

## Zadanie 9

---

Uzupełnić podane klasy i funkcje, tak aby zadziałała podana niżej funkcja main:

```
#include <iostream>
#include <cstring>
using namespace std;

class Person {
    char* name;
public:
    friend class Couple;
    friend ostream& operator<<(ostream& str,
                                const Person& os);

    explicit Person(const char* n);
    Person(const Person& os);
    Person& operator=(const Person& os);
    ~Person();
};

// implementacja konstruktorow,
// destruktor i metod klasy Person

class Couple {
    Person *husband, *wife;
public:
    friend ostream& operator<<(ostream& str,
                                const Couple& p);

    Couple(const Person& m , const Person& z);
    Couple(const Couple& p);
    Couple& operator=(const Couple& p);
    ~Couple();
};

// implementacja konstruktorow,
// destruktor i metod klasy Couple

int main(void) {
    Person john("John"), jenny("Jenny");
    Couple cpl1(john,jenny);
    Couple *pcpl2 = new Couple(Person("Kevin"),
                                Person("Kathy"));
```

```

    Couple cpl3(Person("Bill"),Person("Betty"));
    cpl3 = cpl3; // sic!
    *pcpl2 = cpl3;
    cout << "John : " << john << endl;
    cout << "Jenny: " << jenny << endl;
    cout << *pcpl2 << endl;
    delete pcpl2;
}

```

Po uzupełnieniu implementacji program powinien się skompilować i wypisać coś w rodzaju

```

John : John
Jenny: Jenny
Couple: he Bill, she Betty

```

---

*Termin: do 28 czerwca (włącznie)*

---

Rozwiązania, w postaci **jednego** pliku źródłowego zawierającego treść programu, proszę wrzucać w systemie EDU do katalogu „Foldery zadań / Zadanie\_09”

Nazwą pliku powinno być nazwisko z dużej litery (bez polskich znaków); rozszerzeniem musi być `.cpp`, czyli np. `Malinowska.cpp`.