (okrąg1, okrąg2) funkcję *PrzecinająSię*.

#### Pomoc:



- Za pomocą warunku wyświetl komunikat o przecinaniu się, bądź nie okręgów.
- W funkcji głównej:
  - \* Stwórz dwa obiekty typu *TOkrąg*, nadaj im wartości
  - \* Wywołaj funkcję *PrzecinająSię*.

4,0 pkt.

Suma: 9 pkt.