

LA PIOGGIA

SIMULARE LA PIOGGIA

Come già detto nell'introduzione, simulare significa "ricostruire la realtà", in particolare quell'aspetto che rappresenta il nostro oggetto di simulazione.

In questo piccolo progetto proveremo a simulare la pioggia, dapprima le gocce che la compongono e successivamente la traccia lasciata dalla loro caduta.

Per ricostruire un aspetto della realtà, è necessario effettuare preventivamente delle osservazioni del fenomeno in modo da comprenderlo appieno.

Limitando quindi la nostra osservazione ad una superficie ben definita su cui cadono le gocce di pioggia, si possono osservare alcuni aspetti:

1. la dimensione di ogni goccia può variare
2. non è possibile prevedere dove cadranno le prossime gocce
3. le gocce cadono con frequenza non costante.

Si possono allora stabilire dei valori per i parametri di simulazione relativi a ciascuno degli aspetti osservati; ad esempio:

1. la dimensione delle gocce sia al massimo di 2 mm
2. la distanza massima tra una goccia e l'altra sia di 10 cm
3. il tempo massimo che intercorre tra la caduta di una goccia e l'altra sia di 1 sec

Ovviamente, al variare dei valori assegnati ai parametri nel range fissato per ognuno verrà simulata una pioggia più o meno intensa.

Quindi, simulare la "pioggia", significa in realtà generare in modo casuale i valori per ciascuno dei tre parametri, cioè una terna (dim, dist, t).

Ma analizzando in modo più approfondito la questione, ci si accorge che generare il solo valore della distanza tra una goccia e l'altra non è sufficiente in quanto la prossima goccia si potrebbe trovare in un

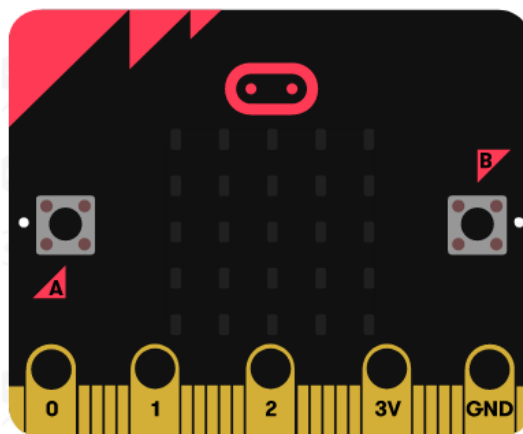
punto qualsiasi di una circonferenza centrata nell'ultima goccia caduta. Quindi la terna deve essere completata con un quarto elemento, l'angolo variabile tra 0° e 360° : (dim, dist, ang, t).

Quando però si passa all'implementazione vera e propria, cioè si sceglie un ambiente tecnologico di supporto per la realizzazione pratica della simulazione, l'algoritmo che si andrà a realizzare dovrà tener conto delle peculiarità e delle limitazioni introdotte dalla scelta effettuata.

UNA SCELTA APPLICATIVA

Una delle possibili scelte applicative potrebbe ricadere sulla scheda micro:bit per una serie di motivi:

- consente la programmazione visuale a blocchi alla portata anche dei più piccoli
- mette a disposizione un simulatore della scheda stessa, quindi non obbliga a possederne una
- consente di vedere immediatamente i risultati della programmazione



Le caratteristiche dello strumento fisico scelto, impongono anche delle scelte implementative: la matrice di 25 led messa a disposizione distribuisce i led in 5 righe e 5 colonne numerate da 0 a 4; il led in

alto a sinistra ha coordinate (0,0) quello in basso a destra (4,4), quindi, ogni goccia di pioggia può essere rappresentata da uno qualsiasi dei led e la sua caduta potrebbe essere simulata dal "passaggio" del led da spento ad acceso; inoltre, poiché non è possibile agire sulla dimensione della goccia-led si sceglie di rappresentarla mediante l'intensità del colore, e si potrà agire sulla casualità della posizione del prossimo led da accendere e sui tempi che intercorrono tra una accensione e la successiva.

Suddividiamo ora il problema in sotto-problemi e procediamo quindi per step.

