

# TECNICHE DI PROGRAMMAZIONE, A.A. 2021/2022

## Esempi di domande di teoria per esame

In questo documento sono raccolti esercizi svolti, risolti e in modalità piattaforma Exam relativi ai seguenti argomenti teorici trattati durante il Corso:

- algoritmi union-find per online connectivity
  - quick-find
  - quick-union
  - weighted quick-union
- algoritmi di ordinamento iterativi quadratici
  - insertion sort
  - bubble/exchange sort
  - selection sort
  - Shell sort
- algoritmi di ordinamento iterativi lineari
  - counting sort
  - radix sort
- algoritmi di ordinamento iterativi linearitmici
  - bottom-up merge sort
- analisi di complessità

## Algoritmi Union-Find per Online Connectivity

### Esercizio svolto: quick-find

Sia data la seguente sequenza di coppie:

3-2 3-4 5-1 7-3 5-7 9-1

dove la relazione  $i-j$  indica che il vertice  $i$  è adiacente al vertice  $j$ . Si applichi un algoritmo di on-line connectivity con quick-find, riportando a ogni passo il contenuto del vettore. I vertici siano denominati con interi compresi tra 0 e 9.

### Soluzione

La tabella riportata di seguito illustra la configurazione del vettore id a ogni passo.

	0	1	2	3	4	5	6	7	8	9
	0	1	2	3	4	5	6	7	8	9
3-2	0	1	2	2	4	5	6	7	8	9
3-4	0	1	4	4	4	5	6	7	8	9
5-1	0	1	4	4	4	1	6	7	8	9
7-3	0	1	4	4	4	1	6	4	8	9
5-7	0	4	4	4	4	4	6	4	8	9
9-1	0	4	4	4	4	4	6	4	8	4

### Esercizio svolto: quick-union

Sia data la seguente sequenza di coppie:

1-3 3-4 5-1 4-0 7-3 5-7 9-4 2-6

dove la relazione  $i-j$  indica che il vertice  $i$  è adiacente al vertice  $j$ . Si applichi un algoritmo di on-line connectivity con quick-union, riportando a ogni passo il contenuto del vettore. I vertici siano denominati con interi compresi tra 0 e 9.

### Soluzione

La tabella riportata di seguito illustra la configurazione del vettore id a ogni passo.

	0	1	2	3	4	5	6	7	8	9
	0	1	2	3	4	5	6	7	8	9
1-3	0	3	2	3	4	5	6	7	8	9
3-4	0	3	2	4	4	5	6	7	8	9
5-1	0	3	2	4	4	4	6	7	8	9
4-0	0	3	2	4	0	4	6	7	8	9
7-3	0	3	2	4	0	4	6	0	8	9
5-7	0	3	2	4	0	4	6	0	8	9
9-4	0	3	2	4	0	4	6	0	8	0
2-6	0	3	6	4	0	4	6	0	8	0

### **Esercizio svolto: weighted quick-union**

Sia data la seguente sequenza di coppie:

3-9 4-3 2-5 6-2 3-8 5-6 0-10 3-5 8-9 10-3

dove la relazione  $i-j$  indica che il vertice  $i$  è adiacente al vertice  $j$ . Si applichi un algoritmo di on-line connectivity con weighted quick-union, riportando a ogni passo il contenuto dei vettori id e sz. I vertici siano denominati con interi compresi tra 0 e 10.

### Soluzione

La tabella seguente riporta il vettore id a ogni passo:

	0	1	2	3	4	5	6	7	8	9	10
	0	1	2	3	4	5	6	7	8	9	10
3-9	0	1	2	9	4	5	6	7	8	9	10
4-3	0	1	2	9	9	5	6	7	8	9	10
2-5	0	1	5	9	9	5	6	7	8	9	10
6-2	0	1	5	9	9	5	5	7	8	9	10
3-8	0	1	5	9	9	5	5	7	9	9	10
5-6	0	1	5	9	9	5	5	7	9	9	10
0-10	10	1	5	9	9	5	5	7	9	9	10
3-5	10	1	5	9	9	9	5	7	9	9	10
8-9	10	1	5	9	9	9	5	7	9	9	10
10-3	10	1	5	9	9	9	5	7	9	9	9

La tabella seguente riporta il vettore sz a ogni passo:

	0	1	2	3	4	5	6	7	8	9	10
	1	1	1	1	1	1	1	1	1	1	1
3-9	1	1	1	1	1	1	1	1	1	2	1
4-3	1	1	1	1	1	1	1	1	1	3	1
2-5	1	1	1	1	1	2	1	1	1	3	1
6-2	1	1	1	1	1	3	1	1	1	3	1
3-8	1	1	1	1	1	3	1	1	1	4	1
5-6	1	1	1	1	1	3	1	1	1	4	1
0-10	1	1	1	1	1	3	1	1	1	4	2
3-5	1	1	1	1	1	3	1	1	1	7	2
8-9	1	1	1	1	1	3	1	1	1	7	2
10-3	1	1	1	1	1	3	1	1	1	9	2

### **Esercizio risolto: quick-find**

Sia data la seguente sequenza di coppie:

1-2 3-5 0-1 7-4 5-1 9-0 6-5

dove la relazione  $i-j$  indica che il vertice  $i$  è adiacente al vertice  $j$ . Si applichi un algoritmo di on-line connectivity con quick-find, riportando il contenuto del vettore al passo finale. I vertici siano denominati con interi compresi tra 0 e 9.

