

TABELLE RIASSUNTIVE PER L'ANALISI DEI DATI

DATI DI INPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
b	REAL	STATICA	$b > 0$	Misura della base del rettangolo
h	REAL	STATICA	$h > 0$	Misura dell'altezza del rettangolo

DATI DI OUTPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
p	REAL	STATICA	$p > 0$	Misura del perimetro del rettangolo secondo la formula $p = 2 * (b + h)$
a	REAL	STATICA	$a > 0$	Misura dell'area del rettangolo secondo la formula $a = b * h$

DATI DI ELABORAZIONE (LAVORO) DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)				
Nome variabile o nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

PSEUDOCODIFICA

```

1  ALGORITMO Rettangolo_3
   PROCEDURA main ( )
2  /* Dichiarazione variabili di input */
   b, h : REAL
   /* Dichiarazione variabili di output */
   p, a : REAL
   INIZIO
   /* Acquisizione dei dati di input */
   RIPETI
     Scrivi ("Inserisci la base ")
     Leggi (b)
     SE (b <= 0)
       ALLORA
         Scrivi ("La base deve essere positiva! ")
       FINE SE
     FINCHE' (b > 0)
     RIPETI
       Scrivi ("Inserisci l'altezza ")
       Leggi (h)
       SE (h <= 0)
         ALLORA
           Scrivi ("L'altezza deve essere positiva! ")
         FINE SE
       FINCHE' (h > 0)
   /* Calcolo dei dati di output attraverso l'utilizzo dei dati di input */
   p ← 2 * (b + h)
   a ← b * h
   /* Comunicazione dei dati di input */
   Scrivi ("Il perimetro e': ")
   Scrivi (p)
   Scrivi ("L'area e': ")
   Scrivi (a)
   FINE
    
```

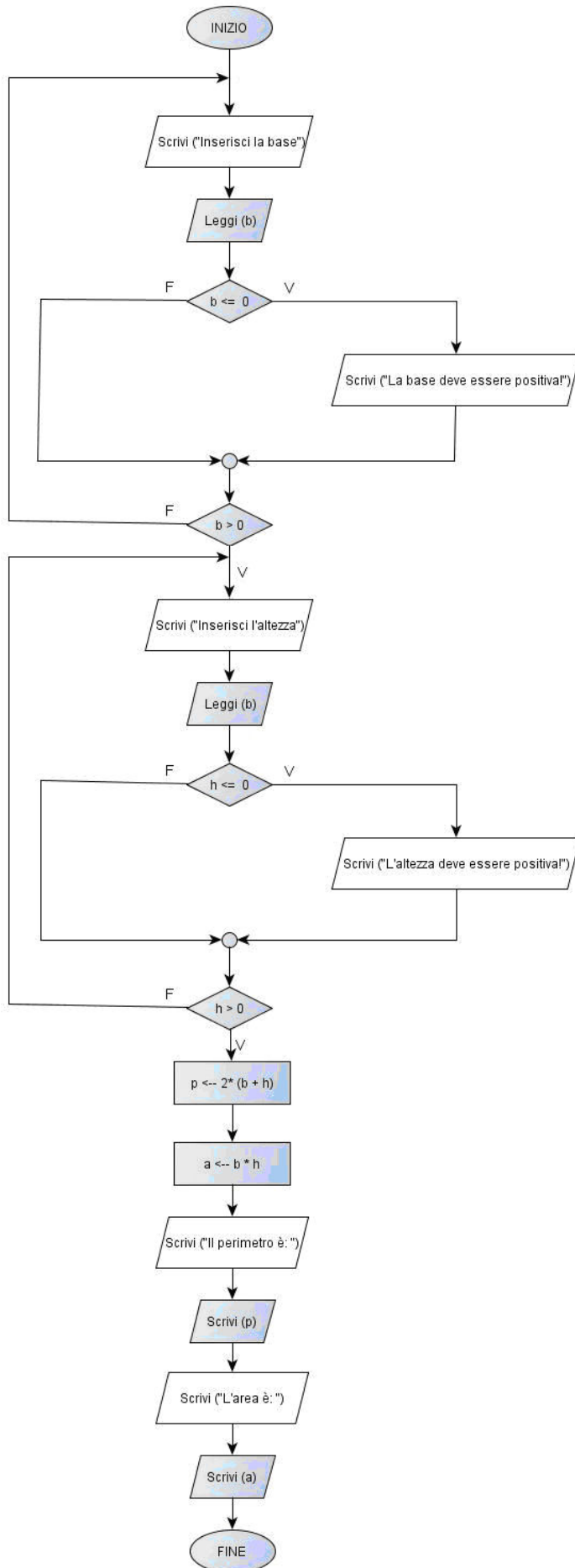
1 Ambiente GLOBALE
 Quanto scritto qui (di solito dichiarazioni di variabili) è visibile da qualunque parte dell'algoritmo

2 Ambiente LOCALE
 Quanto scritto qui (di solito dichiarazioni di variabili) è visibile solamente all'interno della PROCEDURA main ()

3 Corpo della PROCEDURA main ()
 Qui vengono scritte le istruzioni che riportano la sequenza del processo risolutivo dell'algoritmo

