associazione Dschola
LE SCUOLE PER LE SCUOLE

Fondazione CRT

Progetto DIDEROT

Programmo Anch'io

Guida all'uso di SCRATCH

Versione Studenti

Iniziativa realizzata nell'ambito del progetto Diderot della Fondazione CRT

Introduzione

L'Associazione Dschola, ha già sperimentato positivamente la validità del linguaggio Scratch a supporto della didattica, organizzando sia seminari di formazione per i propri associati sia diverse edizioni dell'Italian Scratch Day, con l'obiettivo di condividere buone pratiche e valorizzare i migliori elaborati dei ragazzi provenienti da tutta Italia.

A seguito di tale esperienza ha elaborato il progetto "Programmo anch'io", che si inserisce nella linea denominata "Computing" di Diderot 2014-2015 ed ha lo scopo di diffondere il linguaggio Scratch, far esercitare gli studenti e far produrre agli stessi elaborati personali e/o di gruppo.

Questa guida vuole essere lo strumento necessario ed indispensabile rivolto agli studenti che, dopo aver acquisito le conoscenze necessarie sull'utilizzo del linguaggio di programmazione Scratch, potranno esercitarsi, con la guida ed il supporto dei formatori, per conseguire le abilità e le competenze necessarie per poter cimentarsi in maniera autonoma o di gruppo nella produzione di prodotti personalizzati, frutto della loro fantasia, estro e creatività.

Gli studenti potranno quindi avvicinarsi alla programmazione in modo piacevole e divertente, utilizzare la stessa per la costruzione di giochi o la narrazione di storie ed episodi nuovi o ricavati dallo studio di altre discipline, in modo tale da valorizzare e sviluppare la multidisciplinarità.

La serie di esercitazioni proposte, costituisce un valido pacchetto formativo, in modo tale da privilegiare la dimensione laboratoriale e per consentire agli studenti di esercitarsi nel miglior modo possibile, in sintonia con quanto sostenuto da Diderot con l'espressione: "*Giovane, più che istruirti, vorrei esercitarti*".

Il Presidente Dschola

Alfonso Lupo

LA FONDAZIONE CRT

La Fondazione CRT - Cassa di Risparmio di Torino è un ente privato non profit nato nel 1991. Da oltre 25 anni è uno dei "motori" dello sviluppo e della crescita del Piemonte e della Valle d'Aosta in tre macro-aree: Arte e Cultura, Ricerca e Istruzione, Welfare e Territorio. Interviene con progetti e risorse proprie per la valorizzazione dei beni artistici e delle attività culturali, la promozione della ricerca scientifica e della formazione dei giovani, il sostegno all'innovazione e all'imprenditoria sociale, l'assistenza alle persone in difficoltà, la salvaguardia dell'ambiente, il sistema di protezione civile e di primo intervento.

La sua attività si caratterizza per un'attenzione particolare all'internazionalizzazione, con il duplice obiettivo di rendere più forti le organizzazioni non profit locali attraverso l'apertura all'Europa e al mondo e, nello stesso tempo, di attrarre sul territorio nuove risorse progettuali ed economiche.

In un quarto di secolo di attività, la Fondazione CRT ha distribuito risorse per oltre 1 miliardo e 600 milioni di euro e consentito la realizzazione di più di 38.000 interventi per il territorio, sostenendoli non solo dal punto di vista delle erogazioni, ma anche delle progettualità: il tutto, ascoltando le esigenze del territorio stesso e delle realtà aggregative, istituzionali e del non profit. A questo tradizionale impegno erogativo si sono aggiunti importanti investimenti, come la riqualificazione delle OGR-Officine Grandi Riparazioni di Torino.

www.fondazionecrt.it

IL PROGETTO DIDEROT DELLA FONDAZIONE CRT

La Fondazione CRT realizza il Progetto DIDEROT per offrire agli studenti di tutti gli Istituti di istruzione primaria e secondaria di I e II grado del Piemonte e della Valle d'Aosta una duplice opportunità: avvicinarsi in modo creativo e stimolante a discipline non sempre inserite nei programmi curricolari e, nello stesso tempo, approfondire le materie tradizionali con metodologie innovative.

Il Progetto si articola in workshop, laboratori, video-lezioni, visite, seminari, incontri-dibattiti con esperti e testimonial, e perfino concerti e rappresentazioni teatrali, in ambiti quali l'arte e la matematica, l'economia e il computing, la tutela della salute e dell'ambiente, la filosofia. La partecipazione è gratuita per tutte le scuole (escluso il costo di eventuali trasporti).

Il progetto Diderot della Fondazione CRT ha coinvolto finora circa 886.000 studenti tra i 6 e i 20 anni.

La rete delle scuole Dschola opera sul territorio piemontese dal 2000.

Nata su iniziativa del CSP – Innovazione nelle ICT, Organismo di ricerca regionale riconosciuto dal MIUR, per favorire l'innovazione nelle scuole come parte del progetto ICT finanziato dalla Fondazione Cassa di Risparmio di Torino e successivamente dalla Regione Piemonte sino al 2005, la rete dei poli DSchola, denominati Centri di Servizio, Animazione e Sperimentazione (CSAS), è un'esperienza unica a livello nazionale, apprezzata a livello europeo e replicata in alcuni Paesi dell'America Latina.