#### Soluzione

Il vettore riporta la configurazione finale di id:

0	1	2	3	4	5	6	7	8	9
2	2	2	2	4	2	2	4	8	2

#### **Esercizio risolto: quick-union**

Sia data la seguente sequenza di coppie:

1-2 3-5 0-1 7-4 5-1 9-0 6-5

dove la relazione  $i-j$  indica che il vertice  $i$  è adiacente al vertice  $j$ . Si applichi un algoritmo di on-line connectivity con quick-union, riportando il contenuto del vettore al passo finale. I vertici siano denominati con interi compresi tra 0 e 9.

#### Soluzione

Il vettore riporta la configurazione finale di id:

0	1	2	3	4	5	6	7	8	9
2	2	2	5	4	2	2	4	8	2

#### **Esercizio in modalità piattaforma Exam: quick-find**

*Enunciato:*

Sia data la seguente sequenza di coppie, dove la relazione  $i-j$  indica che il vertice  $i$  è adiacente al vertice  $j$ :

2-6, 5-2, 0-7, 4-3, 3-9

Si applichi un algoritmo di on-line connectivity con quickfind. I nodi sono denominati con interi tra 0 e 9.

*Domanda e formato della risposta:*

riportare il vettore id come sequenza di interi

- dopo il passo 1 (i passi cominciano da 1)
- dopo il passo 2
- dopo il passo 3

*Risposta corretta:*

- dopo il passo 1: id 0 1 6 3 4 5 6 7 8 9
- dopo il passo 2: id 0 1 6 3 4 6 6 7 8 9
- dopo il passo 3: id 7 1 6 3 4 6 6 7 8 9

#### **Esercizio in modalità piattaforma Exam: weighted quick-union**

*Enunciato:*

Sia data la seguente sequenza di coppie, dove la relazione  $i-j$  indica che il vertice  $i$  è adiacente al vertice  $j$ :

0-4, 3-0, 1-5, 2-5, 1-8, 4-7, 6-8, 2-4

Si applichi un algoritmo di on-line connectivity con weighted quickunion. I nodi sono denominati con interi tra 0 e 8.

*Domanda e formato della risposta:*

riportare i vettori id e sz come sequenza di interi dopo il passo 6 (i passi cominciano da 1).

*Risposta corretta:*

id 4 5 5 4 4 5 6 4 5

sz 1 1 1 1 4 4 1 1 1

## Algoritmi di Ordinamento iterativi quadratici

### Insertion sort

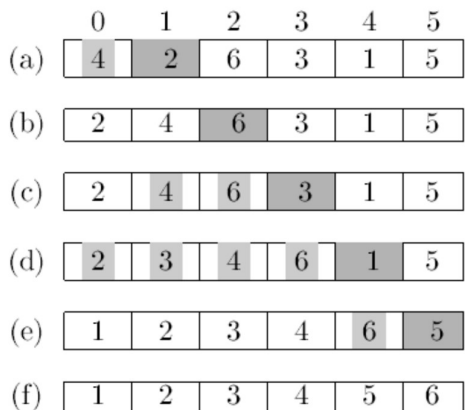
#### Esercizio svolto

Si ordini in maniera ascendente la seguente sequenza di interi mediante insertion sort, indicando i passaggi:

4 2 6 3 1 5

#### Soluzione

La figura seguente fornisce una rappresentazione grafica dei passi dell'algoritmo.