N.B Su sfondo bianco le istruzioni USER-FRIENDLY.
Questo flow-chart è stato realizzato grazie al software gratuito YED

FLOW-CHART



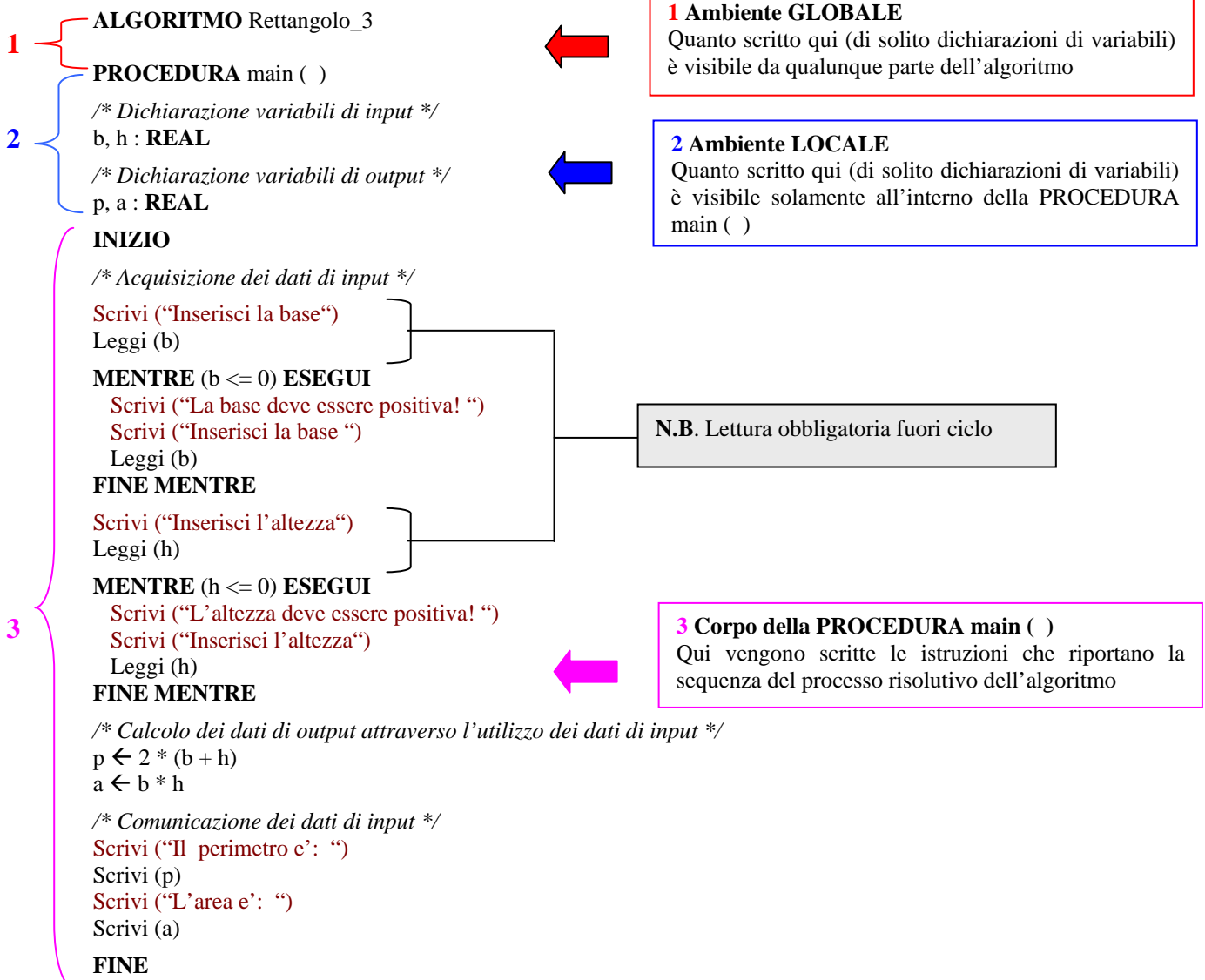
N.B. Sarebbe stato possibile usare per il controllo dei valori ammessi di una variabile in input anche l’istruzione di controllo iterativa di tipo **MENTRE** al posto di quella di tipo **RIPETI** effettuando una lettura preventiva del valore della variabile da controllare fuori dal ciclo (soluzione meno efficiente).

Questo accorgimento si deve a causa della differenza funzionale specifica tra i due tipi di cicli che prevede in sintesi:

- iterativa di tipo **MENTRE**: PRIMA avviene l’esecuzione del controllo della condizione logica (ciclo con verifica della condizione logica in TESTA o pre-condizionale) e POI l’esecuzione del blocco di istruzioni presente al suo interno;

- iterativa di tipo **RIPETI**: PRIMA avviene l’esecuzione del blocco di istruzioni al suo interno e POI l’esecuzione del controllo della condizione logica (ciclo con verifica della condizione logica in CODA o post-condizionale).

PSEUDOCODIFICA



N.B Su sfondo bianco le istruzioni USER-FRIENDLY.
Questo flow-chart è stato realizzato grazie al software gratuito YED

FLOW-CHART

