

Ordinare elementi: iniziamo con il dichiarativo ma con efficienza bassa.

```
1 ordinata([H|T],LO):-  
2     permutazione(L,LO),  
3     ordinata(LO).  
4  
5 ordinata([]).  
6 ordinata([_]).  
7 ordinata([X,Y|L]):-  
8     X>=Y,  
9     ordinata([Y|L]).  
10  
11
```

Conosciamo: BubbleSort, QuickSort, MergeSort. Quale è il candidato migliore per il dichiarativo?

Come FUNZIONA il QuickSort? Si presta a essere il migliore nel dichiarativo in quanto è una proprietà.

```
1 quick_sorted([],[]).  
2 quick_sorted([Pivot|L],LO):-  
3     partitioned(L,Pivot,L1,L2),  
4     quick_sorted(L1,L01),  
5     quick_sorted(L2,L02),  
6     append(L01,[Pivot|L02],LO).  
7  
8 partitioned([],_,[],[]).  
9 partitioned([H|L],Pv,[H|L1],L2):-  
10     H < Pv,  
11     partitioned(L,Pv,L1,L2).  
12  
13 partitioned([H|L],Pv,L1,[H|L2]):-  
14     H >= Pv,  
15     partitioned(L,Pv,L1,L2).  
16  
17
```

≡ ?- partitioned([2,4,3,5,9],4,L1,L2).

Singleton variables: [L01]

L1 = [2, 3],

L2 = [4, 5, 9]

≡ ?- quick_sorted([8,4,2,5,19,7,21],L).

L = [2, 4, 5, 7, 8, 19, 21]

false

BubbleSort: non è dedicato per questo tipo di approccio dichiarativo, o meglio si può fare ma non sfruttiamo il vero concetto delle bolle che salgono, per implementarlo bisogna farlo ricorsivamente quindi procedurale e dunque una vera e propria funzione e non un predicato.

1 Your Prolog rules and facts go here ...

