

Introduzione a SCRATCH

Sfide e esercizi

**Hackspace
Festival letteratura
Mantova**

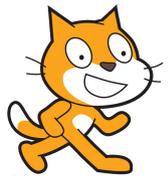
Stefano Lacaprara

stefano.lacaprara@gmail.com

INFN Padova

Mantova, 6-10 Settembre 2017

Sommario per oggi



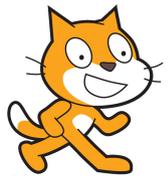
- **Riassunto prima lezione**
 - Cosa abbiamo imparato a fare l'altra volta
- **Nuovi elementi di programmazione**
 - Piu' interazione tra gli sprite
 - Uso delle condizioni:
 - mettere “intelligenza” in un programma
 - Uso delle variabili
 - Mettere “memoria” in un programma
 - ...
- **Oggi: consolidamento di quello che abbiamo imparato**
 - Vi proponno' una serie di esercizi via via piu' complessi
 - Descrizione scopo
 - Analisi elementi necessari per raggiungere lo scopo
 - Svolgimento
 - Possibile soluzione

Cosa abbiamo imparato la lezione 1



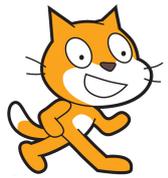
- **Installazione**
- **Introduzione agli elementi della IDE di SCRATCH**
 - Area di programmazione, stage, area degli sprite, librerie, ...
- **Elementi base di uso degli sprite**
 - Movimento
 - Suoni
 - Aspetto
 - Creazione/uso di piu' sprite
- **Elementi di programmazione**
 - Far partire un programma
 - Creare una sequenza di azioni
 - Azioni configurabili: (fare [10] passi)
 - Loop (cicli)
 - Azioni condizionali (Se ... allora ...)
 - Coordinazione (messaggistica)

Cosa ci manca?

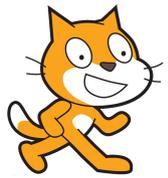


- **Definizione ed uso delle variabili**
 - In realta' qualcuna ne abbiamo gia' vista implicitamente
 - Costume,
 - Posizione,
 - Direzione,
 - Da introdurre e usare variabili nuove
- **Condizioni:**
 - Gia' usati: <condizione>
 - Proviamo a complicarci la vita, combinazione di condizioni
- **Introduzione degli operatori**
 - Combinare insieme le variabili
 - Combinare insieme le condizioni
 - Condizioni su variabili
- **Numeri casuali**
 - ...
 -

Come procediamo



- **Vorrei una sessione interattiva**
 - La prima sessione serve per introdurre un bel po' di concetti nuovi, forse anche troppi
 - Questa vorrei fosse dedicata all'utilizzo di quei concetti
 - Introducendone alcuni di nuovi
- **Propongo una serie di esercizi/sfide**
 - Piu' o meno in ordine di difficulta':
 - I primi banali/facili, gli ultimi saranno complessi sia come analisi del problema sia come soluzione
 - Proviamo a svolgere insieme questi esercizi
 - Potrebbe essere un buon percorso didattico anche per I vostri studenti



1. Il gatto che balla

1. Movimenti semplici di uno sprite

1. Cambiamento di aspetto con interazione (g/p) o suono

2. Il gatto e il drago

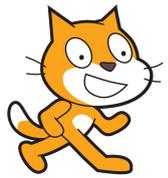
1. Due sprite: movimento e aspetto

1. Azioni condizionate dalla posizione, o interazione con colori o interazione tra sprite

3. Gatto e ballerina

1. Due sprite, interazione tramite messaggi

2. Gestione dello sfondo



4. Pippistrello che vola

1. Movimento, interazione con il mouse

5. Cane e gatto

1. Interazione tra sprite basata su distanza e non contatto.

1. Uso di condizioni su variabili basata su posizione

6. Compagnia di danza

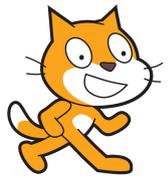
1. Molti sprite, musica, movimento, coordinazione

7. Cartolina di auguri

1. Anche interazione con utente

8. Anima il tuo nome

1. Simile



9. Indovina il numero

1. Input utente, numeri casuali, condizioni

10. Gare delle auto

1. Primo videogame!

1. Movimento, interazione, numeri casuali, coordinamento, variabili

11. Gatto che vola e cattura palloncini

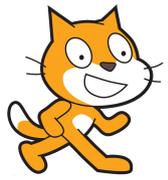
1. Videogame:

12. Disegno con una matita

1. Interazione con il mouse, uso sprite come penne

13. Campo fiorito

1. Applicazione di 12



14. Nascondino

- Game

15. Raccogli le mele che cadono

1. Game

16. Un sub va a pesca, ma attenti ai granchi!

17. Pong

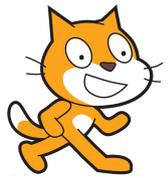
1. La pallina che rimbalza

18. Tris

1. Creiamo una scacchiera per Tris

19. Una pallina che cade

1. Un po' di fisica: pallina che cade (gravita') e rimbalza sul pavimento



20. Segui la linea

- Disegno una linea e il mio sprite deve seguirla (difficile!)

21. Nevica!

1. Cade la neve e si accumula sulle colline

22. Labirinto

1. Il mio sprite deve uscire da un labirinto senza toccare le pareti

23. Misurare le aree

1. L'area di una figura semplice (quadrato, cerchio, triangolo) sappiamo quando vale

1. Ma come si fa a misurare l'area di una figura complicata?

1. Metodo MonteCarlo (o del teppista)

1. E' un metodo che I fisici usano tantissimo



24. Battaglia spaziale

1. Astronave vola e combatte gli alieni

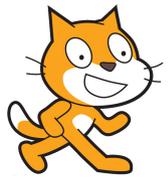
25. Space invaders

1. Classico gioco anni 80

26. Rally

1. Arrivare al traguardo senza uscire fuori strada

Es1: il gatto che balla



- **Creare un programma con un gatto che accenna qualche passo di danza**
 - **Analisi:**
 - Ballare e' (circa) ripetere a ritmo alcuni passi.
 - **Elementi necessari:**
 1. Muovere un passo avanti e indietro (**movimento: in avanti e all'indietro**)
 2. Ripetizione di una azione (**cicli**)
 3. Far partire un programma (**bandiera verde**)
 - **Opzionale:**
 - Ballare e' anche muovere il corpo (**costumi**)
 - Aggiungere un po' di musica (**suoni**)
 - Mettere uno sfondo adeguato ad un ballo (**sfondo**)
- **Facile**
 - essenzialmente e' il primo esercizio che abbiamo fatto nella prima sessione



- **Se premo il tasto G il gatto diventa piu' grande. Se premo il tasto P, diventa piu' piccolo.**
 - Diventa piu' grande/piccolo (**aspetto**)
 - Se premo il tasto G/P (**fare un'azione quando un certo tasto viene premuto**)
 - **Attenzione:** dopo che il programma e' partito (**bandiera verde**) devo stare ad aspettare per sempre se viene premuto il tasto G/P!
- **Quando si sente un suono forte, lo sprite cambia colore**
 - Cambia colore (**aspetto**)
 - Quando sento suono forte (**azione condizionale**)
 - **Condizione e' <seno un suono forte>**
 - Cercatela in [sensori]
 - Serve un microfono



