



# PROGETTO CURRICOLO DIGITALE

10 ore

**Classe 1C**

**docente De Stradis Vita**

# Cos'è il pensiero computazionale?

La scuola e la vita di tutti i giorni ci pongono continue sfide, situazioni e problemi da affrontare: trovare delle soluzioni è spesso faticoso ma può essere anche divertente. In questo capitolo progetto vedremo che:

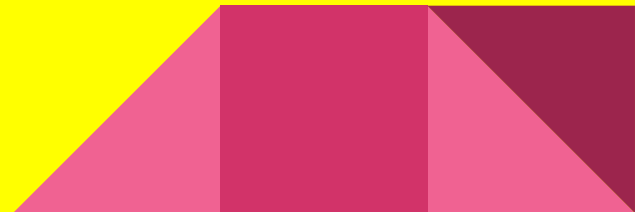
la tecnica del problem solving è un metodo che già applichi;

per risolvere i problemi può essere utile l'intuito, ma occorre anche metodo;

esistono problemi mal formulati, problemi con dati superflui, problemi irrisolvibili e problemi risolvibili ma con tempi di risoluzione enormi;

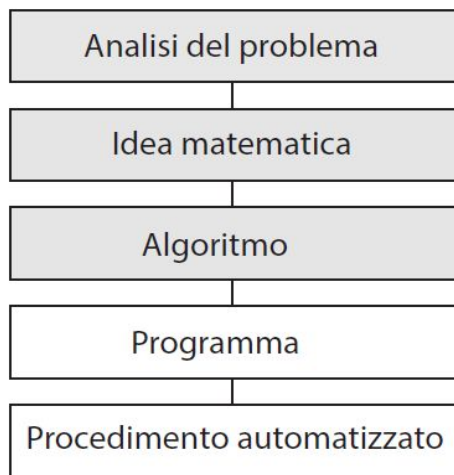
ci sono problemi che hanno come soluzione un risultato e altri problemi che hanno come soluzione un algoritmo;

affrontare un problema con gli strumenti del pensiero computazionale aiuta a trovare una strategia risolutiva efficace, rapida e generalizzata.



## Risolvere problemi (problem solving)

Di fronte a un nuovo compito da svolgere, la prima cosa da fare è analizzarlo in modo approfondito e trovare una *strategia* risolutiva; quando si tratta di un problema matematico, si parla di *algoritmo risolutivo* del problema.



I primi tre blocchi dello schema precedente indicano appunto il riconoscimento del problema, l'idea logico-matematica vincente per risolverlo, l'individuazione della strategia risolutiva. Gli ultimi due blocchi, una volta individuata tale strategia, indicano come affidarla a un computer. Per fare questo occorre trasformare la strategia in una sequenza di istruzioni scritte in un linguaggio che il computer comprende.

# ENTRIAMO NEL VIVO DEL PROGETTO

IN QUESTO [video](#) CI SONO DEI PAPERI CHE COMPIONO MOVIMENTI MOLTO SEMPLICI MA PRECISI.

TUTTI VOI SAPRESTE RIPETERLI PERCHE' IL NOSTRO CERVELLO E' CAPACE DI ACQUISIRE INFORMAZIONI DALL'ESTERNO, RIELABORARLE E UTILIZZARLE PER DARE RISPOSTE ADEGUATE.

I COMPUTER INVECE RIESCONO A ESEGUIRE SOLO UN'ISTRUZIONE PER VOLTA E SOLO SE CONOSCONO IL SIGNIFICATO DELL'ISTRUZIONE ASSEGNATA.

IN QUESTO BREVE CORSO IMPARERETE A "ISTRUIRE" UN SOFTWARE PER FARGLI RISOLVERE SEMPLICI PROBLEMI DI ARITMETICA E GEOMETRIA.

# UN GIOCO DELLE “CORNICETTE”


Gli alunni vengono suddivisi a coppie:

un/una alunno/a è il programmatore

l'altro/a alunno/a è il PC che deve eseguire le istruzioni

Vince la coppia che impiega il minor tempo a raggiungere l'obiettivo: eseguire su un foglio quadrettato la cornicetta sorteggiata!

