Class

Classes in C++ are user defined types declared with keyword class that has data and functions . Although classes and structures have the same type of functionality, there are some basic differences. The data members of a class are private by default and the members of a structure are public by default. Along with storing multiple data in a common block, it also assigns some functions (known as methods) to manipulate/access them. It serves as the building block of Object Oriented Programming.

It also has access specifiers, which restrict the access of member elements. The primarily used ones are the following:

- *public:* Public members (variables, methods) can be accessed from anywhere the code is visible.

- *private:* Private members can be accessed only by other member functions, and it can not be accessed outside of class.

Class can be represented in the form of

```
class ClassName {
    access_specifier1:
        type1 val1;
        type2 val2;
        ret_type1 method1(type_arg1 arg1, type_arg2 arg2,...)
        ...
    access_specifier2:
        type3 val3;
        type4 val4;
        ret_type2 method2(type_arg3 arg3, type_arg4 arg4,...)
        ...
};
```

It's a common practice to make all variables private, and set/get them using public methods. For example:

```
class SampleClass {
    private:
        int val;
    public:
        void set(int a) {
            val = a;
        }
        int get() {
            return val;
        }
};
```

We can store details related to a student in a class consisting of his *age (int), first_name (string), last_name (string) and standard (int)*.

You have to create a class, named *Student*, representing the student's details, as mentioned above, and store the data of a student. Create setter and getter functions for each element; that is, the class should at least have following functions:

- *get_age*, *set_age*

- *get_first_name*, *set_first_name*

- *get_last_name*, *set_last_name*

- *get_standard*, *set_standard*

Also, you have to create another method *to_string()* which returns the string consisting of the above elements, separated by a comma(*,*). You can refer to *stringstream* for this.

Input Format

Input will consist of four lines.
The first line will contain an integer, representing the *age*. The second line will contain a string, consisting of lower-case Latin characters ('a'-'z'), representing the *first_name* of a student.
The third line will contain another string, consisting of lower-case Latin characters ('a'-'z'), representing the *last_name* of a student.
The fourth line will contain an integer, representing the *standard* of student.

Note: The number of characters in *first_name* and *last_name* will not exceed 50.

Output Format

The code provided by HackerRank will use your class members to set and then get the elements of the *Student* class.

Sample Input

```
15
john
carmack
10
```

Sample Output

```
15
carmack, john
10

15,john,carmack,10
```