Simulare le gocce

STEP 1: accendere led in una posizione casuale.

Il codice che permette di realizzare questo primo step è il seguente:

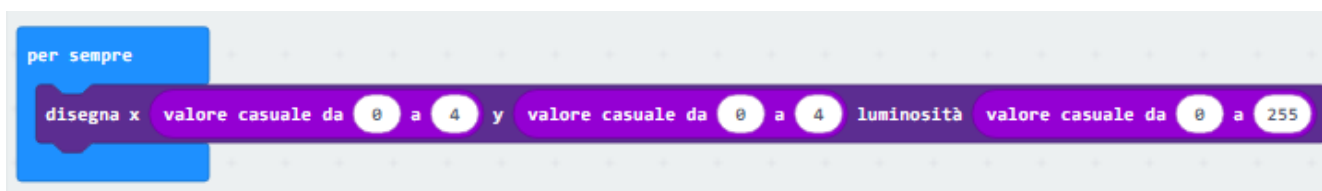


In un blocco "per sempre" viene acceso un led in posizione casuale ; banalmente, ci si accorge che dopo un tempo molto breve la matrice di led sarà tutta accesa, cioè la superficie su cui cade la pioggia simulata è tutta bagnata.

STEP 2: simulare la dimensione della goccia/led.

Come abbiamo osservato, è possibile simulare la dimensione della goccia di pioggia generando in modo casuale la luminosità di

accensione del led (si ricorda che la scelta per il parametro "luminosità" può ricadere tra 0 e 255):

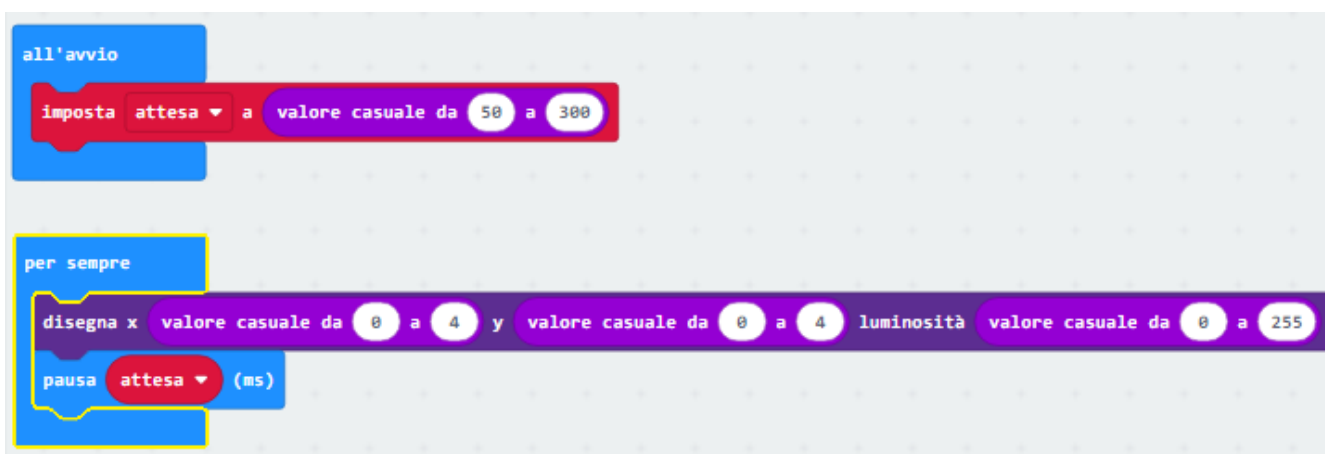


Nel blocco "per sempre" sostituiamo il blocco "disegna" precedentemente utilizzato con quello che prevede anche l'assegnazione di un valore alla luminosità del led.

