

Problem 1

Complete classes and functions, so the **main** function below compiles and runs (don't remove **delete** instructions). The destructor of **Person** should print something, so we know when objects of this class are deleted.

[download Couples.cpp](#)

```
#include <iostream>
#include <cstring>

class Person {
    char* name;
    friend class Couple;
public:
    explicit Person(const char* n);
    Person(const Person& os);
    Person& operator=(const Person& os);
    ~Person();
    friend std::ostream& operator<<(std::ostream& str,
                                    const Person& os);
};

// implementation

class Couple {
    Person *wife, *husband;
public:
    Couple(const Person& she, const Person& he);
    Couple(const Couple& other);
    Couple& operator=(const Couple& other);
    ~Couple();
    friend std::ostream& operator<<(std::ostream& str,
                                    const Couple& p);
};

// implementation

int main(void) {
    using std::cout; using std::endl;
    Person *pjohn = new Person("John"),
            *pjane = new Person("Jane");
    Person mary("Mary"), mark("Mark");
    Couple *pcouple1 = new Couple(mary, *pjohn);
    Couple couple2(*pjane, mark);
```

```

    delete pjohn;
    delete pjane;
    cout << *pcouple1 << endl;
    cout << couple2 << endl;
    couple2 = *pcouple1;
    delete pcouple1;
    cout << couple2 << endl;
}

```

After supplying an implementation, the program should compile and print something like

```

DEL John
DEL Jane
Couple: he John, she Mary
Couple: he Mark, she Jane
DEL Jane
DEL Mark
DEL Mary
DEL John
Couple: he John, she Mary
DEL Mary
DEL John
DEL Mark
DEL Mary

```