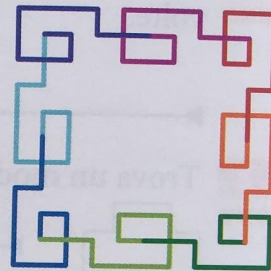
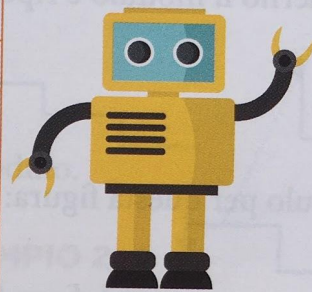
Per cominciare utilizziamo frecce vettori che indicheranno la direzione o la rotazione da eseguire.



Facciamo un gioco. Il robot *Cornicexpress* sa disegnare molti tipi di cornicette. Per fargli disegnare una cornicetta bisogna dargli una sequenza di simboli.

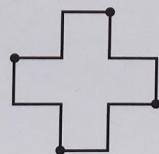
Simbolo	Significato
→	Fai un passo avanti (nella direzione attuale)
←	Fai un passo indietro (rispetto alla direzione attuale)
↶	Cambia la direzione attuale di 90° verso sinistra
↷	Cambia la direzione attuale di 90° verso destra
sequenza di simboli	Esegui i simboli nell'ordine
potenza	Ripeti tutti i simboli della base tante volte quante indicato dall'esponente

$((\rightarrow\curvearrowright\rightarrow\curvearrowright\rightarrow\curvearrowright\rightarrow\curvearrowright\rightarrow\curvearrowright\rightarrow\curvearrowright\rightarrow\curvearrowright)^2\curvearrowleft)^4$



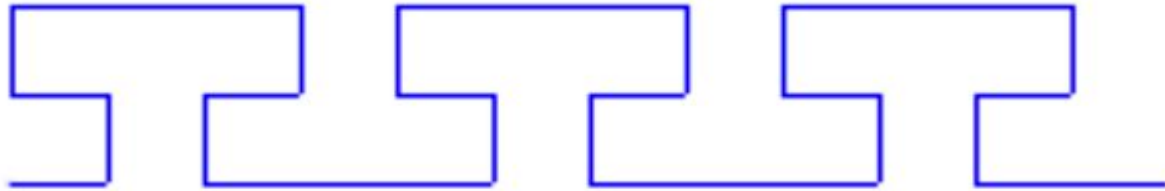
### ■ ESEMPIO

- $\rightarrow\curvearrowright\rightarrow$  significa: fai un passo, ruota a destra, fai un passo.
- $(\rightarrow\curvearrowleft)^4$  significa:  $\rightarrow\curvearrowleft\rightarrow\curvearrowleft\rightarrow\curvearrowleft\rightarrow\curvearrowleft$  perché la base viene ripetuta 4 volte; quindi si ottiene un quadrato di lato un passo.
- $\rightarrow\rightarrow\curvearrowleft\rightarrow\curvearrowleft\rightarrow\rightarrow\curvearrowleft\rightarrow$  produce un rettangolo di dimensioni due passi e un passo.
- $(\rightarrow\curvearrowleft\rightarrow\curvearrowright\rightarrow\curvearrowright)^4$  produce una croce così:



