

Esercizio

Sia data una struttura

```
typedef struct
{ char nome[30];
  char cognome[30];
} element;
```

```
typedef struct list_element {
element value;
struct list_element *next;
} item;
```

```
typedef item* list;
```

che rappresenta un elemento di una lista di nominativi. Si scrivano le seguenti funzioni:

- a. list insord(element e, list l) che, dati il puntatore ad una lista ordinata di elementi ed una persona, inserisce l'elemento nella lista mantenendo l'ordinamento. La funzione restituisce un puntatore alla nuova lista creata. Il criterio di ordinamento è lessicografico, ordinando per cognome e, a parità di cognome, per nome.
- b. void contaOmonimi(list l) che, dato il puntatore ad una lista ordinata di elementi di tipo element, visualizza il numero di omonimi (cioè persone con lo stesso nome e cognome) presenti nella lista.

```
#include <stdio.h>
#include <stdlib.h>
```

```
typedef struct
{ char nome[30];
  char cognome[30];}
element; /* DEFINIZIONE */
```

```
typedef int boolean; /* DEFINIZIONE */
```

```
typedef struct list_element {
element value;
struct list_element *next;
} item;
typedef item* list;
```

```
boolean isLess(element, element);
boolean isEqual(element, element);
element copy(element e);
element getElement(void);
void printElement(element);
```

```
/* ---- PRIMITIVE ---- */
list emptylist(void);
boolean empty(list);
element head(list);
list tail(list);
list cons(element, list);
/* ---- NON PRIMITIVE ---- */
void showlist(list);
```

```
boolean isEqual(element e1,element e2) {
    if (strcmp(e1.cognome,e2.cognome) == 0
        && strcmp(e1.nome,e2.nome) == 0)
        return 1;
    else return 0;
}
```

```
boolean isLess(element e1, element e2) {
    if (strcmp(e1.cognome,e2.cognome) < 0 ||
```

```

                (strcmp(e1.cognome,e2.cognome) == 0 &&
                 strcmp(e1.nome,e2.nome) < 0))
            return 1;
        else return 0;
    }

    element copy(element e){
        element el;
        el=e;
        return el;
    }

    element getElement(){
        element el;
        printf("Cognome: ");
        scanf("%s", el.cognome);
        printf("Nome: ");
        scanf("%s", el.nome);
        return el;
    }

    void printElement(element el){
        printf("\nCognome: %s, Nome: %s", el.cognome, el.nome);
    }

    list emptylist() {
        return NULL;
    }

    boolean empty(list l) {
        return (l==NULL);
    }

    element head(list l) {
        if (empty(l)) abort();
        else
            return l->value;
    }

    list tail(list l) {
        if (empty(l)) abort();
        else

```

```

        return l->next;
    }

```

```

list cons(element e, list l) {
    list t;
    t = (list) malloc(sizeof(item));
    t->value=copy(e);
    t->next=l;
    return t;
}

```

```

void showlist(list l) {

    printf("Elenco Persone");
    while (!empty(l)) {
        printElement(head(l));
        l = tail(l);
        //if (!empty(l)) printf(", ");
    }
    printf("\n");
}

```

```

list insord(element e, list l){

    list t;

    if(empty(l) || !isLess(head(l), e)){

        t = cons(e, l);
        //t->next = l;
        return t;
    }

    return cons(head(l), insord(e, tail(l)));
}

```

```

list insord2(element e, list l){

    list l1 = NULL, root = l;

```

```

list t;

t = cons(e, l);

if(empty(l) || !isLess(head(l), e)){
    return t;
}

while(!empty(l) && isLess(head(l), e)){

    l1=l;
    l = tail(l);
}

l1->next = t;
t->next = l;

return root;
}

void contaOmonimi(list l) {

    int n=0;
    element e1;

    if (empty(l)) return;

    do {
        e1 = head(l);
        l = tail(l);
        if (!empty(l) && isEqual(head(l),e1)) {
            n++;
            while (!empty(l) && isEqual(head(l),e1))
                l = tail(l);
        }
    } while(!empty(l));

    printf("Gli omonimi sono: %d", n);
}

```

```
void main(void) {  
    list l1 = emptylist();  
    element el;  
    int scelta;  
  
    do {  
        printf("\nIntrodurre Persona:\n");  
        el=getElement();  
        l1 = insord2(el, l1);  
        printf("\n 0 per terminare, 1 per continuare:\n");  
        scanf("%d", &scelta);  
        //getch();  
    } while (scelta==1); /* condiz. arbitraria */  
    showlist(l1);  
  
    contaOmonimi(l1);  
  
    getch();  
    getch();  
  
}
```