Ogni vettore rappresentato è quello ottenuto mediante una iterazione del ciclo esterno. L'elemento nella cella evidenziata interamente in grigio scuro rappresenta l'elemento  $x$ , ovvero quello selezionato (ed estratto). Gli elementi nelle celle evidenziate parzialmente in grigio chiaro sono quelli spostati verso destra dall'operazione di scalamento (shift) eseguita dal ciclo interno. Al termine dello scalamento il valore estratto  $x$  viene inserito nella posizione rimasta vuota, ovvero quella corrispondente all'elemento grigio chiaro più a sinistra. Ad esempio durante la prima iterazione (riga (a)), si memorizza nella variabile  $x$  il valore 2, si effettua uno scalamento a destra del valore 4 e si inserisce  $x$  in posizione 0. Durante la seconda iterazione (riga (b))  $x$  assume il valore 6, che però viene immediatamente re-inserito in posizione 2 senza effettuare alcun scalamento. Durante la terza iterazione (riga (c)) il valore 3 viene copiato in  $x$ , il valore 6 e poi il valore 4 (in quest'ordine) vengono scalati di una posizione verso destra e il valore 3 viene re-inserito in posizione 1. Si procede in questo modo copiando in  $x$  i valori 1 e 5 e infine si ottiene la configurazione finale del vettore al termine dell'algoritmo nella riga (f).

#### Esercizio risolto

Si ordini in maniera discendente la seguente sequenza di interi mediante insertion sort, indicando i passaggi rilevanti:

-3 2 1 10 15 5 -5 35 7 19

#### Soluzione

Essendo richiesto un ordinamento discendente, si osservi che la condizione di scalamento è  $(j \geq 1 \ \&\& \ x > A[j])$ . La seguente tabella fornisce una rappresentazione grafica dei passi dell'algoritmo. Ogni riga della tabella corrisponde a una iterazione del ciclo esterno e l'elemento evidenziato quello inserito nella corretta posizione del sotto-vettore di sinistra.

0	1	2	3	4	5	6	7	8	9
-3	2	1	10	15	5	-5	35	7	19
2	-3	1	10	15	5	-5	35	7	19
2	1	-3	10	15	5	-5	35	7	19
10	2	1	-3	15	5	-5	35	7	19
15	10	2	1	-3	5	-5	35	7	19
15	10	5	2	1	-3	-5	35	7	19
15	10	5	2	1	-3	-5	35	7	19
35	15	10	5	2	1	-3	-5	7	19
35	15	10	7	5	2	1	-3	-5	19
35	19	15	10	7	5	2	1	-3	-5

### Esercizio in modalità piattaforma Exam

*Enunciato:*

Si ordini in maniera ascendente la seguente sequenza di interi mediante insertion sort:

5 2<sub>1</sub> 1 3 2<sub>2</sub> 6

Nel caso di chiavi duplicate, il pedice indica la posizione nella sequenza di ingresso.

*Domanda e formato della risposta:*

riportare il vettore come sequenza di interi

- dopo l'inserzione di 2<sub>2</sub>.

*Risposta corretta:*

1 2<sub>1</sub> 2<sub>2</sub> 3 4 6

### Exchange (Bubble) sort

#### Esercizio svolto

Si ordini in maniera ascendente la seguente sequenza di interi mediante bubble sort, indicando i passaggi:

4 2 6 3 1 5

#### Soluzione

La figura seguente riporta una rappresentazione grafica del procedimento di ordinamento per scambio. Ogni riga della tabella corrisponde a una iterazione del ciclo esterno.

	0	1	2	3	4	5
(a)	4	2	6	3	1	5
(b)	2	4	3	1	5	6
(c)	2	3	1	4	5	6
(d)	2	1	3	4	5	6
(e)	1	2	3	4	5	6
(f)	1	2	3	4	5	6

Le celle evidenziate interamente in grigio scuro rappresentano le celle stabili, ovvero quelle in cui l'elemento ha raggiunto la sua posizione finale. Tali elementi costituiscono il sotto-vettore ordinato (inizialmente vuoto). Gli elementi nelle celle evidenziate parzialmente in grigio chiaro sono quelli tra i quali si effettua uno scambio, procedendo da sinistra verso destra. Tali elementi si trovano nel sotto-vettore ancora da ordinare (che inizialmente coincide con l'intero vettore). Per esempio, durante la prima iterazione (riga (a)), si scambiano tra di loro gli elementi (4; 2), (6; 3), (6; 1) e (6; 5) spostando l'elemento 6 in posizione 5. Durante la seconda iterazione (riga (b)) si scambiano gli elementi (4; 3) e (4; 1) e si conferma come posizione finale del valore 5 l'indice 4. La riga (f) mostra la configurazione finale.

### Esercizio risolto

Sia data la sequenza di interi, supposta memorizzata in un vettore:

-1 0 -12 9 -9 16 8 3 1 0

la si ordini in ordine decrescente utilizzando l'algoritmo di exchange (bubble) sort. Si riportino graficamente i passi rilevanti dell'algoritmo e il risultato finale.

### Soluzione

Essendo richiesto un ordinamento discendente, la condizione di scalamento è  $A[j] < A[j+1]$ . La tabella fornisce una rappresentazione grafica dei passi dell'algoritmo. Ogni riga della tabella corrisponde a una iterazione del ciclo esterno che si occupa di inserire in posizione corretta (evidenziata con colore scuro) un nuovo elemento.

