



Piccoli Programmatori

Corso di coding dai 5 ai 14 anni.
Prima si gioca, poi si programma.

Piccoli Programmatori



Manuale del Programmatore



Questo manuale è di _____

≡ Come si usa

Ogni pagina spiega un'idea dei programmatori, con le carte, con i blocchi di Scratch e con Python. In alto trovi le icone dei livelli: **cerca la tua**, sono le pagine per te.



Esploratori · 5-7 anni

Carte e robot. Leggi il manuale con un adulto!



Inventori · 8–10 anni

Carte e Scratch.



Creatori · 11–14 anni

Scratch e Python.

Le 3 regole d'oro

1. Il computer fa **esattamente** quello che gli dici, niente di più.
2. Un bug non è un fallimento: è un mistero da risolvere.
3. Prova, cambia una cosa, riprova.

In fondo trovi la tabella «Da Scratch a Python» e le pagine per i tuoi appunti.

☰ Ancora appunti

≡ **Le mie idee di progetti**

Giochi, storie, invenzioni: scrivi o disegna tutto quello che vorresti creare.

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Che cos'è un programma



Un **programma** è una lista di istruzioni per il computer. Il computer è velocissimo, ma fa solo quello che c'è scritto.

Quasi ogni programma fa tre cose:



Esempio: premi la freccia (entrata) → il programma calcola dove va il gatto
→ il gatto si sposta sullo schermo (uscita).

Linguaggi: le istruzioni si possono scrivere in tanti modi. Al corso usiamo le carte, i blocchi di Scratch e il linguaggio Python.

Prova tu!

Trova 3 oggetti di casa che seguono istruzioni: cosa ricevono, cosa fanno, cosa mostrano? (Esempio: il microonde.)





**Con le
carte**



In Scratch

fai **100** passi

```
from turtle import *
forward(100)
forward(100)
right(90)
forward(100)
```

Scrivi con le frecce la strada dalla tua camera alla cucina. Poi falla seguire a qualcuno, alla lettera!



Da Scratch a Python

Idea	Scratch	Python
Scrivere	dire Ciao	<code>print("Ciao")</code>
Chiedere	chiedi Come ti chiami?	<code>nome = input("Come ti chiami?")</code>
Variabile	porta punti a 0	<code>punti = 0</code>
Cambiare	cambia punti di 1	<code>punti = punti + 1</code>
Ripetere	ripeti 4 volte	<code>for i in range(4):</code>
Per sempre	per sempre	<code>while True:</code>
Se... allora	se ... allora	<code>if punti > 10:</code>
Altrimenti	altrimenti	<code>else:</code>
A caso	numero a caso tra 1 e 6	<code>random.randint(1, 6)</code>
Funzione	definisci salta	<code>def salta():</code>

In Python gli spazi all'inizio della riga contano: dicono quali istruzioni stanno «dentro» un ciclo o un se.



Bug e debug



Un **bug** è un errore nel programma. Fare **debug** vuol dire cercare il bug e correggerlo. Capita a tutti, ogni giorno!

Il bug
dov'è?



Volevo andare a destra, ma ho usato la carta sbagliata.

Come si caccia un bug

1. **Rileggi** le istruzioni una alla volta.
2. **Esegui con il dito** (o con Roby) e guarda dove succede la cosa strana.
3. **Cambia una cosa sola** e riprova.
4. **Spiegalo ad alta voce** a un compagno (o a un pupazzo!): spesso il bug salta fuori da solo.

In Python
manca una
virgoletta

```
print("Ciao!)    # bug!
print("Ciao!")   # corretto
```

Prova tu!

La prossima volta che trovi un bug, scrivilo nel tuo Diario di bordo: cos'era e come l'hai risolto.





Ciclo: ripetere



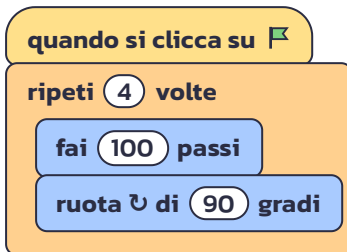
Un **ciclo** ripete le stesse istruzioni più volte. Così il programma è più corto.

Con le
carte



Roby fa il giro di un quadrato e torna alla partenza.

In Scratch
disegna un
quadrato



In Python
disegna un
quadrato

```
from turtle import *
for i in range(4):
    forward(100)
    right(90)
```

Per sempre: in Scratch il blocco «per sempre» ripete senza mai fermarsi. Utile nei giochi!

Prova tu!

Come disegneresti un triangolo con un ciclo? Suggerimento: ripeti 3 volte, gira di 120 gradi.