# Esercizio con algoritmi iterativi: costituiti cioè da una serie di azioni ripetute (ciclo)


**G-6.** Costruire un progetto che traccia la greca mostrata di seguito.



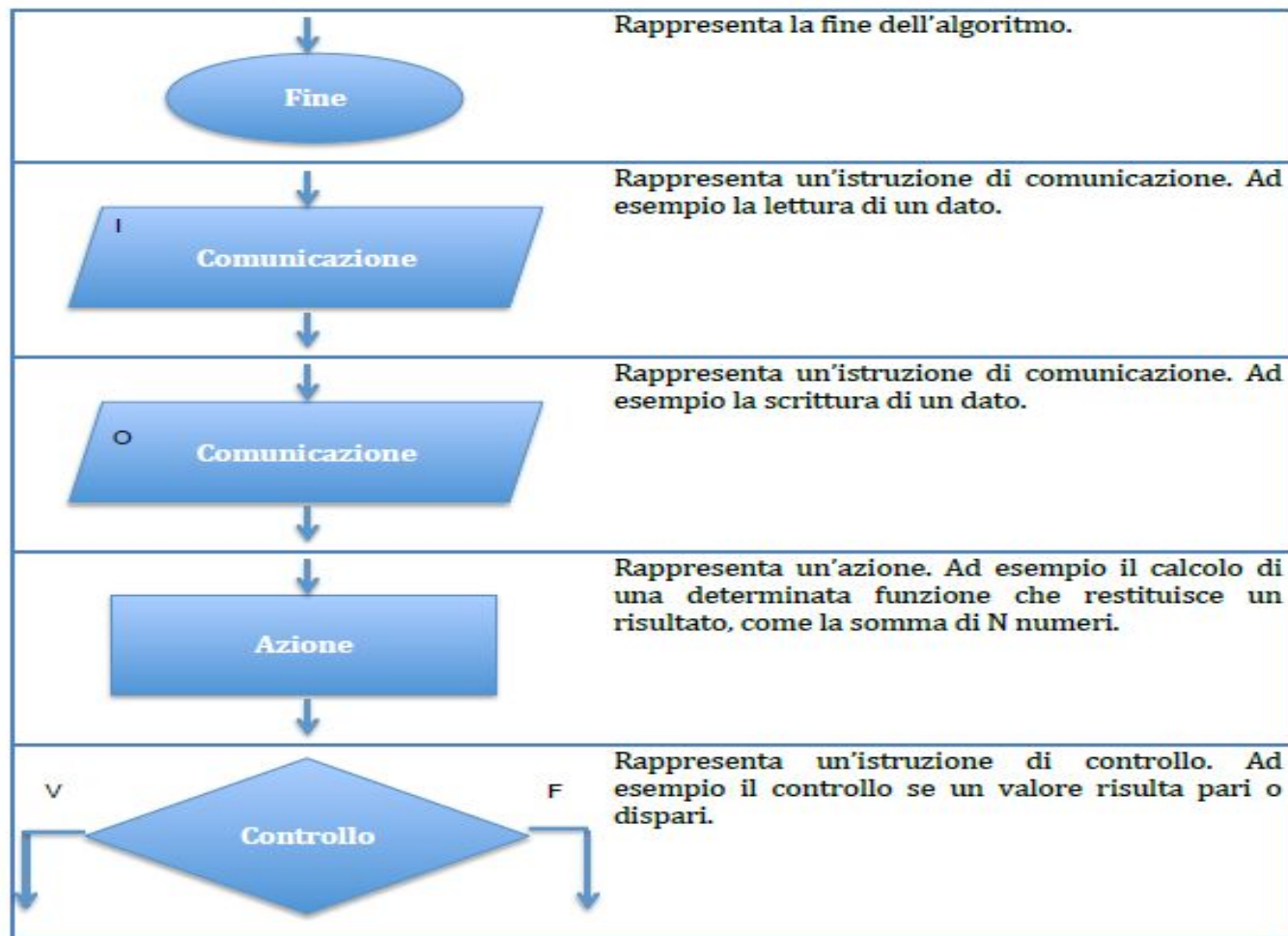
Utilizzare il comando `ripeti 3 volte.`



Gli algoritmi, però, possono anche essere rappresentati graficamente, mediante i diagrammi di flusso (flow diagrams), prima di essere realizzati sul calcolatore, al fine di avere una migliore visione di tutte le istruzioni necessarie per l'implementazione dell'algoritmo. Tali diagrammi utilizzano forme geometriche diverse per rappresentare istruzioni di tipo differente; la sequenzialità delle istruzioni viene rappresentata con delle frecce che collegano nell'ordine la varie istruzioni tra loro:

Forma utilizzata	Descrizione
	Rappresenta l'inizio dell'algoritmo.