La rete si è costituita in Associazione Dschola dal 18 ottobre 2004 e da allora è il punto di riferimento sia per l'Ufficio Scolastico Regionale sia per i diversi soggetti operanti in Piemonte interessati all'innovazione nella scuola.

L'Associazione ha operato in questi anni con un modello che ha favorito la condivisione dei saperi e la formazione fra pari: le competenze tecnologiche, le sperimentazioni didattiche e amministrative, sono messe a disposizione di tutte le scuole del territorio attraverso l'organizzazione di seminari e corsi di formazione gratuiti.

Per l'assistenza, consulenza e formazione si avvale di tecnici e docenti esperti, in possesso delle necessarie competenze ed appartenenti al mondo della scuola.

L'Associazione ha sperimentato negli anni soluzione tecnologiche scalabili ed efficaci per la connettività delle classi (modello wireless campus), per la gestione di laboratori informatici e navigazione protetta (Manutenzione zero e Asso Dschola), per l'ospitalità gratuita dei siti delle scuole (Share Dschola), per l'uso di netbook in classe (1 computer per ogni studente).

L'Associazione lavora anche per l'innovazione didattica attraverso lo sviluppo di comunità di pratiche sull'elearning, sulle mappe mentali e concettuali, sulla media education (Dschola TV), in collaborazione con reti e Associazione europee (Media and Learning Association), sull'introduzione della programmazione, promuovendo l'utilizzo del linguaggio Scratch dal 2012 anche attraverso l'organizzazione dell'Italian Scratch Festival.

Per ulteriori informazioni, visitare il sito www.associazionedschola.it.

Autori

Gli autori fanno parte del gruppo di Ricerca e Sviluppo dell'Associazione Dschola.

Prof. Alberto Barbero - I.I.S. Vallauri, Fossano (CN)

Docente di Informatica e VicePreside dell'I.I.S. "G.Vallauri" di Fossano. Autore di testi scolastici per la Pearson Italia. Collaboratore del Dipartimento di Informatica dell'Università di Torino. Responsabile del gruppo "Ricerca e sviluppo" dell'Associazione Dschola. Referente scolastico del CSAS e dell'Associazione Dschola. Docente in corsi di formazione P.A.S. per docenti, per l'A.U.S.E.R. sulle T.I.C. e per adolescenti per l'introduzione al *coding* con il linguaggio *Scratch*.

Prof. Marco Marchisotti - I.I.S. Sobrero, Casale Monferrato (AL)

Laureato in Informatica ed in Sistemi per il Trattamento dell'Informazione, docente di Informatica presso l'I.S. "Sobrero" di Casale Monferrato. Coordinatore del Dipartimento di Informatica e amministratore della rete locale. Referente scolastico del CSAS e dell'Associazione Dschola. Docente in corsi di formazione per adulti sulle T.I.C. e per adolescenti per l'introduzione alla programmazione con il linguaggio *Scratch*.

Prof. Mattia Davì - I.C. San Michele, Mondovì (CN)

Idr presso l'Istituto Comprensivo di San Michele M.vì (CN). Laureato in Scienze dell'Educazione, Laurea Magistrale in Scienze Religiose e con un Master in Didattica Assistita dalle Tecnologie. Funzione Strumentale per le TIC presso il proprio Istituto e consulente per reti di scuole. Docente in corsi di formazione per insegnanti in riferimento all'uso didattico delle TIC e per l'introduzione al *coding* con il linguaggio Scratch in vari ordini di scuola. Autore di manuali ECDL e presso la casa editrice Elledici.

NOTA

Tutti gli esercizi qui riportati sono disponibili nella galleria denominata "**Diderot 2014**" sul sito Scratch al seguente indirizzo:

http://scratch.mit.edu/studios/567494/

Sono stati svolti tutti con la versione di Scratch 1.4 per poterli applicare anche in scuole dove la dotazione hardware dei laboratori di informatica non permette di utilizzare la nuova versione 2.0 sia on-line che off-line.

Con la versione 2.0 alcuni esercizi potrebbero essere svolti anche con istruzioni leggermente diverse e che potrebbero permettere anche una ulteriore semplificazione della logica di risoluzione.

Scratch on-line: http://scratch.mit.edu/

Download Scratch 1.4 offline editor: http://scratch.mit.edu/scratch_1.4/

Download Scratch 2.0 offline editor: http://scratch.mit.edu/scratch2download/

Quest'opera è distribuita con licenza Creative Commons Attribuzione 3.0 Italia.