STEP 3: simulare l'intensità della pioggia

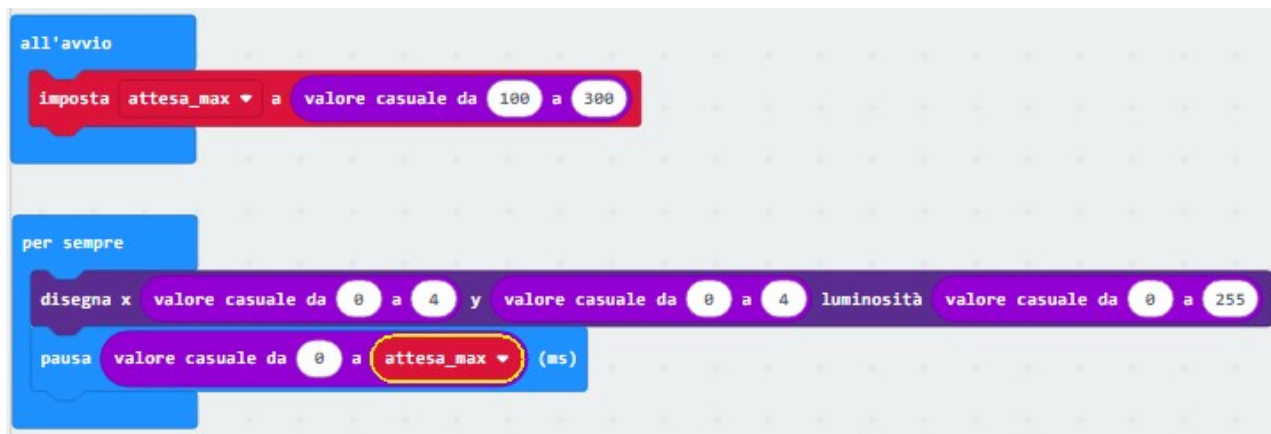
Ora ci manca solo l'aspetto legato all'intensità della pioggia, cioè al tempo che intercorre tra la caduta di una goccia e l'altra.

L'intensità può essere simulata introducendo dopo l'accensione del led una pausa di durata casuale, ad esempio tra un tempo di attesa minimo di 50 ms e un tempo di attesa massimo di 300 ms. Tale valore casuale può essere impostato "all'avvio" e utilizzato per l'intera simulazione:



Oppure, "all'avvio" può essere generato in modo casuale un tempo di attesa massimo, ad esempio tra 100ms e 300 ms, e il tempo di attesa

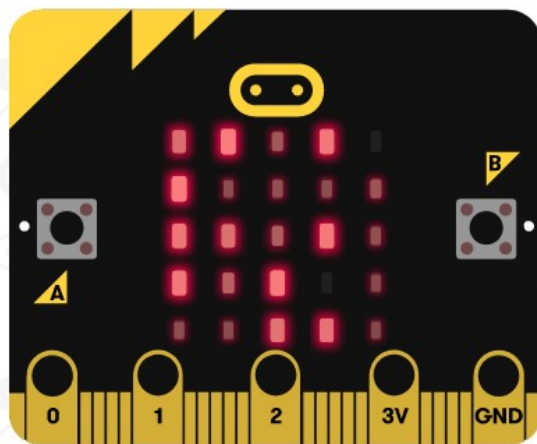
tra una goccia e l'altra può variare in modo casuale tra 0 e quel valore massimo:



```
all'avvio
imposta attesa_max a valore casuale da 100 a 300

per sempre
disegna x valore casuale da 0 a 4 y valore casuale da 0 a 4 luminosità valore casuale da 0 a 255
pausa valore casuale da 0 a attesa_max (ms)
```

Ecco quindi un possibile risultato della simulazione:



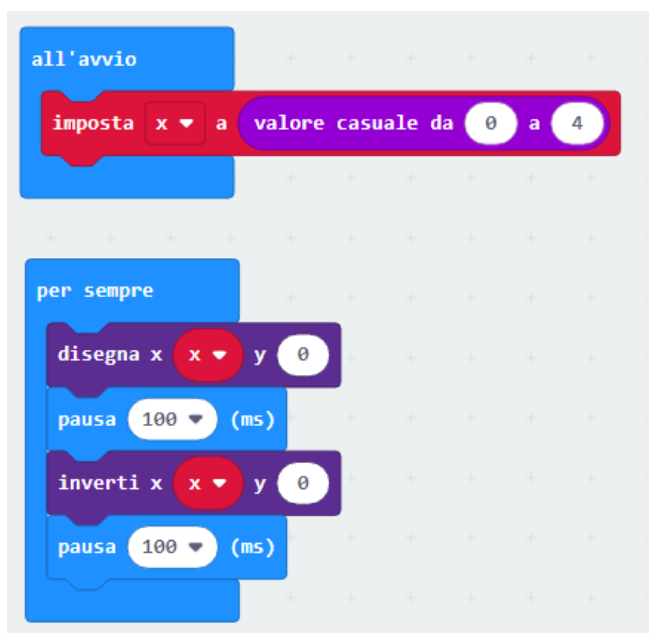
Interessante osservare che nel corso della spiegazione del procedimento applicativo viene utilizzato indifferentemente il riferimento alla realtà simulata (goccia) o all'elemento di simulazione che la rappresenta (led), cioè si stabilisce una corrispondenza biunivoca tra questi due elementi, tanto da poterne utilizzare uno al

posto dell'altro, per cui in questo contesto la goccia diventa led e il led rappresenta la goccia.

Simulare la traccia lasciata dalle gocce

STEP 1: accendere e spegnere un led della prima riga in una colonna di posizione casuale.

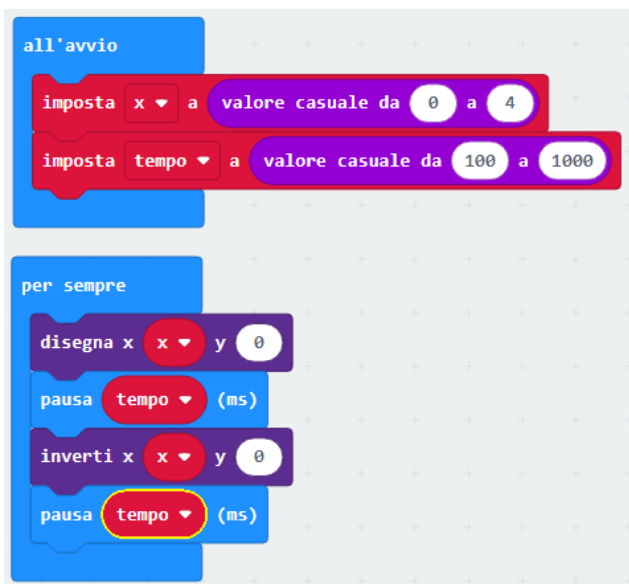
Il codice che permette di realizzare questo primo step è il seguente:



“All’avvio”, dopo aver definito una variabile denominata per convenzione x , le viene assegnato un valore casuale tra 0 e 4, cioè la posizione di colonna del led che andremo ad accendere e successivamente a spegnere.

Viene acceso il led di posizione $(x,0)$, si attende un tempo di 100ms, viene spento lo stesso led, si attende per altri 100 ms e poi vengono ripetuti indefinitamente questi stessi passi nel blocco “per sempre”.

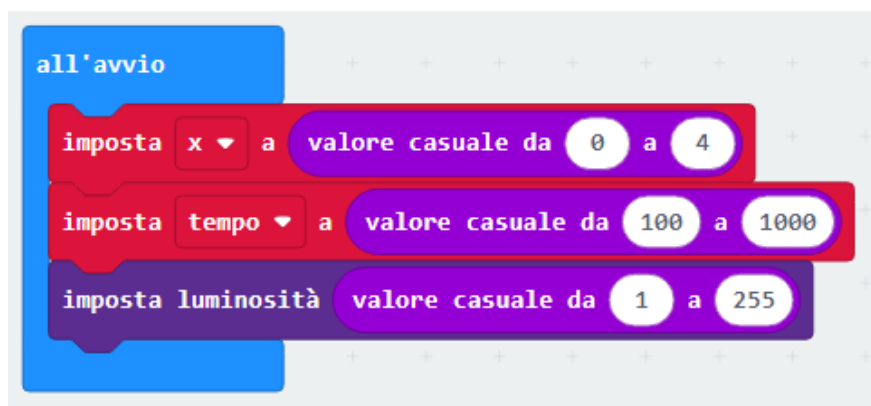
La modifica del tempo di attesa renderà più o meno veloce il processo e ciò può essere realizzato creando una variabile *tempo* alla quale assegnare "all'avvio" un valore casuale e imponendo che il blocco pausa prelevi la durata della pausa da tale variabile.



```
all'avvio
imposta x a valore casuale da 0 a 4
imposta tempo a valore casuale da 100 a 1000

per sempre
disegna x x y 0
pausa tempo (ms)
inverti x x y 0
pausa tempo (ms)
```