Internet, IA e sicurezza



Internet è una rete di computer che si parlano tra loro. Il tuo messaggio viaggia diviso in tanti pacchetti, che poi si rimettono insieme.

L'**IA** è un programma che impara da tanti esempi. Può sbagliare: controlla sempre quello che dice.

Le regole della sicurezza online

- La password è segreta: la fanno solo i tuoi genitori.
- Una buona password è lunga: tre parole a caso più un numero.
- Non scrivere online indirizzo, scuola o numero di telefono.
- Se qualcosa ti mette a disagio, chiudi e parlane con un adulto.
- Condividi solo progetti con foto e voci che tutti hanno detto di poter usare.

Prova tu!

Inventa una password lunga con 3 parole a caso (non scriverla qui!). Riesci a ricordarla domani?



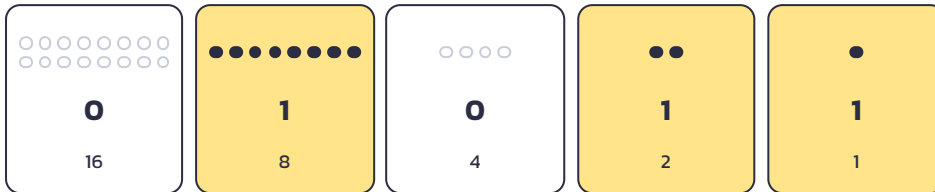


Bit e codice binario



Un **bit** è l'informazione più piccola: 0 oppure 1. Con tanti bit in fila il computer scrive numeri, lettere e colori.

Ogni carta vale il doppio di quella alla sua destra. Gira a faccia in su (1) quelle che ti servono e somma i pallini:



$$01011 = 8 + 2 + 1 = 11$$

Con 5 carte scrivi tutti i numeri da 0 a 31. Con 8 bit (un **byte**) arrivi fino a 255!

Prova tu!

Scrivi in binario la tua età e il giorno del tuo compleanno.

_____ e _____



Evento: quando succede...

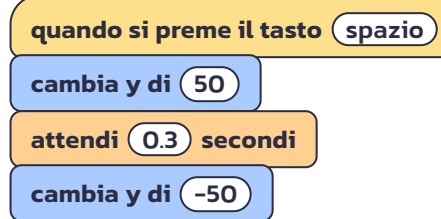


Un **evento** è una cosa che succede e fa partire il programma. Per esempio un clic, un tasto, un tocco.

Nella vita Quando suona la sveglia → mi alzo.
Quando suona la campanella → esco in cortile.

In ScratchJr Il blocco giallo «inizia al tocco»: il personaggio si muove quando lo tocchi.

In Scratch



Il personaggio salta quando premi la barra spaziatrice.

Prova tu!

Pensa a un gioco: quali tasti usi? Cosa succede quando li premi?





Condizione: se... allora



Con una **condizione** si sceglie: se succede questo, allora fai quello.
Se no, fai un'altra cosa.

Nella vita Se piove, allora prendo l'ombrello, altrimenti prendo il cappello.

In Scratch
il labirinto

per sempre

se sta toccando il colore nero? allora

vai a x: -200 y: 0

In Python

```
eta = 9
if eta >= 8:
    print("Livello Inventori!")
else:
    print("Livello Esploratori!")
```

Prova tu!

Inventa 3 regole «se... allora» per un tuo gioco. Esempio: se tocco la stella, allora prendo 5 punti.



Lista: tante cose in fila



Una **lista** è una variabile con tanti valori in fila. Per esempio i punteggi migliori.

In Python

```
mosse = ["sasso", "carta", "forbice"]
print(mosse[0])          # sasso
mosse.append("lucertola")
print(len(mosse))        # 4
```

Attenzione: in Python si conta da 0. Il primo elemento è `mosse[0]`, il secondo `mosse[1]`.

Con random

```
import random
print(random.choice(mosse))
```

In Scratch

aggiungi 120 a classifica

Prova tu!

Fai una lista dei tuoi 5 cibi preferiti in Python e fai scegliere al computer la cena di stasera.



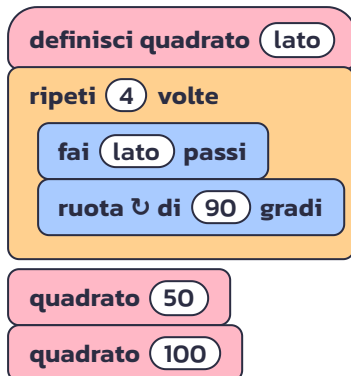


Funzione: un blocco tutto tuo



Una **funzione** è un gruppo di istruzioni con un nome. La crei una volta e la usi quando vuoi.