0	1	2	3	4	5	6	7	8	9
-1	0	-12	9	-9	16	8	3	1	0
0	-1	9	-9	16	8	3	1	0	-12
0	9	-1	16	8	3	1	0	-9	-12
9	0	16	8	3	1	0	-1	-9	-12
9	16	8	3	1	0	0	-1	-9	-12
16	9	8	3	1	0	0	-1	-9	-12
16	9	8	3	1	0	0	-1	-9	-12
16	9	8	3	1	0	0	-1	-9	-12
16	9	8	3	1	0	0	-1	-9	-12
16	9	8	3	1	0	0	-1	-9	-12
16	9	8	3	1	0	0	-1	-9	-12

### Esercizio in modalità piattaforma Exam

*Enunciato:*

Si ordini in maniera ascendente la seguente sequenza di interi mediante optimized bubble sort:

5 3 2 6 4

*Domanda e formato della risposta:*

Dopo quanti passi del ciclo esterno termina l'algoritmo?

*Risposta corretta:*

3

### Selection sort

#### Esercizio svolto

Si ordini in maniera ascendente la seguente sequenza di interi mediante selection sort, indicando i passaggi:

4 2 6 3 1 5

#### Soluzione

La seguente figura fornisce una rappresentazione grafica dei passi dell'algoritmo.

	0	1	2	3	4	5
(a)	4	2	6	3	1	5
(b)	1	2	6	3	4	5
(c)	1	2	6	3	4	5
(d)	1	2	3	6	4	5
(e)	1	2	3	4	6	5
(f)	1	2	3	4	5	6

Ogni riga della tabella corrisponde a una iterazione del ciclo esterno. Le celle evidenziate interamente in grigio scuro rappresentano le celle stabili, ovvero quelle in cui l'elemento ha raggiunto la sua posizione finale. Tali elementi costituiscono il sottovettore ordinato (inizialmente vuoto). Gli elementi nelle celle evidenziate parzialmente in grigio chiaro sono quelli tra i quali si effettua uno scambio, ovvero l'elemento  $A[i]$  e l'elemento di valore minimo tra quelli del sotto-vettore ancora da ordinare. Tale valore viene selezionato dal ciclo interno e scambiato con l'elemento  $A[i]$ . Ad esempio, durante la prima iterazione il valore minimo (1) si trova in posizione 4. Tale elemento viene scambiato con l'elemento in posizione 0 (4) generando la configurazione della riga (b), in cui l'elemento in posizione 1 è nella sua posizione definitiva. Analogamente alla riga (b) il valore minimo del vettore è 2 che quindi viene scambiato con "se stesso". La riga (f) illustra la configurazione finale del vettore.

### Esercizio risolto

Sia data la sequenza di interi, supposta memorizzata in un vettore:

-3 -2 2 9 -9 6 8 13 11 10

la si ordini in ordine decrescente utilizzando l'algoritmo di selection sort. Si riportino graficamente i passi rilevanti dell'algoritmo e il risultato finale.

### Soluzione

La tabella successiva riporta i passaggi e la configurazione finale. Essendo richiesto un ordinamento decrescente, si osservi che il valore ricercato per lo scambio è il massimo del sotto-vettore destro. Ogni riga della tabella corrisponde a una iterazione del ciclo esterno. In tale iterazione, l'elemento massimo del sotto-vettore destro (quello di colore bianco, ancora da ordinare) viene selezionato nell'iterazione del ciclo interno e viene inserito come ultimo elemento del vettore già ordinato (di colore scuro).



0	1	2	3	4	5	6	7	8	9
-3	-2	2	9	-9	6	8	13	11	10
13	-2	2	9	-9	6	8	-3	11	10
13	11	2	9	-9	6	8	-3	-2	10
13	11	10	9	-9	6	8	-3	-2	2
13	11	10	9	-9	6	8	-3	-2	2
13	11	10	9	8	6	-9	-3	-2	2
13	11	10	9	8	6	-9	-3	-2	2
13	11	10	9	8	6	2	-3	-2	-9
13	11	10	9	8	6	2	-2	-3	-9
13	11	10	9	8	6	2	-2	-3	-9

### Esercizio in modalità piattaforma Exam

*Enunciato:*

Si ordini in maniera ascendente la seguente sequenza di interi mediante selection sort:

15 13 2 7 4 18 5 1 90 0 27

*Domanda e formato della risposta:*

Quando l'indice  $i$  del ciclo esterno vale 5, quanto vale il minimo e in quale posizione nel vettore si trova?

*Risposta corretta:*

Il minimo vale 7 e si trova all'indice 6.

