

Programmazione I

Compitino del 21 gennaio 2000

1. Si considerino le seguenti definizioni di tipo:

```
TYPE
  tipolista = ^nodolista;
  nodolista = RECORD
    info: integer;
    pros: tipolista
  END;
```

Scrivere in Pascal una **PROCEDURE** che riceva, tramite un parametro di tipo **tipolista**, un puntatore a una lista di numeri interi, e modifichi tale lista inserendo al suo inizio un nuovo nodo contenente il massimo tra i valori presenti nella lista data. Ad esempio, se la lista contiene inizialmente i valori **9 7 16 13**, dopo l'esecuzione della **PROCEDURE** dovrà contenere **16 9 7 16 13**. Se la lista data è vuota, la **PROCEDURE** dovrà lasciarla immutata.

2. Si considerino le seguenti definizioni di tipo:

```
TYPE
  tipoalbero = ^nodoalbero;
  nodoalbero = RECORD
    info: integer;      {informazione contenuta nel nodo}
    sx, dx: tipoalbero {puntatori ai sottoalberi sinistro e destro}
  END;
```

Scrivere in Pascal una **FUNCTION** che riceva, tramite un parametro di tipo **tipoalbero**, il puntatore a un albero di numeri interi, e restituisca come risultato la somma di tutti i quadrati perfetti memorizzati nell'albero.

Per verificare se un numero è un quadrato perfetto, si supponga di disporre di una **FUNCTION** **quadrato** (**n: integer**): **boolean** (di cui non è richiesta la codifica), che restituisca **true** se il parametro utilizzato nella chiamata è un quadrato perfetto, **false** altrimenti.

3. Disegnare l'albero di ricerca ottenuto inserendo, uno dopo l'altro, i numeri **10 15 20 5 8 12 14 13 22 16** in un albero inizialmente vuoto. Scrivere gli output prodotti visitando tale albero nei tre ordini anticipato, simmetrico e posticipato.

4. Scrivere l'output prodotto da ciascuno dei seguenti programmi.

```
PROGRAM p1 (output);
```

```
  VAR
    p, q: ^integer;
```

```
BEGIN {p1}
  new(p);
  new(q);
  p^ := 7;
  q^ := p^;
  q^ := p^ * p^;
  p^ := q^ DIV p^;
  writeln(p^, q^);
END. {p1}
```

```
PROGRAM p2 (output);
```

```

VAR
  p, q: ^integer;

BEGIN {p2}
  new(p);
  new(q);
  p^ := 7;
  q := p;
  q^ := p^ * p^;
  p^ := q^ DIV p^;
  writeln(p^, q^);
END. {p2}

PROGRAM p3 (output);

VAR
  p, q: ^integer;

BEGIN {p3}
  new(p);
  new(q);
  p^ := 7;
  q^ := -p^;
  q^ := p^ * p^;
  p^ := q^ DIV p^;
  writeln(p^, q^);
END. {p3}

```

5. Per ognuna delle seguenti linee di codice individuare delle dichiarazioni di variabile ed eventualmente di tipo, in modo che le istruzioni che vi appaiono risultino corrette dal punto di vista della compatibilità dei tipi. Se ciò non fosse possibile, spiegare il motivo.

- $a^{\wedge}[b^{\wedge}] := \text{chr}(\text{ord}(\text{NOT}(b^{\wedge})))$
- $a^{\wedge}.b := a^{\wedge}.c \langle \rangle \text{ord}(a^{\wedge}.b > 'A')$
- $a.b^{\wedge} := \text{trunc}(a.c) \langle \rangle \text{ord}(a.b = \text{NIL})$