La dimensione della goccia può essere quindi realizzata mediante l'intensità della luminosità del led e per questo sarà sufficiente introdurre "all'avvio" il blocco luminosità della categoria led:

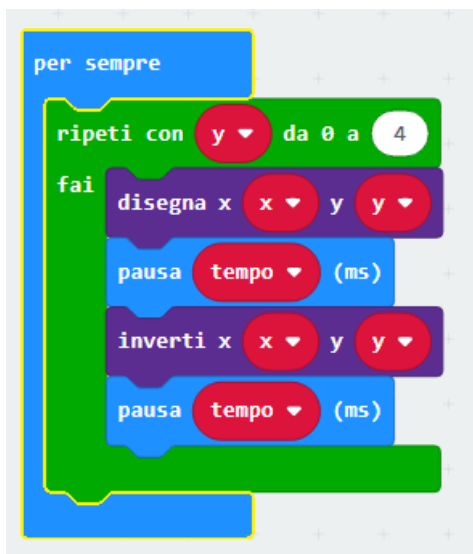


```
all'avvio
imposta x a valore casuale da 0 a 4
imposta tempo a valore casuale da 100 a 1000
imposta luminosità a valore casuale da 1 a 255
```

STEP 2: far scendere la goccia-led

Per simulare la caduta della goccia è necessario far in modo che una volta spento il led che si trova in riga 0 venga acceso quello corrispondente in riga 1 per poi passare a riga 2 e così fino a riga 4, ovviamente sempre nella stessa colonna.

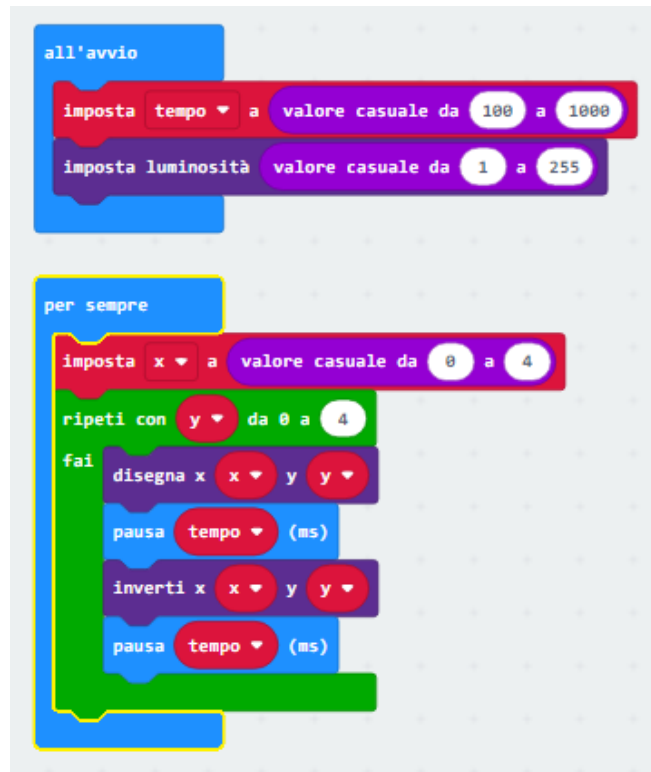
Per realizzare questo è possibile utilizzare il ciclo "ripeti con" in modo da far variare la seconda componente della coppia che rappresenta le coordinate del led, per convenzione denominata *y*, il cui valore sarà contenuto in una variabile creata proprio con nome mnemonico *y*.



