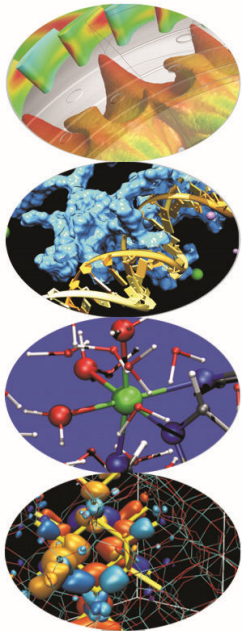
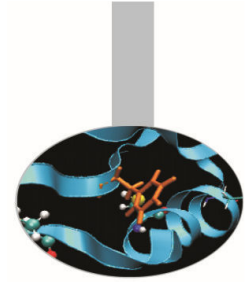


# Esercitazioni dati strutturati





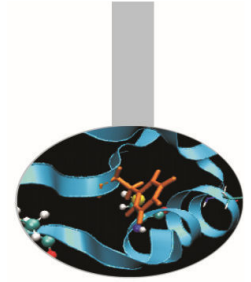
## Esercizio `coda_testo`

Partendo dall'esempio `e-coda_2.c` implementare il programma `coda_testo.c` con una coda che contenga righe di testo anziché interi.

## Esercizio `complex-stru`

Scrivere il programma `complex-stru.c` che implementa un numero complesso usando una struttura contenente i due membri `r` e `i`.

Implementare le funzioni *addition*, *subtraction* e *print\_bra* (quest'ultima per stampare a video un numero complesso tra parentesi tonde).

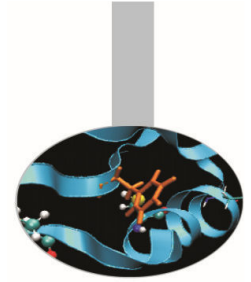


## Esercizio `pagina_semplice`

Partendo dal template `t-pagina_semplice.c` completare il programma aggiungendo il codice mancante.

## Esercizio `pagina_web`

Usando come base il programma `pagina_semplice` sostituire la componente Paragrafo con un oggetto di tipo Coda (implementata nell'esercizio `coda_testo`).



## Esercizio **area-figure2D**

Definire il tipo **Figura2D** associato a una struttura costituita da:

Tipo2D t – enumeratore contenente {Segmento, Cerchio, Triangolo}

double v – da usare per il raggio del cerchio

Punto p[4] – per definire i vertici delle figure istanziate

Il tipo **Punto** è costituito da una coppia (x,y) di valori double.

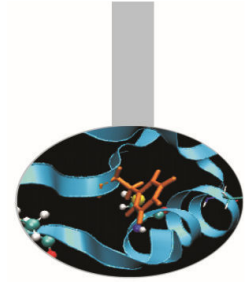
Implementare quindi le funzioni:

```
double area(Figura2D f);          void inputFigura2D(Figura2D *f);
```

e il programma **area-figure2D.c** per testarle.

L'area di un triangolo può essere calcolata partendo dalle coordinate (x,y) dei vertici del triangolo con la formula:

```
fabs( ( y1 - y0)*( x2-x1) + ( y1-y2) * ( x1-x0) ) / 2.0;
```



## Esercizio **area-figure2D-2**

Estendere il tipo **Figura2D** aggiungendo la possibilità di manipolare quadrati.

Come dev'essere cambiato l'enumeratore **Tipo2D**?