

Tempo a disposizione: ore 2.

1. Si dica se la seguente grammatica è ambigua oppure no, motivando quanto asserito:

$$(\{S\}, \{S, C, B\}, \{\text{if, then, else, skip, true, false, endif}\}, R)$$

dove R è il seguente insieme di regole:

$$\begin{aligned} S &\rightarrow C; \\ C &\rightarrow \text{skip} \mid \text{if B then C else C endif} \mid \text{if B then C endif} \\ B &\rightarrow \text{true} \mid \text{false} \end{aligned}$$

2. Alcuni linguaggi di programmazione, usano per il comando `for` una sintassi diversa a seconda del fatto che l'incremento sia positivo o negativo. Come mai? E come mai altri linguaggi usano la stessa sintassi per entrambi i casi?
3. Si consideri un linguaggio con scope statico, implementato mediante catena statica, nel quale tutti gli identificatori sono noti staticamente. Si assuma inoltre che ogni nome usato è dichiarato in un solo blocco. Si prendano in considerazione le operazioni di "accesso ad una variabile non locale x " e "accesso ad una variabile locale y " (nel contesto di un blocco). Per ognuna delle due operazioni di dica a quale dei seguenti parametri è proporzionale il tempo necessario all'esecuzione dell'operazione, motivando brevemente la risposta.
 - (i) Il numero di variabili presenti nel programma;
 - (ii) Il numero di variabili presenti nei blocchi compresi tra quello di dichiarazione della variabile e quello in cui si accede alla variabile stessa;
 - (iii) Il numero di record di attivazione presenti sulla pila e compresi tra quello contenente la dichiarazione della variabile e quello in cui vi si accede;
 - (iv) il numero di blocchi che contengono il blocco in cui si accede alla variabile e che sono contenuti in quello nel quale la variabile è dichiarata;
 - (v) Il tempo è costante, e quindi indipendente da questi parametri;
 - (vi) nessuna delle risposte precedenti.
4. Si dica cosa stampa il seguente frammento in uno pseudolinguaggio con passaggio per riferimento:

```
int X[10];
int i = 1;
X[0] = 10;
X[1] = 10;
X[2] = 10;
void foo (reference int Y,J){
    X[J] = J-1;
    write(Y);
    J++;
    X[J]=J;
    write(Y);
}
foo(X[i], i);
write(X[i]);
```