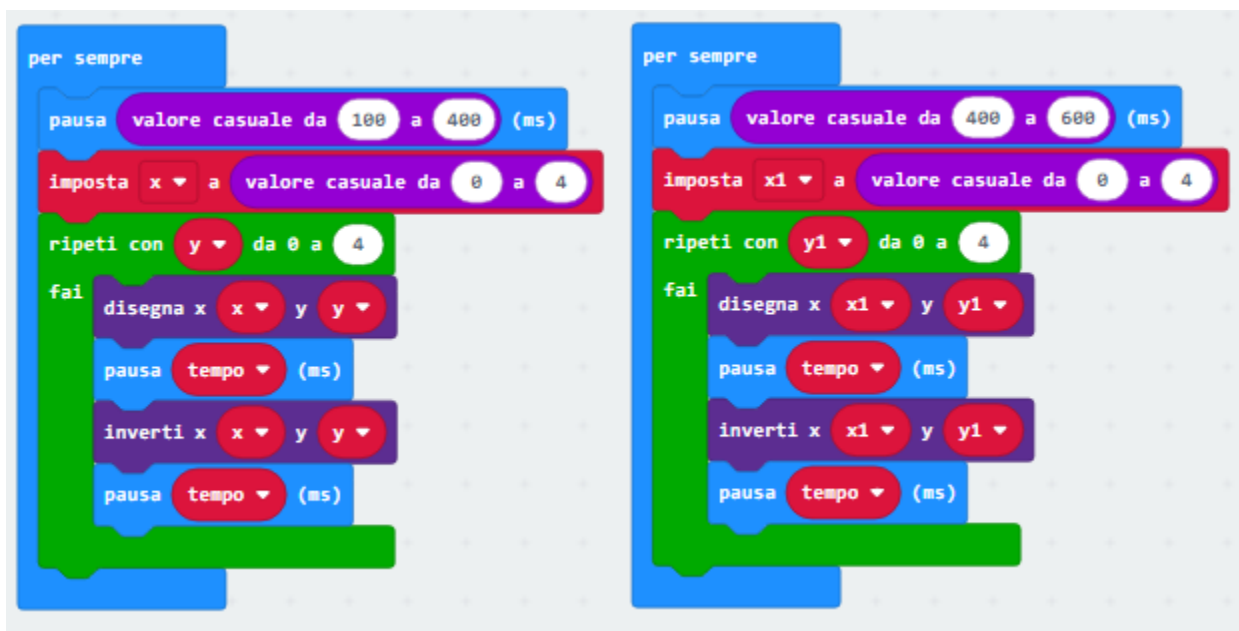
STEP 3: far scendere più gocce-led

Ultimo passo sarà quello di simulare la pioggia vera e propria facendo cadere più gocce una dopo l'altra in colonne sempre diverse.

La prima idea potrebbe essere quella di spostare il blocco "imposta" per assegnare il valore casuale per la variabile x dal blocco "all'avvio" al blocco "per sempre":



Ma questo codice farà sì che la nuova goccia possa cadere solo quando quella precedente abbia concluso la propria caduta. Conviene allora duplicare il blocco "per sempre" e creare altre due variabili $x1$ e $y1$ che permettano di gestire una nuova goccia differenziando anche la pausa iniziale prima che la goccia successiva inizi la propria corsa:



Il blocco "per sempre" può essere nuovamente duplicato e possono essere definite altre due variabili x_2 e y_2 per aggiungere una terza goccia.

Purtroppo potrà accadere che nei tre diversi blocchi "per sempre" venga generato e assegnato alle rispettive variabili x , x_1 e x_2 esattamente lo stesso valore; quindi le tre gocce potranno cadere nella stessa colonna anche se in tempi diversi.

Sarà interessante sperimentare modificando i limiti per la generazione casuale dei tempi per ottenere effetti molti diversi nella simulazione.

CONCLUSIONE

La generazione di numeri casuali è evidentemente fondamentale per questa simulazione: è stata utilizzata sia per la posizione delle gocce di pioggia che per la intensità-luminosità e per i tempi di caduta. La modifica dei valori assegnati come limiti per questi parametri consente di ottenere situazioni sempre diverse.