

Tempo a disposizione: ore 2.

Svolgere gli esercizi 1-4 e 5-8 su fogli differenti.

1. La seguente grammatica è ambigua? Motivare adeguatamente.

$$\begin{aligned} S &\rightarrow SA \mid A \\ A &\rightarrow aAb \mid \varepsilon \end{aligned}$$

2. Con la notazione \mathcal{C}_{L_1, L_2}^L indichiamo un compilatore da L_1 a L_2 scritto in L . Con $\mathcal{I}_{L_1}^L$ indichiamo un interprete scritto in L per il linguaggio L_1 . Cosa indica, più semplicemente, l'espressione

$$\mathcal{I}_{L_1}^L(\mathcal{C}_{L_1, L_2}^L, \mathcal{C}_{L_1, L_2}^{L_1})$$

3. Si consideri la dichiarazione di array multidimensionale `int A[10][10][10]`. Sappiamo che: un intero è memorizzato su 4 byte; l'array è memorizzato in ordine di riga, con indirizzi di memoria crescenti (cioè se un elemento è all'indirizzo i , il successivo è a $i+4$ ecc.) Qual è l'offset dell'elemento `A[4][2][1]` rispetto all'elemento `A[0][0][0]`? (Si risponda in notazione decimale).
4. Si dica cosa stampa il seguente frammento in un linguaggio con eccezioni e passaggio per valore-risultato:

```
{int y=0;
void f(int x){
    x = x+1;
    throw E;
    x = x+1;
}
try{ f(y); } catch E {y=y+10;};
write(y);
}
```

5. Si dica cosa viene stampato dal seguente frammento di codice in un linguaggio con scoping statico e passaggio dei parametri per valore (con `void f(value int A[10])` si intende la dichiarazione di una funzione con parametro formale per valore di tipo array di 10 interi):

```
{int z = 0;
int V[10];
void f(value int A[10]){
    int i;
    for (i=0; i<10; i=i+1){
        A[i] = z;
        z = z + i;
    }
}
for (j=0; j<10; j=j+1)
    V[j] = 1;
f(V);
write(z, V[9]);
}
```

6. Si consideri il seguente frammento di codice:

```
{int x,v;
 x = 10;
 v = 5;

int fie(int w,z){
    x = (w++) + z;
    write(x)
}
{int x = 1;
*****
}}
```

Si scriva al posto degli asterischi una chiamata a `fie` e si formulino delle opportune ipotesi (p.e., sul passaggio dei parametri, le regole di scoping, ecc.) in modo che venga stampato il valore 21. La chiamata deve usare come parametri attuali solo nomi di variabili (e non espressioni più complesse).

7. Sono date, in Java, le seguenti dichiarazioni di classi:

```
class A{
    int a = 0;
    void f(){g();}
    void g(){a=1;}
}

class B extends A{
    int b = 0;
    void g(){b=2;}
}
```

Si dica cosa stampa, nello scope di queste dichiarazioni, il seguente frammento:

```
B x = new B();
x.f();
System.out.print(x.a);
System.out.print(x.b);
```

8. Si consideri il seguente frammento in uno pseudolinguaggio con modello delle variabili a riferimento e garbage collector mediante contatori dei riferimenti; se `OGG` è un generico oggetto nello heap, indichiamo con `OGG.cont` il suo contatore (nascosto).

```
class C { int n; C next;}
C foo(){
    C p = new C(); // oggetto OGG1
    p.next = new C(); // oggetto OGG2
    C q = new C(); // oggetto OGG3
    q.next = p.next;
    return p.next;
}
C r = foo();
```

Si dica quali sono i valori dei contatori dei riferimenti dei tre oggetti dopo l'esecuzione della linea 6 e della linea 9.