In Scratch
I miei blocchi



In Python

```

from turtle import *
def quadrato(lato):
    for i in range(4):
        forward(lato)
        right(90)

quadrato(50)
quadrato(100)
  
```

Prova tu!

Crea una funzione «stella» che disegna una stella. Poi disegna un cielo con 5 stelle.

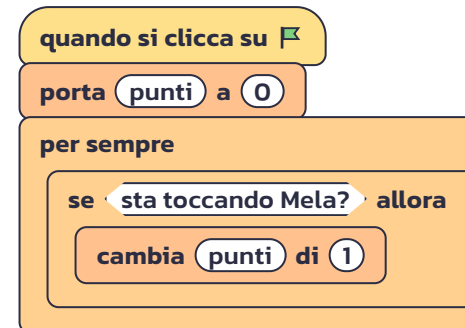


Variabile: una scatola con un nome



Una **variabile** è una scatola con un nome, che ricorda un valore. Per esempio i punti, le vite, il tempo.

In Scratch



In Python

```

punti = 0
punti = punti + 1
print("Punti:", punti)  # Punti: 1
  
```

Attenzione: il nome è come un'etichetta sulla scatola. Scegli nomi chiari: `punti` è meglio di `x`.

Prova tu!

Quali variabili servono in un gioco di corsa? Scrivine almeno 3.

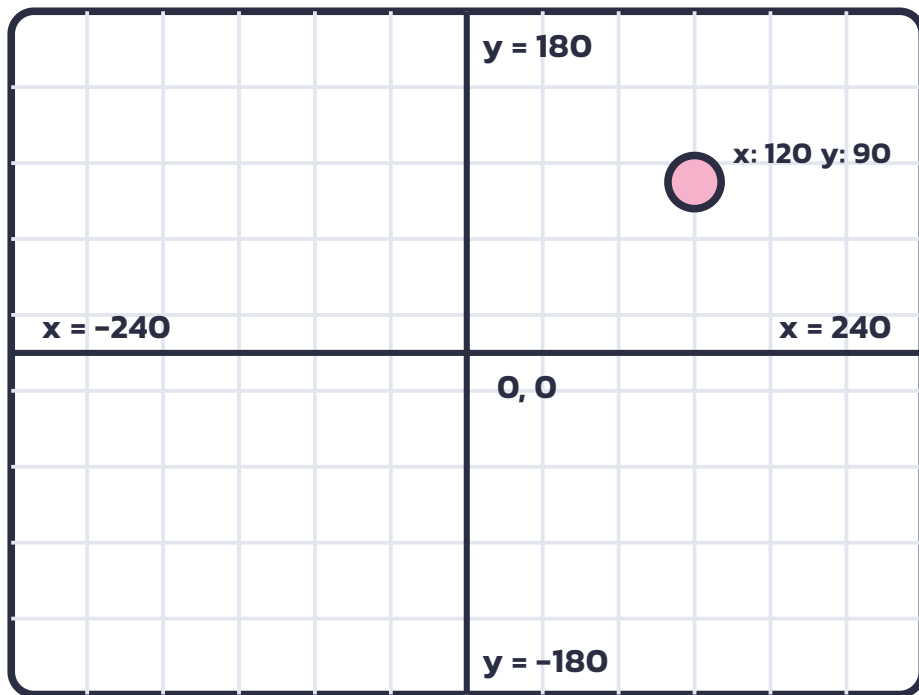




Coordinate x e y



Le **coordinate** dicono dove sta un punto: x va di lato, y va su e giù.



Il centro dello stage di Scratch è **x: 0, y: 0**. Verso destra x cresce, verso l'alto y cresce. È come la battaglia navale!

In Scratch

vai a x: 120 y: 90

cambia x di 10

Prova tu!

Dove si trova un punto con x: -240 e y: 180? Segnalo sulla griglia.



Numeri a caso e messaggi



Un **numero a caso** è un numero scelto dalla sorte. I **messaggi** fanno parlare tra loro i personaggi.

Numeri a caso
Scratch

vai a x: (numero a caso tra -200 e 200) y: 150

Numeri a caso
Python

```
import random
dado = random.randint(1, 6)
print(dado)
```

Messaggi
Scratch

invia a tutti fine partita

quando ricevo fine partita

dire Hai vinto!

Prova tu!

Lancia un dado vero 10 volte e segna i numeri. Escono sempre gli stessi? È questo il «caso».