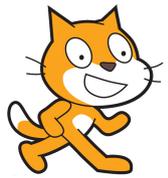
- **Muovo il gatto con il mouse**
 - [raggiungi posizione mouse]
- **Metto un secondo sprite**
 - Se il gatto tocca il secondo sprite, chiede scusa
 - Condizione Se sta toccando [secondo] sprite allora – dice [scusa]
- **Quando il gatto e' nel 25% superiore dello schermo dice "mi piace stare quassu'!"**
 - Facciamo muovere il gatto non solo avanti e indietro, ma anche in alto e in basso (o in diagonale)
- **Quando il gatto tocca qualcosa di blu sullo sfondo, suona una nota alta. Quando tocca qualcosa di rosso, ne suona una bassa.**
 - **Azioni condizionali**
 - Le **condizioni** sono: tocco un certo colore
 - Azione e':
 - **Suonare una nota particolare**

Es2: Il gatto e il dragone

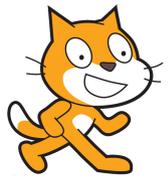


- Un gatto si muove avanti e indietro ed e' preoccupato, un dragone lo vede e ha l'acquolina in bocca, e dopo un po' sputa fiamme.
- **Analisi:**
 - Ho due personaggi da gestire:
 - Ognuno fa una cosa diversa.
 - Voglio che inizino a recitare insieme.
 - Voglio far parlare/pensare I miei personaggi.
- **Elementi:**
 - Devo creare un secondo sprite (dragone)
 - Lo posso trovare nella libreria di quelli disponibili (Fantasy)
 - Voglio che il dragone sputi fuoco (**costumi**)
 - Mi serve un secondo costume con il fuoco. Lo posso creare importandolo sempre da quelli disponibili (Fantasy)
 - Far parlare/pensare gli sprite (**aspetto**)
 - Attesa tra azioni diverse (**controllo/attendi**)
- **Opzionale:**
 - All'inizio, il dragone voglio che sia diretto in direzione del gatto
 - Oppure: all'inizio e' rivolto dalla parte opposta e si gira
- **Facile** Anche questo l'abbiamo sostanzialmente svolto nella prima sessione
-

Es3: Gatto e Ballerina

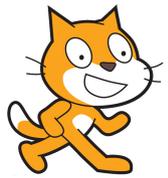


- **Un nuovo progetto:**
 - Proviamo a creare un gattino che cammina avanti e indietro per lo schermo, girandosi e miagolando quando arriva al bordo.
 - Usiamo uno sfondo appropriato: anzi, piu' d'uno. E vogliamo poter scegliere tra 2 diversi sfondi usando le frecce destra e sinistra della tastiera
 - Magari mettiamo anche una musica di sottofondo adeguata e diversa per i due sfondi.
 - Infine, mettiamo un ballerino che danza sullo sfondo, che cambia colore ogni volta che il gattino tocca il bordo dello schermo



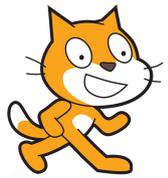
- **Scomponiamo l'azione complessa in elementi via via piu' semplici**
 - Dopo aver “risolto” gli elementi semplici, li metteremo insieme per creare un'azione via via piu' complessa
- **Gatto**
 - cammina avanti e indietro
 - si gira quando arriva al bordo
 - Miagola
- **Ballerina**
 - Balla
 - Cambia colore
 - Cambia colore quanto il gatto tocca il bordo
- **Sfondo**
 - Scelgo fondali adeguati
 - Cambio fondale quando premo un tasto

Es4: Pippistrello che vola



- **Creare uno scenario notturno, con un pipistrello che vola e si dirige verso il puntatore del mouse.**
- **Analisi:**
 - Devo disegnare uno sfondo notturno
 - Ho bisogno di uno sprite pippistrello
 - Devo far volare il pippistrello
 - Il pippistrello si deve muovere verso il puntatore del mouse
- **Elementi:**
 - Sfondo, sprite, costumi, movimento, **direzione verso mouse**
- **Facile:**
 - L'unica cosa nuova e' l'ultima:
 - Mi serve un nuovo elemento!

Interagire con il mouse



- **Ci sono diversi modi di interagire con il mouse**

- **Movimento:**

- **punta verso []**

- Lo stesso elemento si puo' usare per far puntare la direzione verso un altro sprite (se presente)



- **Raggiungi:**

- Questo elemento sposta lo sprite dove e' il mouse.
 - **punta verso** cambia la direzione dello sprite, ma non lo muove
 - Se vogliamo muoverlo dobbiamo farlo noi.



- **Sensori**

- **x-y del mouse**

- E' una variabile che contiene la x (y) di dove si trova il mouse

- **<tasto del mouse premuto>**

- E' una condizione: vero quando faccio click con il mouse

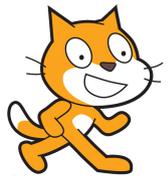


Interagire con il mouse



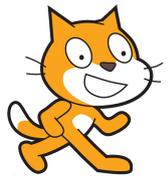
- **Per far volare in modo “naturale” il mio pippistrello devo:**
 - **Volare:**
 - Vuol dire sbattere le ali: sappiamo già’ come farlo
 - (cambio costume sprite ogni tot secondi)
 - **La direzione del pippistrello e’ [verso il mouse]**
 - **La sua velocita’, cioe’ il numero di passi per ogni ciclo, e’ sempre la stessa**
 - Se gli diciamo di seguire il mouse, il movimento puo’ essere troppo veloce e “innaturale”
 - Meglio dire solo di “puntare verso” il mouse, ma volare con la propria velocita’
 -

Sfida: seguire il mouse

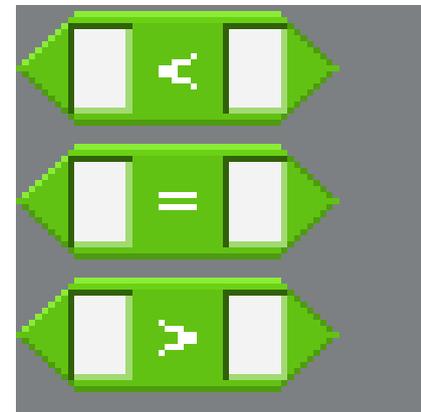


- **Il pippistrello segue il puntatore del mouse, ma non si avvicina mai troppo al puntatore stesso.**
- **Analisi:**
 - **Movimento** verso la **posizione del mouse**
 - Se mi **avvicino** troppo allora mi fermo
 - Avvicino: se **distanza** e' piu' piccola di X (**condizione su variabile**)

