

Esercizio 1

Sia data la struttura seguente:

```
struct elem {int info; elem* pun;};
```

Scrivere una funzione ricorsiva che, data una lista di elementi di tipo `elem` ed un intero `n`, inserisce un elemento in fondo alla lista con campo `info` uguale a `n` se non è già presente nella lista un tale elemento; altrimenti la funzione lascia la lista inalterata.

Esercizio 2

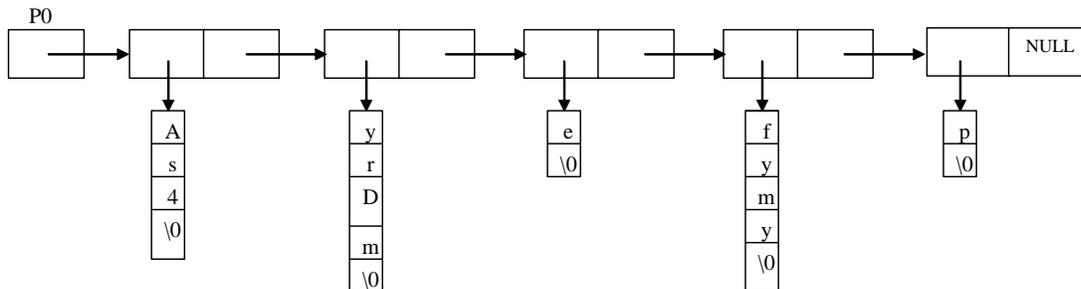
Sia data la struttura seguente:

```
struct elem {char* nome; elem* pun;};
```

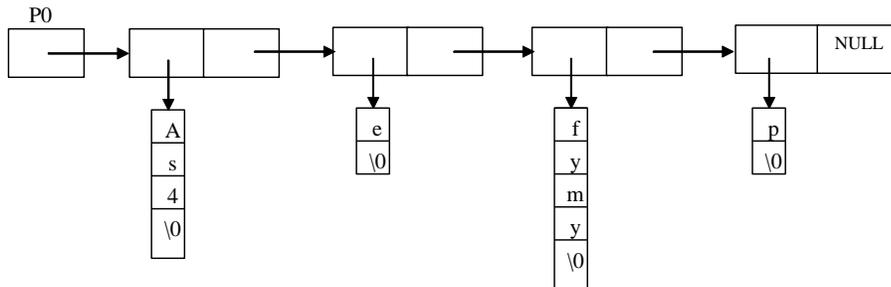
Scrivere una funzione che, dato il file di nome "input.txt" ed una lista di elementi di tipo `elem`, legge dal file prima la dimensione di una stringa e poi la stringa stessa, ed elimina dalla lista il primo elemento con campo `nome` uguale alla stringa letta. Si supponga che i dati nel file siano corretti. Per esempio, se il contenuto del file "input.txt" è:

```
4  
yrDm
```

la lista seguente:



viene modificata dalla funzione come segue:



Esercizio 3

Sia dato il seguente enumerato

```
enum casella {B, N};
```

Scrivere una funzione che, dato un intero positivo `n`, restituisca una matrice di dimensione `n x n` di caselle disposte come una scacchiera. Per esempio, se `n=4`, la matrice restituita è la seguente:

```
B N B N  
N B N B  
B N B N  
N B N B
```

SOLUZIONI

Esercizio 1

```
void aggiungi(elem*& p0, int n){
    if (p0==NULL) {
        p0=new elem;
        p0->info=n;
        p0->pun = NULL;
    }

    if (p0->info ==n)
        return;
    aggiungi(p0->pun, n);
}
```

Esercizio n 2

```
void elimina(elem*& p0) {
    ifstream in ("input.txt");
    if (!in)
        return;
    int l;
    in >> l;
    char* r = new char[l+1];
    in >>r;

    elem*p, *q;
    for(q=p0; q!=NULL && strcmp(q->nome, r); q=q->pun)
        p=q;

    if (q==NULL)
        return;
    if(q==p0)
        p0=p0->pun;
    else
        p->pun = q->pun;
    delete[]q->nome ;
    delete q ;
    delete[]r;
}
```

Esercizio 3

```
casella* crea(int n) {
    if (n<=0)
        return NULL;
    casella* mat = new casella[n*n];
    for (int i=0; i<n; i++)
        for(int j = 0; j < n; j++)
            if (i%2)
                mat[i*n+j] = (j%2)? B : N;
            else
                mat[i*n+j] = (j%2)? N : B;

    return mat;
}
```