

II Compitino di Programmazione

10 marzo 2010

Teoria

(i) Si consideri il frammento di programma: `int y=1, *p=&y; char c=static_cast<char>(p);` spiegate brevemente cosa fa e se è un'operazione corretta o no.

(ii) Si consideri il seguente programma e si spieghi se è corretto o no motivando la risposta:

```
int*& f(int** z) {int a=1; *z=&a; return *z;}
```

```
main(){int a=2, *p=&a, **q=&p; *f(q)=*p+a;}
```

Programmazione

1) data una lista $L = l_1 \rightarrow l_2 \rightarrow l_3 \rightarrow l_4 \rightarrow \dots$, per un $k > 0$ dato, vogliamo prendere i suoi nodi $l_k, l_{(k*2)}, l_{(k*3)}$, eccetera e formare con essi una nuova lista $Q = l_k \rightarrow l_{(k*2)} \rightarrow l_{(k*3)} \rightarrow \dots$, mentre L conterrà i nodi della lista originale che non sono in Q .

(i) Scrivere PRE e POST per una funzione ricorsiva con prototipo `nodo* slice(nodo*& L, int k, int n)` che realizzi l'operazione descritta; il parametro formale k è l'intero, usato nella descrizione precedente, che determina quali nodi "togliere" da L , mentre n serve a scandire il numero dei nodi per sapere "quando è ora" di togliere un nodo da L .

(ii) Scrivere il codice di una funzione ricorsiva che realizzi l'operazione descritta;

(iii) dimostrare (usando l'induzione) che la funzione (ii) è corretta rispetto alla PRE e POST di (i).

2) OPZIONALE: se voleste scomporre una lista L in k nuove liste la prima che contiene i nodi $l_1 \rightarrow l_{(k+1)} \rightarrow l_{(k*2+1)} \rightarrow \dots$, la seconda che contiene i nodi $l_2 \rightarrow l_{(k+2)} \rightarrow l_{(k*2+2)} \rightarrow \dots$,e la k -esima che contiene i nodi $l_k \rightarrow l_{(k*2)} \rightarrow l_{(k*3)} \rightarrow \dots$, come affrontereste il problema? Scrivere una PRE ed una POST ed il prototipo della funzione. Questa funzione deve avere un parametro formale $int k$ che indica il numero di liste in cui scomporre quella originale.