

You can use freely C or C++ syntax to answer the questions.

Question 1

We want to create a program of a simplified directory which associates a name and a telephone number.

1. Define a structure *Person* containing the information (*name* and *tel*). The name can contain 32 characters and the number 16 characters.
2. Define a new structure *Contacts* which will represent the address book. This *Contacts* structure will contain a table of 20 *Person* and a *counter* indicating the number of people in the table.
3. Define a function that takes into argument a name and a telephone and returns an instance of *Person*.
4. Define a function which adds a person to a *Contacts* instance (if there is still a free place).

Question 2

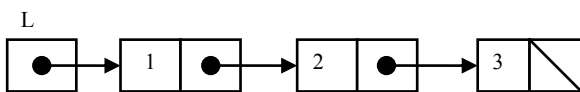


Figure 1: classic linked list

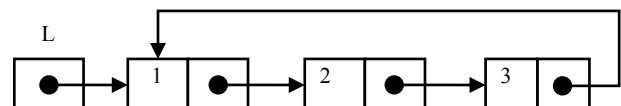


Figure 2: circular linked list

Figure 1 shows a classic linked list of three elements. Figure 2 shows a circular linked list also composed of three elements. The question concerns circular linked lists of arbitrary number of elements. In such lists the last element points to the first instead of pointing to NULL. Suppose the list is already created. We define:

```
struct ListeC {  
    int elt;  
    ListeC* next;  
};
```

1. Write the code of *void printCL (ListeC* L)* which prints on the screen the elements (elt) of the circular chained list pointed by L. (pay attention not to write infinite loops!)

Good luck