Es5: il gatto e il cane



- **Un gatto cammina avanti e indietro sullo stage. Sullo stage c'e' anche un cane. Se il gatto si avvicina troppo al cane si gira e scappa, prima di toccarlo.**
- **Analisi:**
 - Condizione e' **<uno sprite e' vicino ad un altro>** (NB e' diverso <tocca>, se il cane tocca il gatto, e' tardi per scappare!)
 - **<vicino>** vuol dire: la **distanza da** altro sprite e' **piu' piccola** di un certo **valore**
 - Condizione e' confronto tra una variabile (distanza) e un valore (**operatore**)
 - Azione e':
 - **Girati e scappa (movimento)**



Es6: la compagnia di danza



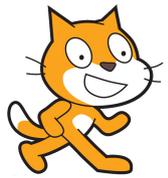
- **Su uno sfondo appropriato, ho diversi personaggi. Quando clicco su un bottone [VIA], tutti I personaggi iniziano a ballare.**
- **Analisi:**
 - **Sprite che ballano**
 - **Devono ballare quando clicco su pulsante [VIA]**
 - **Aspettano un messaggio**
 - **Il pulsante [VIA] e' anche lui uno sprite**
 - **Quando premo lo sprite, mando il messaggio a tutti I ballerini**

Es7 : cartolina di auguri



- **Facciamo una cartolina di auguri (per esempio per un compleanno)**
 - **La vogliamo interattiva**
- **Scegliamo un sfondo adeguato**
 - **Quando clicco sullo sfondo, I colori cambiano**
 - **Scegliamo un paio di figurine che sono sulla scena e fanno gli auguri**
 - **Prima uno e poi l'altro**
 - **Mettiamo una torta nel mezzo**
 - **All'inizio le candele sono spente**
 - **Quando clicco sulla torta parte una musica e si accendono le candele**
 - **E la torta diventa piu' grande e piu' piccola (per simulare la fiamma delle candele)**
 - **Se siete bravi, potete creare un nuovo costume della torta con la fiamma delle candele un po' cambiata, e ruotare tra I due costumi per far muovere le fiammelle.**
 - **Se clicco su un'altra figurina, questa fa una capriola.**

Es8: animate il vostro nome



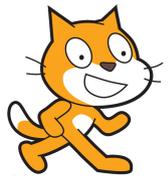
- **Scrivete il vostro nome e animatelo**
 - **Ogni lettera e' uno sprite**
 - **Ci sono diversi tipi di caratteri**
 - **Potete animare ogni lettera come piu' vi piace**
 - **Es: cambia colore, cambia dimensione, si muove, fa capriole, fa dei suoni, appare e scompare**
 - **Potete animarle tutte insieme**
 - **Oppure una dopo l'altra**
 - **Inizia la prima lettera, comunica [messaggi] quando ha finito alla seconda, che comincia quando riceve il messaggio e cosi' via fino all'ultima**
 - **Puo' essere animato anche lo sfondo**

Es9: indovina un numero



- **Il computer sceglie un numero casuale tra 1 e 100, l'utente deve indovinare il numero.**
 - Per rendere le cose piu' facili all'utente, il computer indichera' se il numero e' troppo alto o troppo basso
 - Quando il numero viene indovinato, si deve indicare la vittoria
 - E magari anche quanti tentativi sono stati necessari

Es9: analisi



- All'inizio devo scegliere un numero casuale tra 1 e 100
 - E ricordarmi che numero ho scelto (**Variabile**)
- L'utente deve poter immettere un numero (**Chiedi numero**)
 - Devo confrontare il numero di input con quello casuale (**confronto tra variabili**)
 - **Se** il numero e' piu' alto(basso) **allora** devo dire "e' piu' alto(basso)" (**azione condizionale**)
 - **Se** il numero e' giusto **allora** ho vinto
 - Annuncia vittoria in modo appariscente (**costume, suono, ...**)
- Devo **ripetere** l'immissione **finche'** non ho vinto (**Iterazione condizionale**)



- E' una variabile
 - Una variabile e' un numero che puo' variare a seconda di determinate condizioni.
 - Fai [10] passi:
 - [10] e' una costante: ho scritto 10 e non cambiera' finche' non lo cambiero' a mano.
 - Fai [N] passi:
 - Il risultato dipende da quanto vale N. E N potrebbe cambiare nel corso del programma.
 - Nel nosto caso, [numero casuale] e' evidentemente una quantita' che puo' cambiare ogni volta che eseguo quella istruzione.

Creare variabili



- Dalla libreria **Variabili** [nuova variabile]



- Si apre una finestra:

- Nome della variabile

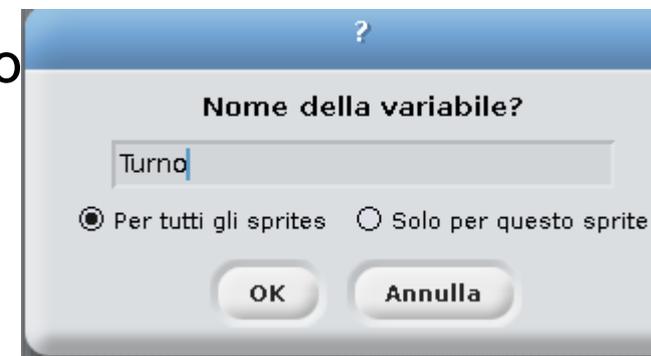
- Mia scelta, mettiamo un nome significativo

- Per tutti gli sprite/solo questo sprite

- Chi puo' usare la variabile

- tutti o solo questo sprite

- Per noi il turno e' qualcosa che e' lo stesso per tutti gli sprite, attiene a tutto il gioco.



- Appaiono nuovi elementi in **Variabili**

- Per mostrare la variabile sullo stage

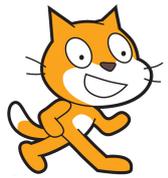
- Per dare un valore alla variabile

- Per cambiare il valore della variabile (es +42)
Var=Var+42

- Per mostrare o nascondere la variabile

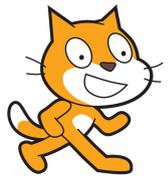


Es10: La corsa delle macchine

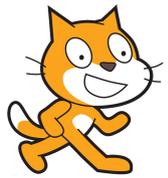


- **Voglio creare una corsa tra due macchine:**
 - Le auto partono insieme su una strada dritta, e si muovono, sempre in rettilineo, verso un un traguardo.
 - Il movimento in avanti di ciascuna auto deve essere casuale, in modo da rendere piu' avvincente la gara.
 - Oppure le auto si muovono in avanti ogni volta che si preme un tasto (diverso per le due auto)
 - La prima auto che arriva al traguardo vince, e lo proclama





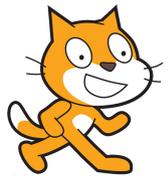
- **Analisi:**
 - **Ho bisogno di uno scenario con una strada**
 - **Ho bisogno di due auto**
 - **Inizialmente devo mettere le auto sulla linea di partenza**
 - **Al via, le auto si muovono verso il traguardo**
 - **L'avanzamento di ogni auto deve essere casuale**
 - **Quando un'auto raggiunge il traguardo deve succedere:**
 - **L'auto stessa dichiara di aver vinto**
 - **L'altra auto si ferma dove si trova.**
- **Un po' piu' difficile!**
 - **Ci mancano alcuni elementi:**
 - **Come mi accorgo che un'auto ha raggiunto il traguardo?**
 - **Come faccio a muovere un'auto di un numero casuale di passi?**
 - **Per cominciare mi concentro sul resto dei problemi e lascio questo per ultimo: inizialmente mi muovo di N passi per volta.**



