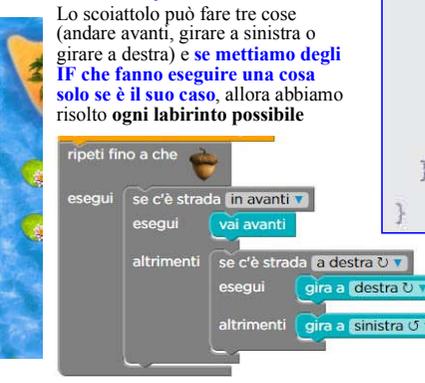
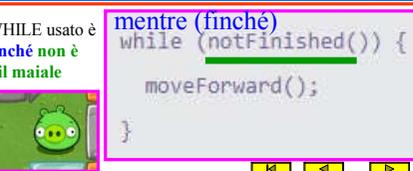
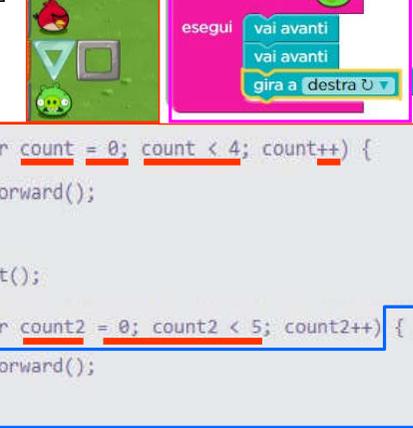
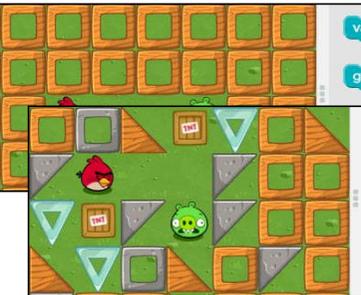


Labirinto classico (7)

Di solito la programmazione è fatta scrivendo un testo detto **codice** (o scrivendo **pseudocodice**), ma qui useremo **Blockly**, che utilizza blocchi colorati da unire tra loro per scrivere programmi.

Un **uccello rosso** si muove verso un **maiale verde** in un labirinto e può **avanzare di un passo**, o **svoltare a sinistra** o **svoltare a destra**.



```
while (notFinished()) {
  if (isPathForward()) {
    moveForward();
  } else {
    if (isPathRight()) {
      turnRight();
    } else {
      turnLeft();
    }
  }
}
```

In JavaScript (=>Wk)

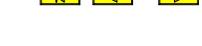
```
moveForward();
moveForward();
moveForward();
```

```
while (notFinished()) {
  moveForward();
  if (isPathLeft()) {
    turnLeft();
  }
}
```

```
while (notFinished()) {
  if (isPathForward()) {
    moveForward();
  } else {
    altrimenti
    turnLeft();
  }
}
```

```
for (var count = 0; count < 4; count++) {
  moveForward();
}
turnLeft();
for (var count2 = 0; count2 < 5; count2++) {
  moveForward();
}
```

```
while (notFinished()) {
  moveForward();
}
```



Strutture di controllo

Ripetizione (o iterazione o ciclo) (FOR o WHILE)

Selezione semplice

Selezione (IF...)

Selezione multipla

Selezione multipla

Lo scoiattolo può fare tre cose (andare avanti, girare a sinistra o girare a destra) e se mettiamo degli IF che fanno eseguire una cosa solo se è il suo caso, allora abbiamo risolto ogni labirinto possibile

Ciclo standard per i vettori di N elementi

Il FOR è un ciclo pre-condizionale (nel quale quindi il corpo del ciclo può non essere eseguito mai) adatto a situazioni nelle quali si conosce il numero N di ripetizioni volute

Se non sappiamo quante volte va ripetuto il ciclo ma sappiamo che va ripetuto fino a che è vera una certa condizione (verificata all'inizio per cui posso non farlo neppure una volta), allora uso WHILE (condizione) {corpo} (MENTRE è vero che o FINCHE' è vero che)

Ciclo standard