

**Linguaggi Formali e Compilatori**  
**Proff. Breveglieri, Crespi Reghizzi, Morzenti**  
**Prova scritta<sup>1</sup>: Domanda relativa alle esercitazioni**  
**05/03/2008** **Acse**

COGNOME: .....  
NOME: ..... Matricola: .....  
Iscritto a: ☐ Laurea Specialistica ☐ V. O. ☐ Laurea Triennale ☐ Altro:.....  
Sezione: ☐ Prof. Breveglieri ☐ Prof. Crespi ☐ Prof. Morzenti

Per la risoluzione della domanda relativa alle esercitazioni si deve utilizzare l'implementazione del compilatore **Acse** che viene fornita insieme al compito.

Si richiede di modificare la specifica dell'analizzatore lessicale da fornire a **flex**, quella dell'analizzatore sintattico da fornire a **bison** ed i file sorgenti per cui si ritengono necessarie delle modifiche in modo da estendere il compilatore **Acse** con la possibilità di gestire un costrutto *switch*, del quale viene qui fornito un esempio.

```
int a;
...
switch (a) {
    case 0:
        ...
        break;
    case 1:
        ...      /* Blocco di codice privo di break      */
    case 2:
        ...
        break;
    default:
        ...
}
```

Il costrutto **switch** é composto da un numero arbitrario di blocchi *case* ed da un blocco opzionale, *default*. Questo tipo di costrutto **switch** ha la seguente semantica: il valore tra parentesi viene confrontato con i valori riportati dai vari *case*. In caso di test positivo, il codice all'interno del blocco case corrispondente viene eseguito. Un *break* all'interno di un blocco porta l'esecuzione alla prima istruzione dopo il costrutto *switch*. Se un blocco di codice non dovesse contenere l'istruzione *break*,

---

<sup>1</sup>Tempo 45'. Libri e appunti personali possono essere consultati.  
È consentito scrivere a matita. Scrivere il proprio nome sugli eventuali fogli aggiuntivi.

allora l'esecuzione deve continuare al successivo blocco *case* (se presente), altrimenti occorre saltare alla fine del costrutto *switch*. Se nessun blocco *case* viene eseguito, il controllo passa o al blocco di *default* (se presente) o alla prima istruzione dopo il costrutto *switch*. Deve essere dichiarato almeno un blocco *case*.

Ipotizzando di aver assegnato il valore 0 alla variabile **a** dell'esempio riportato, verrà eseguito solo il ramo *case 0*. Nel caso in cui sia assegnato il valore 1 alla variabile **a**, allora dovranno essere eseguite le istruzioni associate al secondo *case* e quelle associate al terzo (*case 1*, *case 2*). Nel caso in cui il valore alla variabile **a** sia diversa da 0,1 e 2, allora l'esecuzione salta al solo blocco *default*.

Costrutti *switch* possono essere nidificati.

Le modifiche devono mettere il compilatore **Acse** in condizione di analizzare la correttezza sintattica dei costrutti sopra descritti e di generare una traduzione corretta nel linguaggio assembler della macchina **Mace**.

1. Definire i token (e le relative dichiarazioni in `Acse.lex` e `Acse.y`) necessari per ottenere la funzionalità richiesta. (3 punti)
2. Definire le regole sintattiche necessarie per ottenere la funzionalità richiesta. (8 punti)

3. Definire le azioni semantiche necessarie per ottenere la funzionalità richiesta, senza considerare il blocco *default* e il costrutto *break*. (10 punti)

4. Definire le azioni semantiche necessarie per gestire il blocco opzionale *default*.  
(6 punti)

5. Definire le azioni semantiche necessarie per gestire il costrutto *break*. (6 punti)