Per come ho scritto l'analisi, dovrebbe essere evidente che questa e' una condizione

- **Quando un'auto raggiunge il traguardo ha vinto:**
 - E' analoga a “**Se** succede questo **allora** deve succedere quello”
- **E' l'esecuzione condizionale di una azione: quando (o se) una condizione si verifica, una azione avviene**
 - **Nel nostro caso:**
 - **Quando:**
 - [indica una azione condizionale]
 - **<auto raggiunge il traguardo>**
 - [e' la nostra condizione]
 - **Allora ha vinto**
 - [e' l'azione che si deve svolgere quando la condizione e' vera]
 - [l'azione e' complessa, ma poco male]

Condizioni (Controllo)

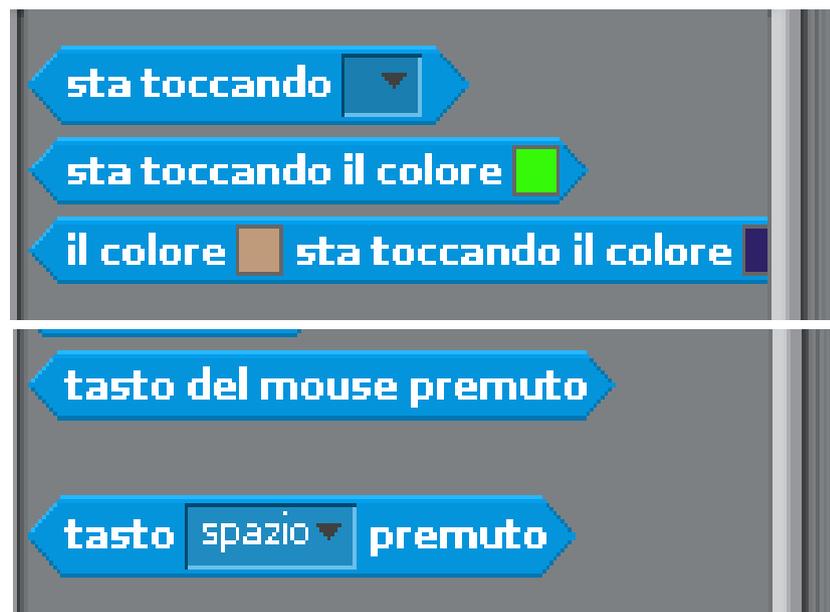


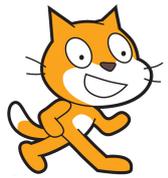
- **Ci sono diversi costrutti per azioni condizionali**
 - Si puo' scegliere quello piu' adatto alla specifica esigenza.
 - Di solito una buona analisi, cioe' scrivere in buon italiano quello che si vuole fare, rende naturale la scelta.
 - Per noi:
 - Ripeti l'azione di avanzare finche' <raggiungi il traguardo>

Condizioni (sensori)



- **Tutte le azioni condizionali controllano che una condizione sia vera o falsa**
 - **E' l'elemento <condizione>**
 - **In (sensori) ce ne sono diversi.**
- **Nel nostro caso, la condizione e' che uno sprite tocchi il traguardo**
 - Se disegniamo il traguardo di un certo colore (es: rosso), allora **<sta toccando il colore [rosso]>** e' quello che fa per noi.
 - Siccome questa azione la metteremo nel programma di uno sprite, il soggetto della frase <sta ...> e' il nostro sprite.
 - La stessa condizione nel programma del secondo sprite si riferira' al secondo sprite.

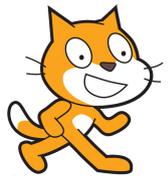




- **Elementi:**

- Mi serve uno sfondo adeguato: posso disegnarlo da zero oppure partire da uno sfondo esistente e modificarlo (**sfondo/stage**)
- Mi servono due sprite, come prima, che faranno entrambi (quasi) la stessa cosa (**Copiare blocchi di comandi**)
 - Posizionamento iniziale (**posizione**)
 - Partenza simultanea (**bandiera verde per partire**)
 - Avanzamento (**Movimento**)
 - Inizialmente passi tutti uguali
 - Dopo ogni passo devo controllare di aver raggiunto il traguardo (**condizione**)
 - La prima auto che arriva al traguardo ferma il programma (**ferma programma**)

Numeri casuali



- **Devo muovermi di un numero di passi casuale, altrimenti la corsa non e' interessante.**

- Dovrei lanciare un dado e sulla base del risultato del dado muovere l'auto di 1,2,...,6 passi

- SCRATCH ha un “dado” in (operatori)

A Scratch code block for generating a random number. It is a green rounded rectangle with a white border. The text inside is 'numero a caso tra' followed by a white box containing the number '1', the word 'e', and another white box containing the number '10'.

- Restituisce un numero casuale (pseudo-) tra 1 e 10 o tra N e M a nostra scelta

- Notate la forma del mattoncino (numero casuale ...)

- E' la forma che posso inserire in

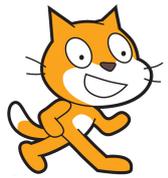
A blue Scratch code block for moving a sprite. It is a blue rounded rectangle with a white border. The text inside is 'fai' followed by a white box containing the number '10' and the word 'passi'.

A Scratch code block for moving a sprite with a random number of steps. It is a blue rounded rectangle with a white border. The text inside is 'fai' followed by a green rounded rectangle containing 'numero a caso tra', a white box with '1', 'e', a white box with '10', and 'passi'.



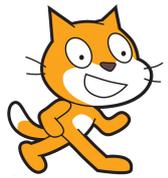
- E' una variabile
 - Una variabile e' un numero che puo' variare a seconda di determinate condizioni.
 - Fai [10] passi:
 - [10] e' una costante: ho scritto 10 e non cambiera' finche' non lo cambiero' a mano.
 - Fai [N] passi:
 - Il risultato dipende da quanto vale N. E N potrebbe cambiare nel corso del programma.
 - Nel nosto caso, [numero casuale] e' evidentemente una quantita' che puo' cambiare ogni volta che eseguo quella istruzione.