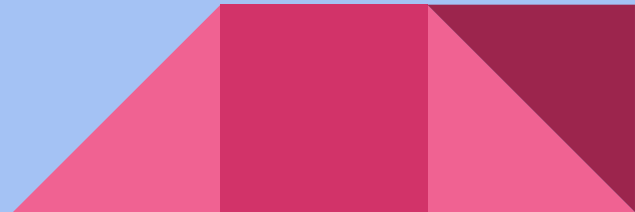




# Diagramma a blocchi

**Abbiamo già utilizzato i blocchi per rappresentare l'algoritmo necessario per la fattorizzazione di un numero.**

**Alla luce di ciò che imparerai in questo corso però ti renderai conto che può essere migliorato, reso più snello ed efficace.**



IN QUESTO CORSO IMPARERAI A PROGRAMMARE USANDO UN SOFTWARE, MOLTO INTUITIVO PERCHE' UTILIZZA IL LINGUAGGIO VISUALE.

# Impariamo a programmare

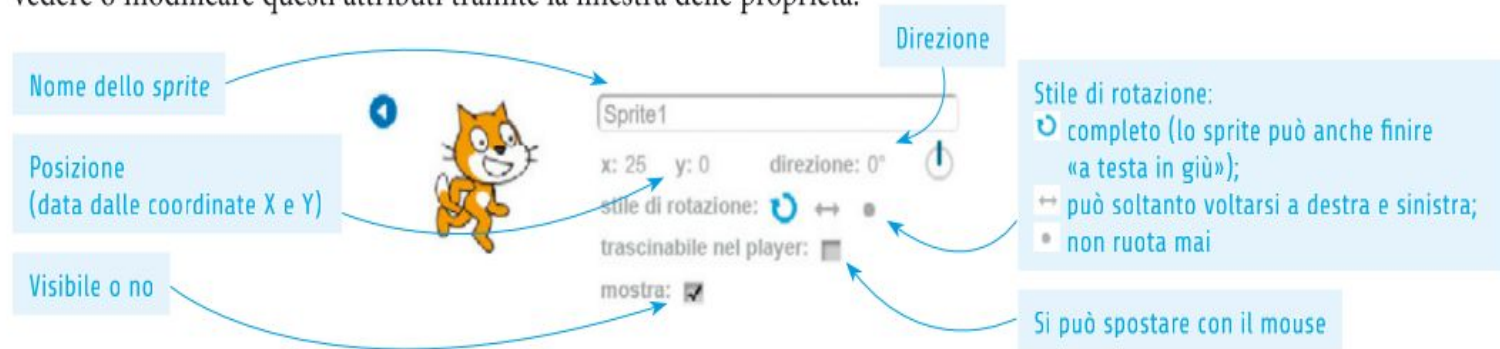
## Introduzione al linguaggio Scratch



# CODING CON SCRATCH

Scratch è un progetto didattico del MIT (Massachusetts Institute of Technology): è disponibile online ed è completamente gratuito. Ti permette di muovere e animare sullo schermo alcuni oggetti, detti *sprite*, associando a ogni oggetto una o più sequenze di blocchi che descrivono le azioni da far compiere loro.

Uno *sprite* ha una posizione, una direzione e uno stile di rotazione; può essere trascinato con il mouse per essere spostato e può essere visibile o invisibile. Puoi vedere o modificare questi attributi tramite la finestra delle proprietà.



The image shows the Scratch sprite editor interface with several annotations:

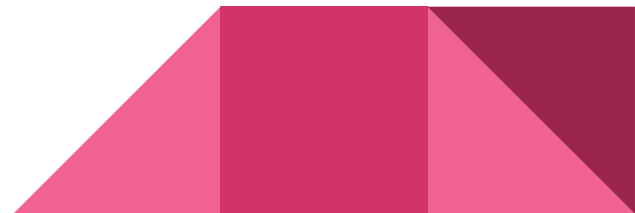
- Nome dello sprite**: Points to the text input field containing "Sprite1".
- Direzione**: Points to the direction control element, which includes a circular arrow icon and a numeric input field showing "0°".
- Stile di rotazione:** Points to the rotation style control, which includes three radio buttons: "completo" (selected), "solo a destra e sinistra", and "non ruota mai".
- Si può spostare con il mouse**: Points to the "trascinabile nel player" checkbox, which is currently checked.
- Posizione (data dalle coordinate X e Y)**: Points to the "x: 25 y: 0" coordinate display.
- Visibile o no**: Points to the "mostra:" checkbox, which is checked.