### Shell sort

#### Esercizio svolto

Si ordini in maniera ascendente il seguente vettore di interi mediante Shell sort con la sequenza di Knuth:

7 6 8 9 8 6 2 1 8 7 0 4 5 3 0 1 0 0

Si indichino i passaggi principali.

#### Soluzione

Data la dimensione del vettore, l'ultimo valore della sequenza di Knuth da considerare è  $h = 13$ . Applicando l'insertion sort ai sotto-vettori i cui elementi successivi distano tra di loro 13 si ottengono sotto-sequenze 13-ordinate:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
3	0	1	0	0	6	2	1	8	7	0	4	5	7	6	8	9	8

Proseguendo con  $h = 4$ , applicando l'insertion sort ai sotto-vettori i cui elementi successivi distano tra di loro 4 si ottengono sotto-sequenze 4-ordinate:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	0	0	0	3	6	1	1	5	7	2	4	8	7	6	8	9	8

Infine con  $h = 1$ , applicando l'insertion sort ai sotto-vettori i cui elementi successivi distano tra di loro di una posizione, si ottiene il vettore ordinato:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	0	0	0	1	1	2	3	4	5	6	6	7	7	8	8	8	9

### Esercizio risolto

Si ordini in maniera ascendente il seguente vettore di interi mediante Shell sort con la sequenza di Knuth:

17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

Si indichino i passaggi principali.

### Soluzione

Data la dimensione del vettore, l'ultimo valore della sequenza di Knuth da considerare è  $h = 13$ .

sotto-sequenze 13-ordinate:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
4	3	2	1	0	12	11	10	9	8	7	6	5	17	16	15	14	13

sotto-sequenze 4-ordinate:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	3	2	1	4	8	7	6	5	12	11	10	9	13	16	15	14	17

vettore ordinato:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

### Esercizio in modalità piattaforma Exam

*Enunciato:*

Si ordini in maniera ascendente la seguente sequenza di interi mediante Shell sort con la sequenza di Knuth:

15 13 2 7 4 18 5 1 90 0 27 25 23 12 17 44

*Domanda e formato della risposta:*

- Data la dimensione del vettore da ordinare, da quale valore di  $h$  si parte?  
 $h =$
- Riportare il vettore come sequenza di interi quando è 4-ordinato.

*Risposta corretta:*

$h = 13$

vettore 4-ordinato: 4 0 2 1 12 13 5 7 23 15 17 25 90 18 27 44

## Algoritmi di Ordinamento iterativi lineari

### Counting sort

#### Esercizio svolto

Si ordini in maniera ascendente mediante counting sort il seguente vettore A di interi:

1 4 5 1 0 4 2 4

Si indichino le strutture dati usate nei passi intermedi.

### Soluzione

Nell'esercizio si ha  $n = 8$  e  $k = 6$ . La tabella successiva riporta in:

- il vettore A di partenza
- il vettore C dopo l'inizializzazione ( $C^1$ )
- il vettore C delle occorrenze semplici ( $C^2$ )

- d) il vettore C dopo il calcolo delle occorrenze multiple ( $C^3$ )  
 e) il vettore risultato B all'inizio.