Es10: opzionale



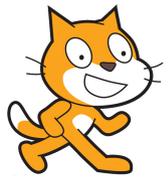
- **Ripetiamo tutta la gara per [N-dispari] volte, tenendo conto di quante volte vince la prima auto e quante volte la seconda.**
 - **Serve aver capito bene come funzionano le variabili**
 - **Suggerimento: creare due variabili, una per ciascuna auto, per tenere conto del numero di vittorie.**
- **Facciamo una scommessa prima della gara.**
 - **Ci serve un elemento di **Sensori****
 - **La risposta e' salvata nella variabile (risposta)**
 - **Se spunto il radio box [] allora il valore della variabile viene scritto sullo stage.**
 - **Dovro' usare quella variabile per vedere se ho vinto o no alla fine della gara.**





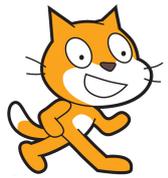
- **Invece di muovere le due macchine un un numero casuale di passi, facciamo una sfida tra due persone**
 - **L'auto 1 si muove in avanti ogni volta che viene premuto un tasto**
 - **L'auto 2 quando viene premuto un altro tasto**
- **Per il resto come prima:**
 - **l'auto che arriva prima al traguardo vince**
 - **Poi si puo' fare una sfida con N gare**
 - **E' da capire quanti passi far fare alle auto ogni volta che si preme un tasto: provate un po'**

Es11: Il gatto che vola



- **Vorrei che il mio gatto volasse tipo superman, sfrecciando tra le nuvole e i grattacieli**
 - Si sposta in alto e in basso comandato dalle frecce
 - E volando raccoglie i palloncini che incontra
- **Se sposto il gatto, presto finisce fuori dello schermo**
 - Invece sposto lo sfondo dietro al gatto nella direzione opposta, in modo da dare la sensazione del volo
 - **Creo un grattacielo sul bordo destro che si sposta verso sinistra**
 - Quando il grattacielo esce dallo schermo a sinistra, lo faccio ricomparire a destra (con un nuovo costume, così sembra diverso) e ricomincio
 - **Attenzione: il gatto deve essere in primo piano**

Gatto che vola

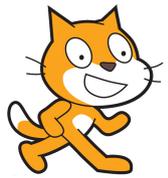


- **Oltre al grattacielo posso aggiungere anche un albero e qualche nuvola, che magari viaggia a velocità diversa**
- **Poi, nello stesso modo, metto dei palloncini**
 - **Posizione y casuale, che viaggiano verso sx**
- **Vogli poter comandare il mio gatto volante**
 - **Se premo freccia su, si alza, e viceversa**
- **Se gatto tocca palloncino (condizione), punteggio (variabile) aumenta di 1**
-

Es12: disegnare con una matita



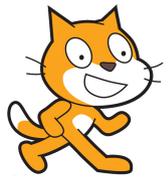
- **Su uno sfondo che mostra gli assi cartesiani, voglio disegnare usando una matita comandata dal mouse.**
- **Analisi:**
 - Sfondo con assi (e' disponibile tra gli sfondi standard)
 - Matita (**sprite**)
 - Da scegliere tra quelle disponibili
 - La matita si deve muovere con il mouse
 - Posizione segue mouse
 - La matita deve scrivere **quando** premo il bottone del mouse (**condizione**)
 - Per scrivere, uso la libreria delle azioni [**penna**]
- **Opzionale:**
 - **Aggiungiamo la possibilita' di cancellare tutto se premo spazio**
 - **Azione condizionale**



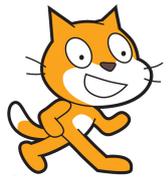
- **Posso usare gli sprite per scrivere sullo sfondo**
 - **Penna giu'/su** per cominciare (finire) di scrivere
 - Posso scegliere il **colore**, la **luminosita'** e la **dimensione** della penna
 - La punta della penna e' nel centro dello sprite
 - **Cosi' come definito dallo sprite**
 - Si puo' vedere/cambiare usando l'editor di immagini dello sprite stesso (costumi/modifica)
 - Posso **pulire** quello che ho scritto
 - **Timbra** copia l'immagine dello sprite sullo sfondo, come se fosse un timbro.



Es13: il campo fiorito

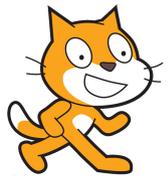


- **Ho un prato verde. Se faccio click con il mouse su un punto, un fiore appare in quel punto.**
- **Analisi:**
 - (e' piu' complicato di quanto puo' sembrare ad un primo esame)
 - Mi serve uno sfondo con il prato verde (**sfondo**)
 - All'inizio e' bene pulire il prato da tutti i fiori
 - **Quando** clicco sullo sfondo **allora** appare un fiore (**condizione**)
 -

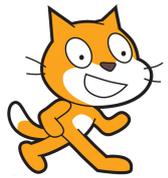


- **Fiore:**
 - E' uno sprite
 - Lo sprite deve apparire a comando
 - Lo sprite deve essere posizionato dove e' il mouse (**movimento, posizione, interazione con il mouse**)
 - Voglio molti fiori, non solo uno
 - Invece di fare molti sprite (quanti?) preferisco che quando appare lo sprite la sua immagine venga **stampata** sullo sfondo (**Penna→Timbra**)

Es13: Analisi (continua)

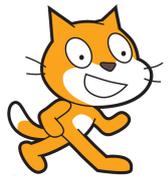


- **Quando** clicco sullo sfondo **allora** appare un fiore (**condizione**)
- Quando clicco su uno sprite (lo sfondo) un altro sprite deve fare qualcosa (timbrare)
 - Gli sprite si devono parlare (**messaggi**)
 - Quando (condizione) clicco su sfondo
 - **Sfondo manda messaggio**
 - Fiore e' in ascolto del messaggio
 - Quando riceve il messaggio fa la sua azione
 - Si mette dove e' il mouse
 - Timbra



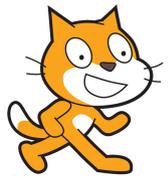
- Potrei avere diversi tipi di fiori (sprite) e selezionare quello che voglio piantare con un numero
 - Gli sprite iniziano il loro programma non quando viene premuto il tasto verde di inizio ma quando viene premuto il numero corrispondente

Es14: nascondino



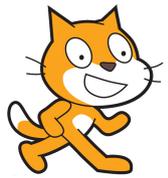
- **Facciamo una specie di nascondino.**
 - Su uno sfondo adeguato, ci sono una (o piu') figure che compaiono per breve tempo in un posto casuale.
 - Io devo cliccarci con il mouse prima che spariscano di nuovo
- **Analisi:**
 - Devo avere una o piu' figure che:
 - Vengono posizionate in un posto a caso
 - Compaiono sulla scena solo per un periodo limitato e poi spariscono
 - Mostra/nascondi
 - Se ci clicco con il mouse, allora il mio punteggio aumenta
 - Interazione con il mouse
 - Se mouse tocca figurina → figura dice “mi hai trovato” e punteggio aumenta
 - Punteggio e' una variabile

Nascondino passo passo



- Scegliamo uno sfondo adeguato (es foresta)
- Scegliamo una figurina
 - (se prendiamo un uccello, facciamolo volare!)
- La figurina scompare,
 - cambia posizione,
 - riappare per 1 secondo
- E poi ricomincia (iterazione)
 - Se io riesco a cliccare sulla figurina quando appare, dice "mi hai trovato!"
- Invece di apparire e scomparire a tempi fissi, facciamolo a tempi casuali
- La figurina appare in posizione casuale, altrimenti e' troppo facile!
- Contiamo quante volte riesco a trovare la figurina
 - Serve una variabile (punteggio) che deve essere aumentata di uno ogni volta che trovo la figurina
 - Per mostrare una variabile fare click sul quadratino vicino
- Potresti aggiungere diverse figurine
- Potrebbe essere una gara a tempo
 - Usa cronometro

