

Osservazioni sull'esercizio 1 del 15/10/2014

La soluzione dell'esercizio potrebbe venire organizzata in 3 parti:

- 1) riempimento di A;
- 2) considerazione dei primi 3 elementi di A e determinazione tra essi del primo, secondo e terzo valore;
- 3) ciclo che considera i successivi elementi di A uno alla volta e per ciascuno di essi modifica eventualmente primo, secondo e terzo in modo che continuino a svolgere il loro ruolo anche considerando il nuovo elemento di A.

Il punto (2) lo si può fare generalizzando la soluzione vista in classe per l'esercizio che chiedeva di determinare il secondo valore. Anche il punto (3) dovrebbe essere una facile generalizzazione dell'analogo ciclo di quella soluzione. Per (1) dobbiamo fare attenzione a non leggere inutilmente un valore da cin che potrebbe anche non esserci secondo la PRE.

Ripetiamo la PRE_1:

PRE_1=(cin contiene (almeno) 20 interi) || (cin contiene 0)

quindi è possibile che cin contenga **esattamente** 20 valori diversi da 0. Per cui, se in questo caso il nostro programma provasse a leggerne 21, rimarrebbe appeso in eterno aspettando un input che non arriverebbe mai. Il seguente frammento di programma (peraltro ragionevole) farebbe questo errore:

```
int N=0,x, A[20];
cin>>x;
while(x!=0 && N<20)
/*R=(letti N+1 valori)&&(i primi N sono diversi da 0 e sono in A[0..N-1]) &&
(x è l'N+1-esimo) && (0<=N<=20). */
{
  A[N]=x;
  N=N+1;
  cin>>x;
}
```

Si osservi che l'invariante dice chiaramente che N può arrivare a 20 e che vengono letti N+1 valori, quindi fino a 21 valori. Insomma se lo guardiamo bene ci indica l'errore.