		0	1	2	3	4	5	6	7
(a)	A	1	4	5	1	0	4	2	4

		0	1	2	3	4	5
(b)	$C^1$	0	0	0	0	0	0

		0	1	2	3	4	5
(c)	$C^2$	1	2	1	0	3	1

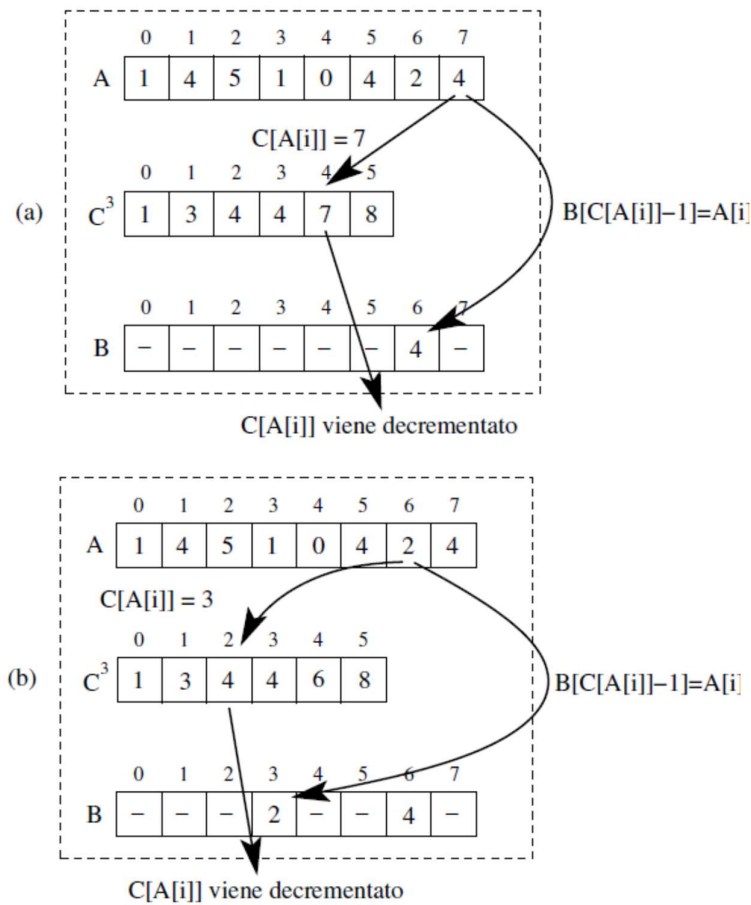
  

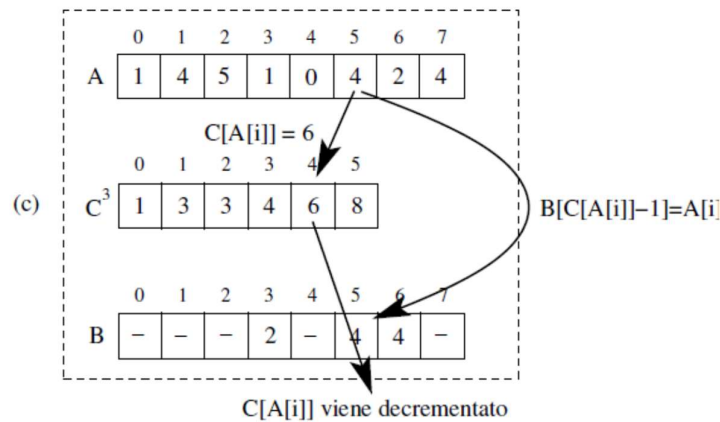
		0	1	2	3	4	5
(d)	$C^3$	1	3	4	4	7	8

		0	1	2	3	4	5	6	7
(e)	B	-	-	-	-	-	-	-	-

I singoli elementi del vettore A sono quindi analizzati in successione, a partire dall'estremo destro procedendo verso quello sinistro. La figura seguente riporta le prime tre fasi del procedimento.





Il primo valore esaminato è 4, usando il quale come indice si ricava dal vettore delle occorrenze multiple che il numero di elementi di A di valore minore o uguale a esso è pari a 7. Questo vuol dire che 4 può essere memorizzato direttamente nel settimo elemento (quello di indice 6) del vettore ordinato, proprio perché si sa già che tutti gli elementi del vettore ordinato di indice inferiore a 6 alla fine conterranno valori minori di (o al più uguali a) 4. Il vettore B assumerà dunque l'aspetto di figura (a). Nel contempo, l'elemento di indice 4 del vettore delle occorrenze multiple viene decrementato, per cui il contenuto di C diventa quello riportato in figura (b). Si esamina poi il penultimo elemento di A, ovvero 2. In questo caso, il numero di elementi di A con valore minore o uguale a 2 è pari a 4, quindi 2 può essere ricopiato nel quarto elemento (all'indice 3) di B; l'elemento di C di indice 2 viene inoltre decrementato. I vettori B e C corrispondenti sono riportati nelle figure (b) e (c), rispettivamente. A questo punto si passa a considerare il terz'ultimo elemento di A, che è di nuovo 4. Questa volta si deduce dal vettore delle occorrenze multiple che il numero di elementi "rimasti" in A di valore minore o uguale a 4 è pari a 6, quindi 4 viene copiato nel sesto elemento di B, e il corrispondente elemento di C viene ancora decrementato: il vettore B è riportato in figura (c). Si procede in questo modo sino a quando tutti gli elementi di A non sono stati considerati. La situazione finale a cui si giunge è:

	0	1	2	3	4	5	6	7
B	0	1	1	2	4	4	4	5

in cui il vettore B corrisponde esattamente al vettore A ordinato. Si omette, in quanto banale, il passo finale in cui si ricopia il vettore B nel vettore A.

### Esercizio risolto

Si ordini in maniera ascendente mediante counting sort il seguente vettore di interi:

5 6 5 3 3 7 4 4 4 5 3 8 8

Si indichino le strutture dati usate nei passi intermedi.

### Soluzione

La tabella successiva riporta nell'ordine: il vettore iniziale A, il vettore C dopo l'inizializzazione (1), il calcolo delle occorrenze semplici (2) e di quelle multiple (3) e il vettore finale B. Sono inoltre riportati i decrementi dei valori degli elementi di C durante le varie fasi di inserzione degli elementi originali nel vettore finale.