Es15: raccogli le mele



- **E' stagione di mele, e cadono dagli alberi!**
 - **Noi dobbiamo raccoglierne il piu' possibile con il nostro cestino**
 - **(oppure palle da baseball in un campo: siate creativi!)**
- **Analisi:**
 - **Abbiamo delle figure (mele/palloni) che cadono dall'alto**
 - **All'inizio compaiono in alto, e scendono fino in basso [movimento]**
 - **Meglio se compaiono in un posto casuale, per rendere il gioco piu' interessante**
 - **Quando arrivano in basso, spariscono e ricominciano**
 - **Il cestino e' un'altra figurina:**
 - **E' in basso**
 - **Si muove a destra e sinistra comandata da noi**
 - **Per esempio con le frecce ← → oppure seguendo il mouse**
 - **Ma deve rimanere sempre in basso!**
 - **Se una mela cade nel cestino allora la mela sparisce e il punteggio aumenta**
 - **Interazione tra sprite**
 - **Punteggio e' una variabile**

Raccogli le mele passo passo



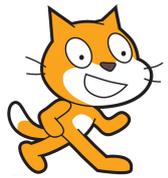
- **Scegliamo o creiamo uno sfondo**
 - **Scegliamo la frutta da far cadere**
 - **Un frutto compare in alto e cade**
 - Cadere vuol dire muoversi verso il basso
 - Potremmo far cadere gli oggetti con velocità diverse (casuali)
 - **Quando arriva a terra scompare e ne cade un altro**
 - **Magari ne facciamo cadere più di uno contemporaneamente**
 - **Facciamo una gara a tempo**
 - **Mi serve il cronometro**
 - Deve essere azzerato quando il gioco parte
 - **Alla fine contiamo quanti frutti abbiamo raccolto.**
 - Quando il cronometro arriva a tempo massimo (condizione)
 - Si ferma tutto e scrivo il punteggio finale

Es16: Un sub a pesca

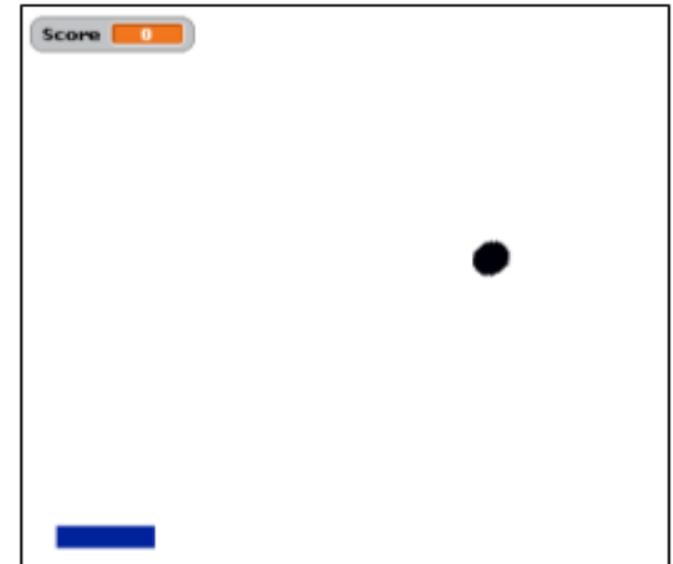


- **Un sub e' a pesca, e deve prendere 10 pesciolini nel tempo piu' breve possibile senza farsi mordere da un granchio che va su e giu'**
 - (Da un'idea di Michele Maffucci)
- **Ambiente marino, un sub, pesciolino e un granchio**
 - Sub insegue pesciolino, comandato da frecce tastiera o dal mouse
 - Quando lo prende lo dice e conta
 - Il pesciolino nuota qua e la in modo casuale, rimbalza sulle pareti
 - Quando viene preso scompare e riappare in un altro posto a caso
 - Anche il granchio si muove, ma sempre verso il sub
 - Se lo tocca lo morde (e conto i morsi)
 - Quando il tempo e' scaduto conto i pesciolini e i morsi
 - Oppure quando sono stato morso 3 volte il gioco finisce
 - Aggiungiamo opportune scritte e suoni
 - Potrebbe essere utile ridurre la dimensione delle figure

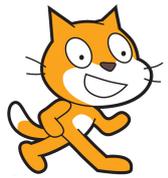
Es17: Pong



- **Una pallina si muove sullo schermo, rimbalzando sulle pareti.**
 - in basso un giocatore (barretta blu) viene mosso con il mouse.
 - Se la barretta intercetta la pallina, questa rimbalza
 - Il punteggio aumenta di 1
 - La pallina rimbalza con un angolo casuale tra -45 e +45 gradi verso l'alto
 - Se la pallina tocca il bordo inferiore, perdo una vita.
 - Se perdo 3 vite, il gioco e' finito
 - Dopo N colpi, la velocita' della pallina aumenta

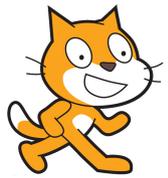


Es17: Analisi



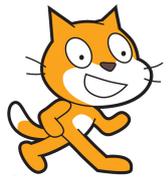
- **Movimento della pallina**
 - Rimbalzare sulle pareti
 - Se tocca il bordo inferiore allora gioco finito (**azione condizionale**)
 - Ricordatevi di come ci siamo accorti che le auto della corsa arrivavano al traguardo!
- **Movimento della barretta**
 - Movimento solo orizzontale!
 - La posizione lungo x deve essere quella del puntatore del mouse
- **Punteggio**
 - Inizialmente e' 0 (**variabile da inizializzare**)
 - Aumenta di 1 ogni volta che la barretta colpisce la pallina (**condizione, variable che cambia valore**)

Es18: Tris!



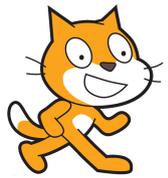
- **Programmate una scacchiera di Tris tra due persone. Le due persone giocano a turno su uno schema standard, mettendo le solite X e O nelle nove caselle con un click del mouse.**
- **Analisi:**
 - **Mi serve una scacchiera con lo schema vuoto**
 - **Devo far alternare i due giocatori, prima uno e poi l'altro**
 - **Segnalando a chi tocca giocare**
 - **Quando un giocatore fa click su una casella vuota, in quella casella compare il suo simbolo (X/O)**
 - **Se invece la casella e' gia' occupata non succede nulla**
- **Ha alcuni elementi interessanti!**

Es18: analisi



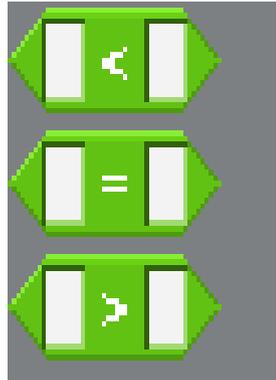
- **Mi serve una scacchiera (sfondo)**
- **Mi servono le pedine**
 - La pedina puo' essere una X o un O o nessuna **(costume!)**
 - Quando click del mouse su una pedina “vuota”, la pedina deve diventare X o O a seconda di chi sta giocando
 - **Mi concentro su UNA pedina, poi la copiero' nove volte nelle varie caselle!**
 - **Se** il turno e' di X allora diventa X, **altrimenti** diventa O **(azione condizionale!)**
- **Devo alternare I giocatori, un turno per volta**
 - Ogni volta che uno gioca, il turno passa all'altro.

