

## Compitino 5 8/6/2016

### Teoria (6 punti)

1) spiegare se il seguente programma è corretto o meno. In ogni caso motivare la risposta. Nel caso pensiate che sia corretto, dite cosa stampa:

```
int* F(int** a){int*b=*a; b=b+1;*b=*b+1;(*a)=(*a)+1; return *a;}
main(){int B[]={0,1,2,3,4},*x=B+2, *y; y=F(&x); cout<<*x<<' '<<*y<<' '<<*(y-1)<<endl; }
```

2) Scrivere adeguate pre e post-condizioni per la seguente funzione ricorsiva dove Lista(l) è una lista concatenata:

```
int G(nodo*l)
{
    if(! l->next->next) return 2;
    else return 1+ G(l->next);
}
```

### Programmazione (18 punti)

Si tratta di scrivere una funzione ricorsiva che, dato un albero binario ed un array C, cerca la foglia a profondità minima (in caso di più foglie a profondità uguale tra loro e minore delle altre foglie, va considerata la foglia più a sinistra) e restituisce un puntatore a questa foglia e in C inserisce il cammino fino a questa foglia. Come sempre il cammino in C è costituito da una sequenza di 0/1 terminata da -1. Nella seconda pagina trovate alcuni esempi. L'array C ha 50 elementi (vedi main) e si deve assumere che la profondità degli alberi che vengono considerati dal programma non superi 49. Quindi C, se verrà usato correttamente, sarà sempre sufficiente a contenere qualsiasi cammino di qualsiasi albero venga considerato nel programma. La funzione CF da definire deve essere **ricorsiva** e deve soddisfare le seguenti pre e postcondizioni:

PRE=(Tree(r) è corretto **e non vuoto** ed ha profondità al più 49, C ha più elementi della profondità di Tree(r))

nodo\* CF(nodo\* r, int\*C)

POST=(CF restituisce la foglia f di Tree(r) a profondità minima (e in caso di parità quella più a sinistra) e in C restituisce il cammino in Tree(r) da r alla foglia f)

**Osserva:** CF restituisce sempre un nodo e relativo cammino in C visto che la PRE richiede che Tree(r) non sia vuoto e quindi Tree(r) ha sempre almeno una foglia.

### Consiglio:

i) conviene introdurre alcune semplici funzioni ausiliarie.

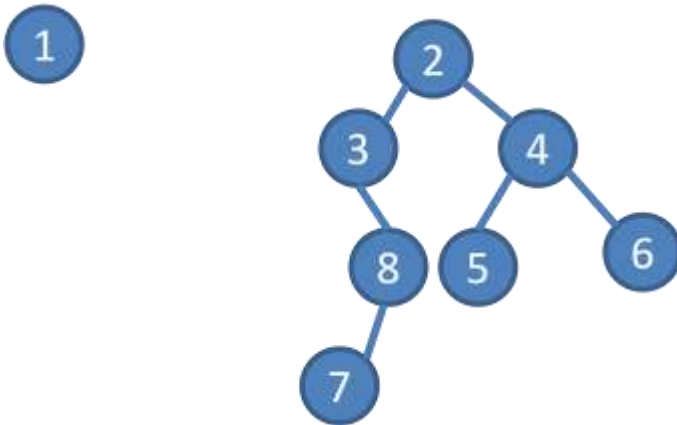
ii) attenzione all'uso dell'array C. Se ce ne fosse uno solo rischiereste interferenze in C tra le diverse invocazioni ricorsive.

### Correttezza (8 punti)

(i) Dimostrare la correttezza di CF.

(ii) Definire pre e postcondizione per ogni funzione ausiliaria.

### Esempi:



L'albero a sinistra consiste della sola radice che è anche l'unica foglia, quindi CF dovrebbe restituire il puntatore alla radice e in C dovrebbe inserire  $C = [-1]$ . Il resto di C non importa. In questo caso il programma stamperà: foglia con info= 1  $C = [-1]$

Nell'albero a destra ci sono 2 foglie a profondità minima: quella con info=5 e quella con info=6. Visto che in questo caso l'esercizio richiede di restituire la foglia più a sinistra, CF dovrà restituire il puntatore al nodo con info=5 e in C dovrà inserire  $C = [1,0,-1]$ . Di conseguenza il programma stamperà: foglia con info= 5  $C = [1,0,-1]$ .