	0	1	2	3	4	5	6	7	8	9	10	11	12
A	5	6	5	3	3	7	4	4	4	5	3	8	8

	0	1	2	3	4	5	6	7	8
C <sup>1</sup>	0	0	0	0	0	0	0	0	0

	0	1	2	3	4	5	6	7	8
C <sup>2</sup>	0	0	0	3	3	3	1	1	2

	0	1	2	3	4	5	6	7	8
C <sup>3</sup>	0	0	0	3	6	9	10	11	13

				2	5	8	9	10	12
			1	4	7				11
			0	3	6				

	0	1	2	3	4	5	6	7	8	9	10	11	12
B	3	3	3	4	4	4	5	5	5	6	7	8	8

### Esercizio in modalità piattaforma Exam

*Enunciato:*

Si ordini in maniera ascendente mediante counting sort il seguente vettore di interi:

2<sub>1</sub> 4<sub>1</sub> 3<sub>1</sub> 1 3<sub>2</sub> 5<sub>1</sub> 4<sub>2</sub> 2<sub>2</sub> 4<sub>3</sub> 5<sub>2</sub> 3<sub>3</sub> 9 8

*Domanda e formato della risposta:*

- Dato il vettore da ordinare, quanto vale k?  
k =
- riportare come sequenza di interi il contenuto del vettore C delle occorrenze semplici
- a quale indice nel vettore risultato verrà memorizzata la chiave 4<sub>2</sub>?

*Risposta corretta:*

k = 10

C: 0 1 2 3 3 2 0 0 1 1

all'indice 7

### Radix sort

#### Esercizio risolto

Si ordini in maniera ascendente mediante radix sort il seguente vettore A di numeri di 2 cifre in base 8:

11 42 35 10 70 34 26 54

Si indichino le strutture dati usate nei passi intermedi.

#### Soluzione

Si eseguono 2 passi di counting sort su dati espressi su 1 cifra ottale, partendo dalla colonna delle cifre meno significative.

Passo 1: il vettore D della colonna delle cifre meno significative contiene 1, 2, 5, 0, 0, 4, 6, 4. Trattandosi di base 8 il vettore delle occorrenze semplici/multiple consta di 8 elementi. Il vettore delle occorrenze semplici contiene 2 1 1 0 2 1 1 0, quello delle occorrenze multiple 2 3 4 4 6 7 8 8. Applicando il counting sort A contiene 10 70 11 42 34 54 35 26.

Passo 2: dato il vettore A calcolato al passo precedente, il vettore D della colonna delle cifre più significative contiene 1, 7, 1, 4, 3, 5, 3, 2. Trattandosi di base 8 il vettore delle occorrenze semplici/multiple consta di 8 elementi. Il vettore delle occorrenze semplici contiene 0 2 1 2 1 1 0 1, quello delle occorrenze multiple 0 2 3 5 6 7 7 8. Applicando il counting sort A contiene 10 11 26 34 35 42 54 70.

### Esercizio svolto

Si ordini in maniera ascendente mediante radix sort il seguente vettore A di interi in base 10:

101 942 85 1023 70 4 26 154

### Soluzione

d=4

Passo 1

D = 1, 2, 5, 3, 0, 4, 6, 4

C<sub>occ. sempl.</sub> = 1, 1, 1, 1, 2, 1, 1, 0, 0, 0

C<sub>occ. mult.</sub> = 1, 2, 3, 4, 6, 7, 8, 8, 8, 8

A = 70, 101, 942, 1023, 4, 154, 85, 26

Passo 2

D = 7, 0, 4, 2, 0, 5, 8, 2

C<sub>occ. sempl.</sub> = 2, 0, 2, 0, 1, 1, 0, 1, 1, 0

C<sub>occ. mult.</sub> = 2, 2, 4, 4, 5, 6, 6, 7, 8, 8

A = 101, 4, 1023, 26, 942, 154, 70, 85

Passo 3

D = 1, 0, 0, 0, 9, 1, 0, 0

C<sub>occ. sempl.</sub> = 5, 2, 0, 0, 0, 0, 0, 0, 0, 1

C<sub>occ. mult.</sub> = 5, 7, 7, 7, 7, 7, 7, 7, 7, 8

A = 4, 1023, 26, 70, 85, 101, 154, 942

Passo 4

D = 0, 1, 0, 0, 0, 0, 0, 0

C<sub>occ. sempl.</sub> = 7, 1, 0, 0, 0, 0, 0, 0, 0, 0

C<sub>occ. mult.</sub> = 7, 8, 8, 8, 8, 8, 8, 8, 8, 8

A = 4, 26, 70, 85, 101, 154, 942, 1023

### Esercizio in modalità piattaforma Exam

*Enunciato:*

Si ordini in maniera ascendente mediante radix sort il seguente vettore A di interi su 3 cifre in base 10:

201 742 185 123 970 444 261 184

*Domanda e formato della risposta:*

- passo 1 colonna delle unità: riportare come sequenza di interi il contenuto iniziale del vettore C delle occorrenze semplici e multiple
- passo 2 colonna delle decine: riportare come sequenza di interi il contenuto del vettore A dopo il passo di counting sort.

