

Estensione del countingSort che permette di ordinare in tempo lineare coppie (chiave, valore), invece che singoli interi. Le chiavi devono essere comprese fra 1 e k .

```
// ordina un vettore di RECORD in base al campo numerico key associato ad ognuno di essi
```

```
pigeonholeSort(RECORD[] A, int n, int min, int max)
```

```
    int size  $\leftarrow$  max - min + 1
```

```
    LIST() L  $\leftarrow$  new LIST[0...size - 1]
```

```
    da j  $\leftarrow$  1 fino a size fai
```

```
    |   L[j]  $\leftarrow$  LIST
```

```
    // scansione iniziale
```

```
    da i  $\leftarrow$  1 fino a n fai
```

```
    |   LIST M  $\leftarrow$  L[A[i].key - min]
```

```
    |   M.insert(M.tail, A[i])
```

```
    i = 1
```

```
    // scansione vettore B
```

```
    da j  $\leftarrow$  0 fino a size - 1 fai
```

```
    |   Pos p  $\leftarrow$  L[j].head
```

```
    |   finché not L[j].finished(p) fai
```

```
    |   |   A[i]  $\leftarrow$  L[j].read(p)
```

```
    |   |   i++
```

```
    |   |   p  $\leftarrow$  L[j].next
```