Condizioni su variabili



- **Devo fare una certa azione quando la mia variabile ha un certo valore**

- **Se e' il turno di X diventa X, altrimenti 0 (condizione!)**
- **Mi serve una condizione che dipende dalla variabile**
- Da libreria **Operatori**
 - **Se (turno) = X e' vera, altrimenti e' falsa**
 -

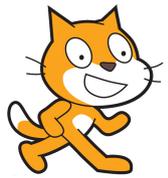


- **Devo ricordarmi di cambiare turno!**

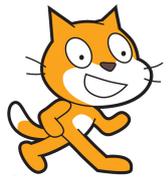
- Dovrei anche controllare di non aver gia' giocato su quella casella, cioe' che il costume sia null!
- **Sugg: condizione su numero costume**



Es18: le pedine e il turno

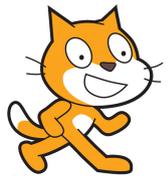


- **Metto una pedina in ogni casella.**
 - La pedina e' uno sprite che ha tre **costumi**
 - Vuoto (bianco) - X - O
 - Quando faccio click su una pedina, cambio costume (**Interazione con il mouse e cambiare aspetto**)
 - Se e' il turno di X diventa X, altrimenti O (**condizione!**)
- **Turno:**
 - All'inizio e' il turno di un giocatore.
 - Quando lui gioca, il turno passa all'altro giocatore.
 - Uso una variabile che rappresenta a chi tocca (es turno=1 o 2)
 - E' una variabile nuova:
 - devo crearla,
 - inizializzarla (decidere di chi e' il primo turno)
 - modificarla (ogni volta che cambia il turno)



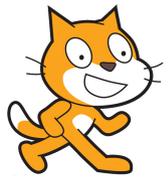
- **Quando tutte le caselle sono piene, lo sfondo cambia.**
- **Analisi:**
 - Devo accorgermi che tutte le caselle sono “piene”
 - Per esempio, conto quanti turni sono passati
 - Quando arrivo a 9, allora le caselle sono tutte piene
 - Aggiungo una variabile che conta I turni giocati
 - All'inizio e' 0 e aumenta di 1 ogni volta che viene fatta una mossa valida
 - Altre soluzioni sono possibili!
- **Sfida (tosta): siamo capaci di capire chi ha vinto, o se nessuno ha vinto?**
 - Devo analizzare bene cosa vuol dire vincere
 - tre elementi uguali allineati in orizzontale, verticale o obliquo.
 - Lo lascio come (non) facile esercizio per lo studente!

Es19: la pallina che cade e rimbalza



- **Una pallina cade come se fosse soggetta alla gravita', e rimbalza sul pavimento**
- **Analisi:**
 - **Esercizio di fisica:**
 - **serve conoscere il moto uniformemente accelerato**
 - **Una pallina (sprite) all'inizio la mettiamo in alto a sinistra sullo sfondo**
 - **Si sposta (muove N passi) dentro un ciclo (il tempo)**
 - Ogni iterazione, si sposta di X in orizzontale e Y in verticale
 - I valore di X e Y dipendono dalla velocita' lungo X e Y
 - **Si muove secondo le leggi della gravita'**
 - La velocita' lungo x (orizzontale) rimane costante
 - La velocita' lungo y (verticale) all'inizio e' 0
 - Dopo ogni passo, la V_y aumenta di una quantita' costante (accelerazione di gravita')
 - **Mi muovo in modo indipendente lungo X e lungo Y**

Es19: segue



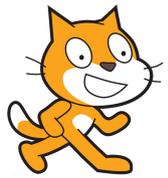
- Partiamo da un moto non accelerato, cioè senza la gravità.
 - La velocità è costante sia in x che in y
- Faccio due cicli, uno per x e uno per y. Ad ogni iterazione:
 - $X' = X + v_x$
 - Cambia X di v_x
 - $Y' = Y + v_y$
 - Cambia Y di v_y
- Definisco due **variabili** v_x e v_y e gli **assegno** dei valori iniziali.
- Per “vedere” il movimento, stampo **[timbro]** la posizione della pallina ad ogni iterazione
 - come se fosse una foto stroboscopica.

Es19: segue



- **Facciamo rimbalzare la pallina sul pavimento:**
 - Disegno sullo sfondo il pavimento di un dato colore
 - **Quando** la pallina **tocca il colore** del pavimento
 - **Rimbalza: cosa significa?**
 - Rimbalzare vuol dire che la sua velocita' lungo y viene invertita: $v'_y = -v_y$ (**assegnare variabile a valore, operatore su variabile: $-v_y = 0 - v_y$**)
 - Mentre quella lungo x rimane la stessa
 - **Se voglio anche emulare la parte non elastica dell'urto, allora la velocita' dopo l'urto sara' un po' piu' piccola di quella prima dell'urto,**
 - oltre che cambiata di segno

Es19: segue



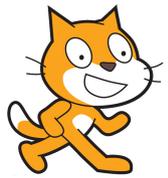
- **Mettiamo la gravita':**
 - La gravita' fa aumentare la velocita' lungo y di una costante (l'accelerazione di gravita' g) verso il basso ad ogni iterazione
 - $v'_y = v_y - g$
 - Mi serve una terza iterazione che fa semplicemente questo.
- **Metto tutto insieme, e ho una pallina che cade sotto gravita' e rimbalza!**

Es20: segui la linea rossa



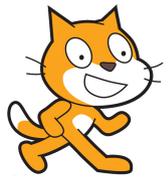
- **Disegno una linea rossa sullo sfondo. Lo sprite si muove seguendo la linea che ho disegnato**
- **Analisi:**
 - Disegnare una linea rossa (**facile**)
 - Segui la linea
 - Questo non e' per niente facile, occorre una buona analisi prima di partire.
 - Se foste voi a dover seguire una linea disegnata per terra, come fareste?
 - Provate a analizzare le vostre azioni in modo da capire l'algoritmo che usate per un tale compito
 - Tenete presente che alcune azioni che per voi sono possibili (es guardare avanti e vedere in che direzione va la linea) non sono possibili per uno sprite.
 - Provate a pensare se potete emulare questa azione (guardare avanti) usando quello che lo sprite sa fare (muoversi, condizioni, ...)

Es21: nevicata!