*Risposta corretta:*

passo 1: semplici C = 1, 2, 1, 1, 2, 1, 0, 0, 0, 0, multiple C = 1, 3, 4, 5, 7, 8, 8, 8, 8, 8

passo 2 A = 201, 123, 742, 444, 261, 970, 184, 185

# Algoritmi di Ordinamento iterativi linearitmici

## Bottom-up merge sort

### Esercizio risolto

Si ordini in maniera ascendente mediante bottom-up merge sort il seguente vettore A di interi:

121 42 58 13 7 41 161 314 15 9 27

### Soluzione

Si osservi che  $n=11$ , quindi non è una potenza di 2. I vettori formati da un singolo elemento sono per definizione ordinati. Ad ogni passo si fondono coppie contigue di vettori ordinati per ottenere un vettore ordinato di dimensione pari alla somma delle dimensioni dei vettori di partenza.

Passo 1: 121 42 58 13 7 41 161 314 15 9 27

Passo 2: (42, 121) (13, 58) (7, 41) (161,314) (9, 15) (27)

Passo 3: (13, 42, 58, 121) (7, 41, 161,314) (9, 15, 27)

Passo 4: (7, 13, 41, 42, 58, 121, 161, 314) (9, 15, 27)

Passo 5: (7, 9, 13, 15, 27, 41, 42, 58, 121, 161, 314)

### Esercizio svolto

Supponendo di aver ottenuto ad un certo passo dell'algoritmo di bottom-up merge sort i 2 seguenti sottovettori ordinati

(2, 6<sub>1</sub>, 19, 35, 67<sub>1</sub>, 89<sub>1</sub>, 110, 230) e (1, 6<sub>2</sub>, 6<sub>3</sub>, 45, 67<sub>2</sub>, 89<sub>2</sub>, 89<sub>3</sub>, 89<sub>4</sub>, 95)

si applichi ad essi la funzione merge.

### Soluzione

(1, 2, 6<sub>1</sub>, 6<sub>2</sub>, 6<sub>3</sub>, 19, 35, 45, 67<sub>1</sub>, 67<sub>2</sub>, 89<sub>1</sub>, 89<sub>2</sub>, 89<sub>3</sub>, 89<sub>4</sub>, 95, 110, 230)

### Esercizio in modalità piattaforma Exam

*Enunciato:*

Si ordini in maniera ascendente mediante bottom-up merge sort il seguente vettore A di interi:

21 742 85 13 70 4 261 14

*Domanda e formato della risposta:*

- riportare come sequenza di interi il contenuto dei sottovettori ordinati di dimensione 2
- riportare come sequenza di interi il contenuto dei sottovettori ordinati di dimensione 4.

*Risposta corretta:*

sottovettori ordinati di dimensione 2: (21, 742) (13, 85) (4, 70), (14, 261)

sottovettori ordinati di dimensione 4: (13, 21, 85, 742) (4, 14, 70, 261)

## Analisi di complessità

Sia data la seguente funzione

```
int trova (int *v, int n, int s) {
    int i, j, somma;
    for (i=0; i<n-4; i++) {
        somma = 0;
        for (j=0; j<4; j++)
            somma += v[i+j];
        if (somma == s)
            return i;
    }
}
```

```

}
return -1;
}

```

- Determinare la complessità asintotica di caso peggiore (limite lasco) della funzione  $T(n) = O(\dots)$ 
  - la funzione presenta 2 cicli annidati. Quello esterno esegue un numero di iterazioni che dipende da  $n$ , quello interno esegue un numero fisso di iterazioni. Quindi  $T(n) = O(n)$ . La complessità è lineare.
- Determinare la complessità asintotica di caso peggiore (limite stretto) della funzione  $T(n) = \Theta(\dots)$ 
  - non si può definire un limite asintotico stretto a causa dell'uscita non strutturata `return 1` quando è soddisfatta la condizione `somma == s`. Il ciclo esterno viene eseguito al massimo  $n-4$  volte, quindi  $O(n)$ , non esattamente  $n-4$  volte.
- Quale caratteristica ha l'intervallo di dati consecutivi che la funzione cerca nel vettore?
  - Si tratta di un intervallo di 4 interi consecutivi la cui somma vale  $s$ .
- Nel caso in cui il vettore contenesse più intervalli con tale caratteristica, si ritorna il primo, l'ultimo o uno a caso?
  - La funzione termina con una `return` non strutturata non appena trova l'intervallo con le caratteristiche specificate, quindi il primo.