On the left side of the interface, there is a small stage area showing a cat sprite with a blue arrow pointing to it, indicating its current position and direction.

# Dove scaricare Scratch?

PER ESEGUIRE IL DOWNLOAD GRATUITO CLICCA SUL SEGUENTE LINK E SEGUI LE ISTRUZIONI

<https://scratch.mit.edu/download>



Puoi installare l'editor di Scratch 2.0 per lavorare sui tuoi progetti senza dover usare una connessione internet. Questa versione funziona su Mac, su Windows e su alcune versioni di Linux (32 bit).

[installazione](#)[Aggiornamenti](#)[Altre Versioni di Scratch](#)[Problemi conosciuti](#)

Note for Mac Users: the latest version of Scratch 2.0 Offline requires Adobe AIR 20. To upgrade to Adobe AIR 20 manually, go [here](#).

1

### Adobe AIR

Se non è ancora installato sul tuo computer scarica e installa l'ultima versione di [Adobe AIR](#)

Mac OS X - [Scarica](#)  
Mac OS 10.5 & Precedenti - [Scarica](#)  
Windows - [Scarica](#)  
Linux - [Scarica](#)

2

### Editor Offline di Scratch (Beta)

Ora scarica e installa l'Editor Offline di Scratch 2.0

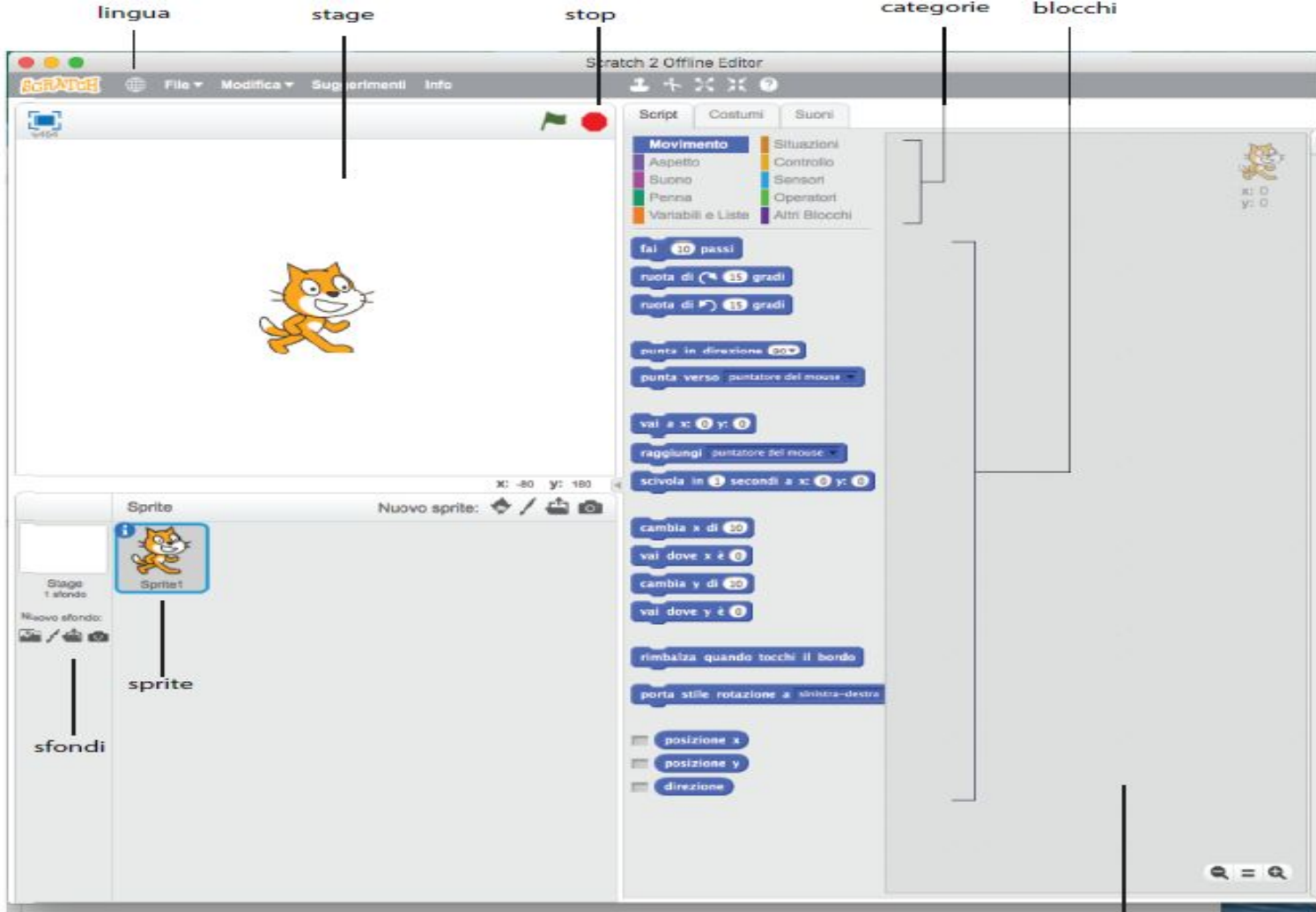
Mac OS X - [Scarica](#)  
Mac OS 10.5 & Precedenti - [Scarica](#)  
Windows - [Scarica](#)  
Linux - [Scarica](#)

