



- Codici impostati da completare
- Indicazioni per realizzare i codici passo per passo

# Disuguaglianza triangolare e area del triangolo

► Per prima cosa occorre creare le variabili di appoggio usate dal programma (Variabili e liste | Crea variabile).

1. a, b, c per i lati
2. area area
3. perimetro perimetro
4. p semiperimetro

► Il primo passaggio del codice è l'acquisizione delle misure dei lati e la loro memorizzazione in tre variabili.

Per questo usiamo un blocco personalizzato.

È necessario verificare che i dati immessi siano compatibili con la costruzione di un triangolo.

Acquisendo i valori in ordine crescente, come specifica richiesta, il codice si semplifica e applica la verifica per la disuguaglianza triangolare solo alla condizione tra il lato di lunghezza maggiore e gli altri due ( $c > a + b$ ).

Si gestisce in questo modo anche il caso del triangolo degenerare ( $c = a + b$ ).

Solo se è soddisfatta la **disuguaglianza triangolare**, si procede con il resto del codice.

► Una volta che sono disponibili i dati dei lati, occorre procedere alla verifica di costruibilità usando il connettivo logico "o" (OR).

Il costrutto "se ... altrimenti ..." consente di eseguire i passaggi del codice secondo i casi.

In caso positivo, calcoliamo il perimetro e l'area usando due blocchi distinti e personalizzati.

```

definisci input
dire Inserisci, dal minore al maggiore, i valori dei lati del triangolo. per 2 secondi
chiedi Inserisci il valore minore e attendi
porta a ← a risposta
chiedi Inserisci il secondo valore e attendi
porta b ← a risposta
chiedi Inserisci il valore maggiore e attendi
porta c ← a risposta

definisci 2p
porta perimetro ← a + b + c

```

```

definisci 2p
porta perimetro ← a + b + c

definisci area
porta p ← a perimetro / 2
porta area ← a radice quadrata di p * (p - a) * (p - b) * (p - c)

```

Alla fine, restituiamo a video il risultato.

```

quando si clicca su
se c > a + b o c = a + b allora
dire IL TRIANGOLO NON ESISTE o è DEGENERE per 2 secondi
altrimenti
2p
area
dire unione di >> 2p = unione di perimetro unione di >> area = area
ferma tutto

```

## PROVA TU

- 1 Verifica che il programma dia i risultati attesi per triangoli noti. Per fare delle verifiche usa prima i lati del triangolo rettangolo di 3 cm, 4 cm e 5 cm. Utilizza come secondo test i dati di tre segmenti con cui non è possibile costruire un triangolo, ad esempio, 3 cm, 4 cm e 7 cm. Esegui ora dei test con altri valori dei lati, anche presi dal libro di testo, e di cui conosci l'area. Scratch non dispone, contrariamente ad altri ambienti di programmazioni professionali, di un debugger per la verifica e l'individuazione di eventuali errori di programmazione, detti *bug*.
- 2 Costruisci il codice per l'area di un rombo note le misure delle due diagonali. Esegui dei test con valori noti. Inserisci un controllo per evitare che sia introdotto un valore zero (0).
- 3 Costruisci il codice per l'area di un trapezio isoscele note le due basi e la sua altezza. Esegui dei test con valori noti.

```

se d1 = 0 o d2 = 0 allora
dire Una diagonale NON puo' essere 0 (zero) per 2 secondi
ferma tutto

```