

Q

12/11/

1° Vermisue

Supponiamo l'errore in cui misurare un
elemento in posizione $0 \leq p \leq a$. Length-1
me

Q.

10	11	12	13	14	15
----	----	----	----	----	----

 \uparrow a.length-1

Supponiamo di voler misurare 1 in $a[1]$.
La strategia è:

- creare un nuovo array b tale che $b.length = a.length + 1$.
- copiare ogni elemento $a[i]$ in $b[i]$.
- spostare tutti gli elementi di b , a destra di p di una posizione verso destra.
- Inserire 1 in $a[p]$.

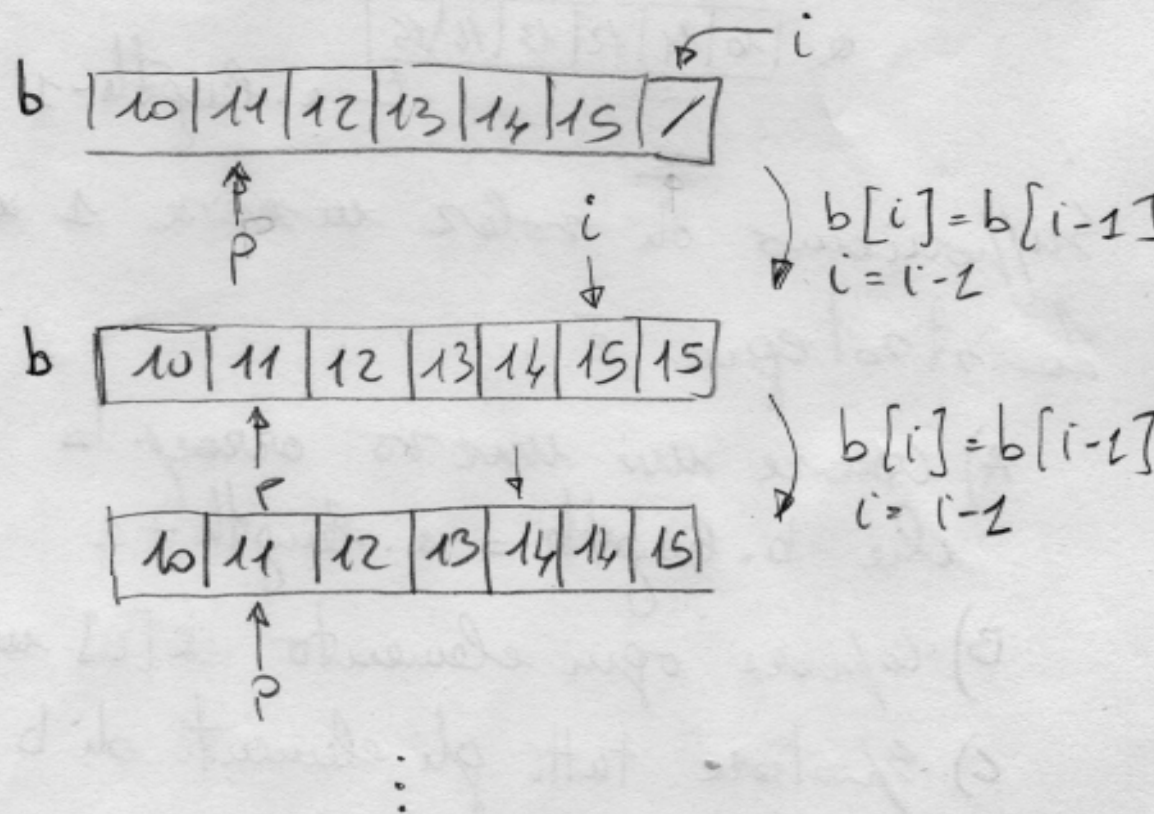
La situazione finale sarà

Q

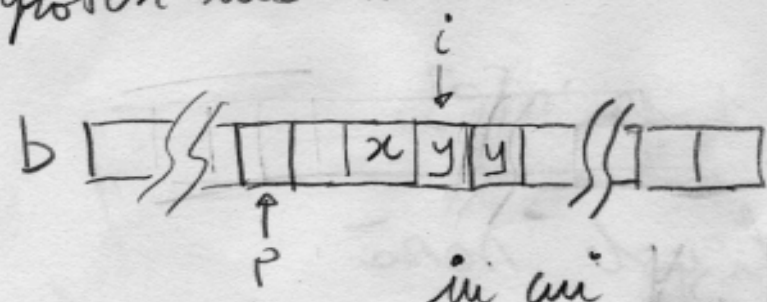
10	11	12	13	14	15
----	----	----	----	----	----