3

### Materiali di Supporto

Hai bisogno di aiuto per iniziare? Ecco alcune utili risorse.

Progetti per Iniziare - [Scarica](#)  
Guida per Iniziare - [Scarica](#)  
Le Schede di Scratch - [Scarica](#)



zona di lavoro



Il gatto Cody, è lo sprite standard del linguaggio Scratch (ma sono disponibili molti altri sprite).

Il rettangolo in alto a sinistra in cui compare Cody è lo **Stage** cioè la scena; il retroscena è un sistema di coordinate cartesiane che va da  $x = -240$  a  $x = 240$ ,  $y = -180$  a  $y = 180$ . Per scoprire le coordinate dei vari punti basta muovere il puntatore e guardare i valori  $x$ ,  $y$  nella striscia sotto lo Stage.

Possiamo anche nascondere Cody. Andiamo nella parte inferiore sinistra della schermata principale. Sotto la scritta **Sprite** compare l'immagine di Cody entro un quadrato; nell'angolo a sinistra in alto compare una lettera "i", cliccando la quale si apre un menu. Togliendo il segno di spunta dalla voce "mostra", Cody sparisce (figura 2).

**Figura 2**



## Con Scratch

- calcoleremo i divisori di un numero naturale qualunque
- fattorizzeremo un numero
- calcoleremo il M.C.D tra due numeri
- classificheremo le frazioni
- disegneremo poligoni regolari



I programmi realizzati dai corsisti, individualmente o in coppia, sono stati pubblicati sul sito ufficiale di Scratch.

Di seguito i link

FATTORIZZAZIONE DI UN NUMERO

<https://scratch.mit.edu/projects/313410464>

DIVISORI COMUNI DI DUE NUMERI

<https://scratch.mit.edu/projects/313414545>

DIVISORI PROPRI DI UN NUMERO (esclusi i divisori banali)

<https://scratch.mit.edu/projects/313413634>



CLASSIFICAZIONE DI FRAZIONI

<https://scratch.mit.edu/projects/313412346>

ANGOLI COMPLEMENTARI

<https://scratch.mit.edu/projects/316253179>

IL TRAPEZIO ISOSCELE

<https://scratch.mit.edu/projects/316255055>

TRIANGOLO EQUILATERO DI LATO VARIABILE (con calcolo del perimetro)

<https://scratch.mit.edu/projects/314983573>



POLIGONO REGOLARE DI N LATI

<https://scratch.mit.edu/projects/314984019>

ESAGONO DI LATO VARIABILE CON CALCOLO DEL  
PERIMENTO

<https://scratch.mit.edu/projects/314983371>