- **Simuliamo una nevicata.**
 - **I fiocchi scendono**
 - **oscillando a destra e sinistra mentre scendono.**
 - **Compaiono in posizione casuale in alto**
 - **Movimento verso il basso**
 - **Movimento casuale dx/sx**
 - **Quando arrivano sul terreno,**
 - **si accumulano su di esso o sulla neve che c'era prima.**
 - **Condizione: fiocco tocca il terreno o altro sprite**

22 Labirinto

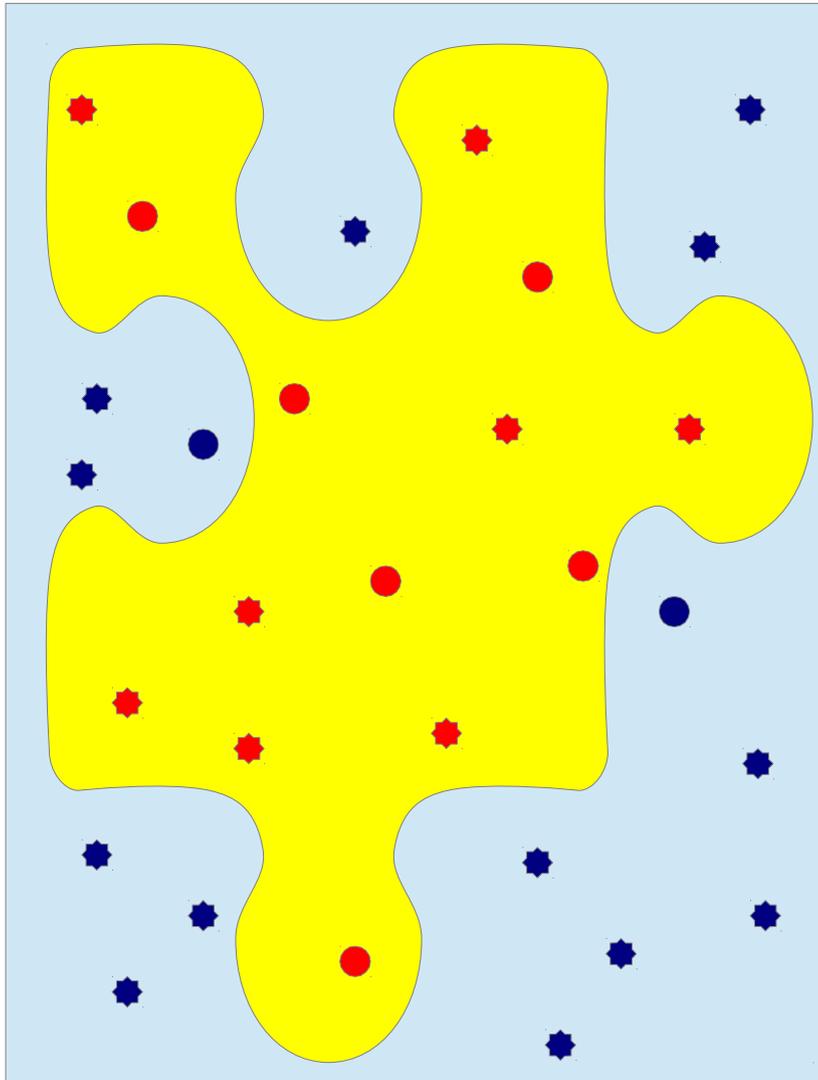
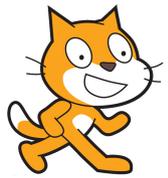


- **Un labirinto: un personaggio parte da un punto e deve arrivare ad un traguardo, senza attraversare le pareti del labirinto.**
 - **Il movimento del mio personaggio e' gestito o dalle frecce o dal mouse**
 - **Mettiamo anche un cronometro per vedere quanto ci mette ad arrivare al traguardo**
 - **Se tocca le pareti, torna alla partenza**
 - **Magari mettiamo piu' di un labirinto tra cui scegliere all'inizio del gioco**
 - **Ci sono moltissimi esempi nella pagina di scratch**

23 Misurare le aree

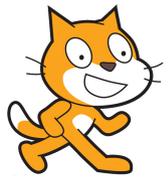


- **Problema: data una superficie di forma qualsiasi (non regolare) stimare la sua area A.**
 - **Metodo MonteCarlo**
 - Disegno la superficie per terra
 - La racchiudo dentro una superficie S piu' grande di area nota (es un rettangolo, o un cerchio o quello che voglio)
 - Lancio un buon numero di sassi dentro la superficie piu' grande.
 - Conto quanti di questi sassi sono dentro la superficie di cui voglio calcolare l'area
 - L'area delle due superfici sara' proporzionale al numero di sassi che sono dentro ciascuna.
 - $A/S = \#sassi \text{ in } A / \#sassi \text{ in } S$
 - $A = S * (\#sassi \text{ in } A / \#sassi \text{ in } S)$



- **Voglio calcolare l'area A della superficie gialla.**
 - La racchiudo dentro il rettangolo azzurro
 - Che ha area nota $S=(base*altezza)$
 - Lancio (tanti) sassi in modo casuale
 - Ho lanciato 30 sassi
 - 14 sono dentro la superficie gialla
 - $A=S*14/30 \sim 1/2*S$
- **E' un metodo approssimato, tanto piu' preciso quanti piu' sassi uso!**
 - Permette di calcolare l'area di una superficie generica.
- **Implementare questo metodo in scratch e' abbastanza facile**
 - Numeri casuali per lanciare I sassi
 - Input di quanti sassi lanciare
 - Variabile (#sassi)
 - Variabile (#sassi dentro)
 - Occorre capire quando un sasso e' dentro o e' fuori
 - Area nota e' lo sfondo (sappiamo base e altezza, in pixel)

Es24 Battaglia spaziale



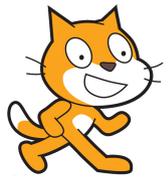
- **Una astronave vola tra le stelle, incontra una astronave aliena e cerca di colpirla sparando**
 - **Questa volta facciamo “volare” la nostra astronave verso l’alto**
 - **Usiamo il trucco di far scorrere le stelle verso il basso!**
 - **Comandiamo la posizione della nostra astronave con le frecce o il mouse**
 - **Attenzione: solo dx/sx, non anche su e giu’**
 - **L’altra astronave compare in alto e scende**
 - **Ma fa zig-zag per evitare I nostri colpi!**
 - **Se premiamo un tasto, parte un missile verso l’alto**
 - **Se il missile colpisce l’astronave aliena, quella esplode**
 - **Se l’astronave ci scontriamo esplodiamo tutti e due!**
 - **Facciamo sparare anche gli alieni?**

Es25 Space Invaders



- **Lo conoscete?**
 - **Gli alieni si spostano dx e sx**
 - **Es quando arrivano al bordo**
 - **Scendono di un livello**
 - **Noi comandiamo la difesa**
 - **Spariamo per distruggerli**
- **Per semplificare**
 - **niente bastioni di difesa**
 - **non cosi' tanti alieni**





- **Un rally con un'auto.**
 - Devo completare il percorso senza andare a fuori strada
- **Percorso:**
 - Ho bisogno di una strada, di un bordo strada (per capire quando l'auto esce di strada) e un traguardo (per capire quando l'auto arriva in fondo)
- **Auto**
 - Guida: uso le frecce per girare dx/sx e per accelerare/frenare
 - Uso una variabile (speed) che viene aumentata/diminuita se premo su o giu'
 - Uso quella variabile per decide quanti passi avanti fare
 - La direzione e' comandata da frecce dx/sx
 - Se esco di strada, la velocita' diventa 0
- **Gara a tempo?**