↳ 10 | 1 | 11 | 12 | 13 | 14 | 15

Il punto c è quello interessante 17/11
 del punto di vista opportuno e
 del punto di vista della struttura
 dell'invariante di ciclo. Occorre
 osservare il seguente processo su b:



L'ipotesi induttiva sembra essere:



in cui
 $\forall k. se\ b.length - 1 \geq k > i$
 allora $b[k] = e[k-1]$

applicando ad esse la copia

(3)
17/11

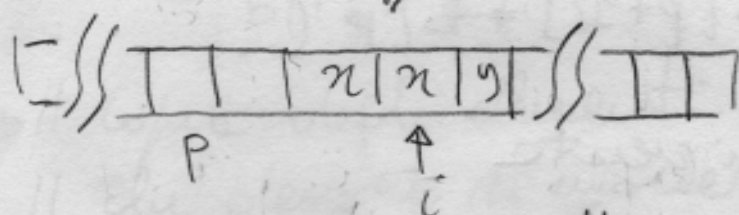
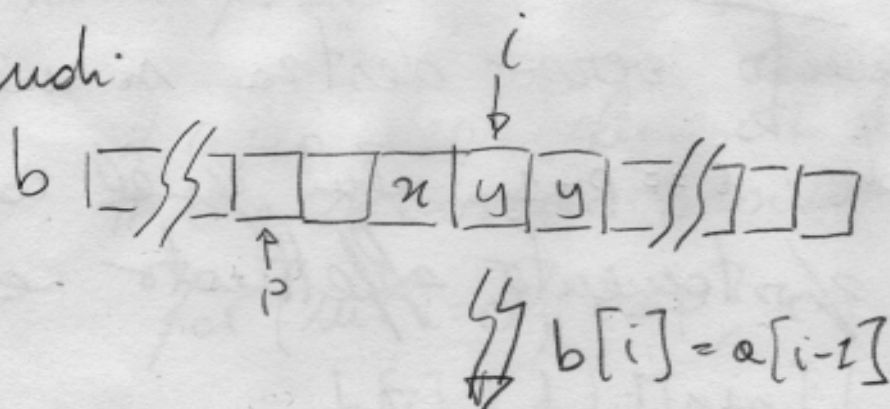
$$b[i] = b[i-1]$$

è come se stessimo in realtà scrivendo

$$b[i] = a[i-1]$$

in cui $a[i-1] = x$

Quindi:



$$\begin{cases} \text{in cui } b[b.length-1] = a[a.length-1] \\ b[b.length-2] = a[a.length-2] \\ \vdots \end{cases}$$

$$b[i+1] = a[i]$$

mi ottengo appunto

$$b[i] = a[i-1]$$

predicato che si può scrivere come

$$\forall k. \text{ se } b.length-1 \geq k \geq i$$

$$\text{allora } b[k] = a[k-1]$$

applicando l'aggiornamento
 $i = i - 1$

il predicato ridiventa

(4)

17/11/14

$\forall k. \text{if } b.\text{length} \geq k > i$

allora $b[k] = a[k-1]$

confermando che è l'invariante

lo spostamento verso destra si interrompe quando $i == p+1$. In quel caso l'ultimo spostamento effettuato è

$b[p+1] = b[p]$.

Il codice diventa

```
for (int i = b.length-1; i > p; i--)  
    b[i] = b[i-1];
```

Esso può essere scritto anche

$b[p] = e$

2° Versione

(5)

L'inconveniente della prima versione
suggerisce di evitare la copia completa
degli elementi di a in b , per poi spostar
ne una parte.

```
// copia degli elementi di  $a$   
// che stanno a sinistra di  $p$   
for (int  $i=0$ ;  $i < p$ ;  $i++$ )  
     $b[i] = a[i]$ 
```

```
// copia degli elementi di  $a$   
// che devono rimanere a  
// destra di  $p$ .
```

```
for (int  $i=p$ ;  $i < a.length$ ;  $i++$ )  
     $b[i+1] = a[i]$ 
```

```
// inserimento di  $e$  in  $p$ .  
 $b[p] = e$ .
```