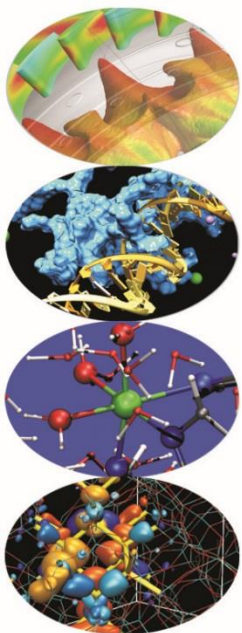
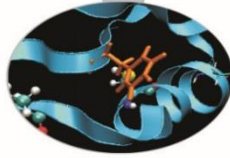


# Esercitazioni





## Esercizio 1

Implementare una classe template array che contiene come membri private:

```
int size;
```

```
T *myarray;
```

E come metodi

public:

```
array (int s) // costruttore che alloca memoria per myarray alla dimensione s
```

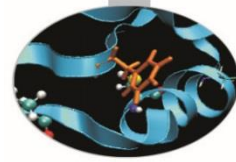
```
~array() // distruttore
```

```
void setArray ( int elem, T val) // per impostare a val l'elemento in posizione elem
```

```
void getArray () // stampa il tipo e gli elementi dell'array
```

```
void replaceElem(T newE, T oldE) //per sostituire il valore oldE con newE
```

```
void findElem(T val, int &index) // per trovare la prima occorrenza di val
```

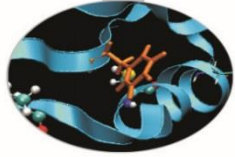


## **Esercizio 1 STL**

Definire una classe Point3D per la rappresentazione di un punto. La classe dispone degli attributi x,y,z e weight.

Creare un vettore (vector) di Point3D.

- Stampare il vettore ( fare overloading << per Point3D).
- Ordinare il vettore con la funzione sort sulla base della coordinata z (definire una funzione friend per l'ordinamento su z)
- Trovare il massimo e il minimo del vettore (max\_element e min\_element, fare overloading delle funzioni < e > in base alla variabile weight)
- Stampare il vettore ordinato con la funzione reverse
- Trovare il punto Point(1,1,1,20) con la funzione find.



## **Esercizio 2 STL**

Scrivere un programma che legga da tastiera 10 parole e le stampi in ordine inverso rispetto al loro inserimento.

Scrivere il codice usando tre diversi contenitori dati:

- 1- array di puntatori a char
- 2- vector
- 3- stack