

Linguaggi di Programmazione

Nome e Cognome	
Corso di laurea	
Telefono	
Email	

1. Specificare la grammatica EBNF di un linguaggio imperativo, in cui ogni frase definisce una lista (non vuota) di procedure. Ecco un esempio di definizione di procedura:

```

proc alfa(in a,b: int, out s: string, out r: record(x,y: real, i: int), inout j: int)
  v, w: real;
  m,n: int;
  dispari: set of int;
  t: record(e,f: int, s: set of real);
  s1, s2: set of record (a: int, b: set of record (c: string, d: set of int));
  nome, cognome: string;
begin
  r.x := 2.34;
  r.y := r.x;
  m := 10;
  n := m;
  dispari := [1,3,5,7,9];
  t.e := n;
  t.s := [1.0, 3.14, 12.375, 0.01];
  s1 := [(2, [("sole", [4, 6, 10]), ("luna", []), ("terra", [1,100])]),
        (5, [("giove", [16]), ("martedì", [2,7,9])]),
        (3, [])];
  s2 := s1;
  s1 := [];
  nome := "luigi";
  cognome := "rossi";
end;

```

Ogni procedura ha una lista (non vuota) di parametri. Ogni parametro è qualificato mediante una keyword che ne specifica la modalità di passaggio: **in**, **out**, **inout**. Dopo l'intestazione, è possibile (ma non necessario) definire un insieme di variabili locali. I tipi atomici sono **int**, **real** e **string**. I costruttori di tipo (perfettamente ortogonali tra loro) sono **record** e **set** (insieme). Il corpo della procedura è costituito da una lista non vuota di assegnamenti, in cui la parte sinistra è un identificatore (di variabile o parametro) o un campo (eventualmente annidato) di record. La parte destra dell'assegnamento può essere una costante, un identificatore o un campo (eventualmente annidato) di record. Una costante di tipo **record** è racchiusa tra parentesi tonde. Una costante di tipo **set** è racchiusa tra parentesi quadre.

2. È data la seguente tabella di operatori, per la quale si assume priorità decrescente dall'alto verso il basso,

Operatore	Tipo	Associatività	Ordine valutazione	Corto circuito
=	binario	-	da sinistra a destra	no
not	unario	destra	-	-
and, or	binario	sinistra	da sinistra a destra	sì
?:	ternario	-	da destra a sinistra	no

e la seguente espressione:

```
(not x) = y or z = w and not p ? a or b : not c
```

- a) Rappresentare l'albero della espressione.
- b) Specificare la semantica operativa della espressione.

NB: Il linguaggio di specifica operativa è così caratterizzato:

- Contiene gli operatori di negazione (!), uguaglianza (==) ed assegnamento (:=);
- Non contiene gli operatori **and**, **or**, ne l'operatore **?:**, ne le costanti logiche **true**, **false**;
- Ogni operatore non può essere applicato ad espressioni, ma solo a variabili;
- Contiene le istruzioni condizionali (*if-then* ed *if-then-else*) i cui predicati possono essere solo variabili;
- Contiene l'istruzione **return** il cui argomento è la variabile che contiene il valore della espressione;
- L'esecuzione della **return** termina immediatamente l'esecuzione del programma di specifica operativa.

3. Definire nel linguaggio *Scheme* la funzione **take** che, ricevendo una intero $n \geq 0$ ed una *lista*, restituisce i primi n elementi di *lista*. Nel caso in cui n sia maggiore della lunghezza di *lista*, **take** restituisce *lista*. Ecco alcuni esempi:

n	lista	(take n lista)
3	(1 2 3 4 5)	(1 2 3)
3	(a b)	(a b)
0	(a b c)	()
2	(a (b) c ())	(a (b))
0	()	()

4. Definire nel linguaggio *Haskell*, mediante la notazione di pattern-matching, la funzione **naturals**, avente in ingresso un intero $n \geq 0$, che computa la lista dei primi n numeri naturali (partendo da zero), come nei seguenti esempi:

n	naturals n
0	[]
1	[0]
2	[0,1]
5	[0,1,2,3,4]

5. Assumendo di avere una base di fatti *Prolog* relativa alla specifica di una serie di famiglie, come nel seguente esempio (ogni fatto specifica il padre, la madre ed i figli di una famiglia):

```
famiglia(guido, ester, [franco, nella, elena]).
famiglia(bruno, nella, [andrea, dario, zen]).
famiglia(franco, lucia, [giovanni, paola, letizia, sofia]).
famiglia(antonio, elena, [francesco, maria, maddalena]).
...
```

si chiede di specificare il predicato **cugini(X,Y)** che risulta vero qualora **X** ed **Y** siano cugini di primo grado, entrambi da parte di madre (ad esempio, **andrea** e **maria**).

6. Nell'ambito del paradigma orientato agli oggetti, definire e giustificare (sulla base di un semplice esempio) la regola di covarianza del parametro di uscita nei metodi.