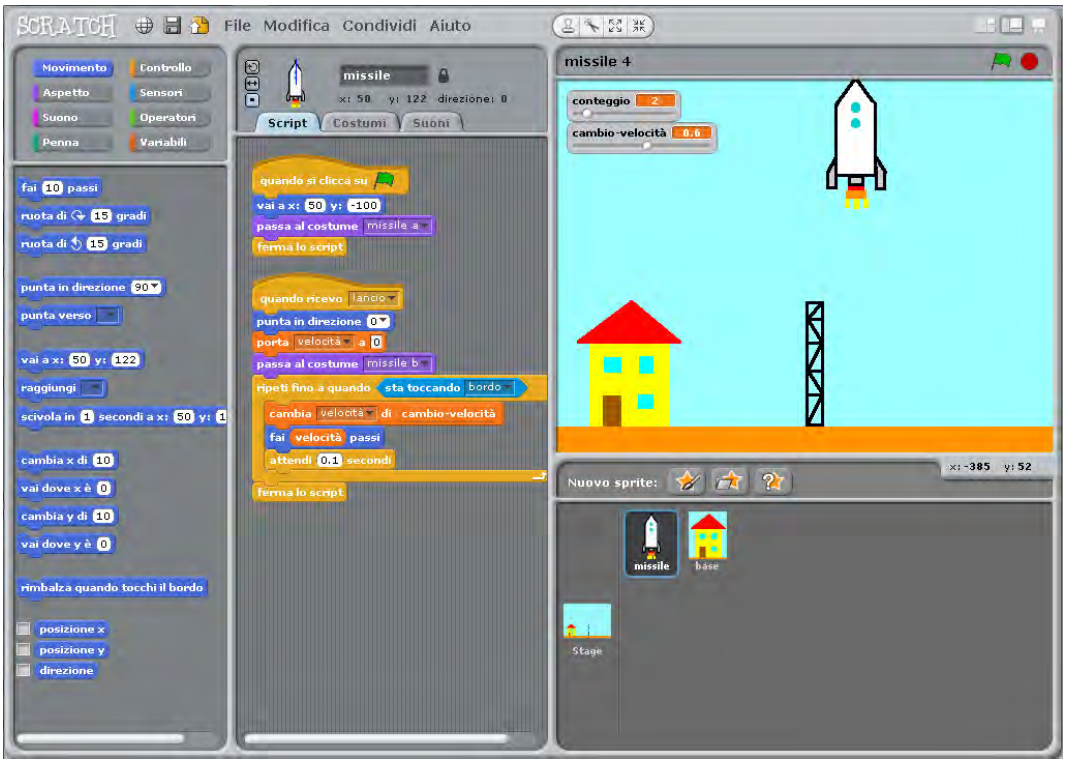
Scratch: la programmazione per gioco

Scratch è un ambiente grafico di lavoro per imparare lo sviluppo di programmi in modo semplice e creare storie interattive, giochi, animazioni grafiche, simulazioni e altro ancora, per poi condividere i programmi realizzati attraverso il Web.

La codifica dei programmi in Scratch consiste nell'impilare blocchi, che presentano forma e colore dipendenti dall'istruzione che si vuole utilizzare, come si fa con i mattoncini delle costruzioni, pezzo dopo pezzo. Così facendo, è possibile avvicinarsi alla programmazione e capirne la logica alla base in modo divertente e creativo.

I programmi di Scratch agiscono su oggetti grafici, disegni, immagini chiamati sprite, come la figurina del gatto che rappresenta l'icona di Scratch. E' possibile disegnare gli sprite a piacere attraverso un semplice programma di disegno, così come è possibile importare un'immagine o una foto scattata con una macchina fotografica digitale o con la webcam. Gli sprite, inoltre, possono essere personalizzati associando costumi diversi, in modo da animarli dando loro la forma che più interessa, e suoni diversi. A ogni sprite sono associati uno o più script, insiemi di blocchi-istruzioni che indicano allo sprite che cosa deve fare: parlare, muoversi, suonare, nascondersi, mostrarsi, eseguire calcoli e tanto altro ancora.

Scratch versione 1.4



Questa è la videata dell'ambiente di sviluppo di **Scratch 1.4**. Tale versione ha il vantaggio di essere semplice da installare, funziona egregiamente anche su PC con poche risorse a disposizione e non ha la necessità di essere connesso ad Internet. Il download può essere effettuato al link http://scratch.mit.edu/scratch2download.

Oltre alla barra dei menu in alto, dove si trovano i soliti comandi di apertura e salvataggio dei file, l'ambiente è suddiviso fondamentalmente in tre aree diverse:

- a destra della videata troviamo lo **stage**, il palcoscenico, l'area in cui gli sprite interagiscono durante l'esecuzione del programma, che parte di solito quando si fa clic sul pulsante con la bandierina verde, posto sopra lo stage; gli sprite sono dei disegni o delle immagini, presi tra quelli messi a disposizione da Scratch o creati da noi, con cui si interagisce quando è lanciato il programma e che eseguono i blocchi (le istruzioni) che costituiscono lo script (il programma);
- alla sinistra troviamo l'elenco dei **blocchi** che servono a codificare il programma, suddivisi in 8 categorie caratterizzate ciascuna da un diverso colore; sono in pratica i mattoncini che si utilizzano per costruire il programma impilandoli nella sequenza desiderata;
- al centro, infine, vi è l'area per programmare, detta **area Script**, in cui trasciniamo e impiliamo con la tecnica del drag and drop i blocchi che servono alla codifica del programma.

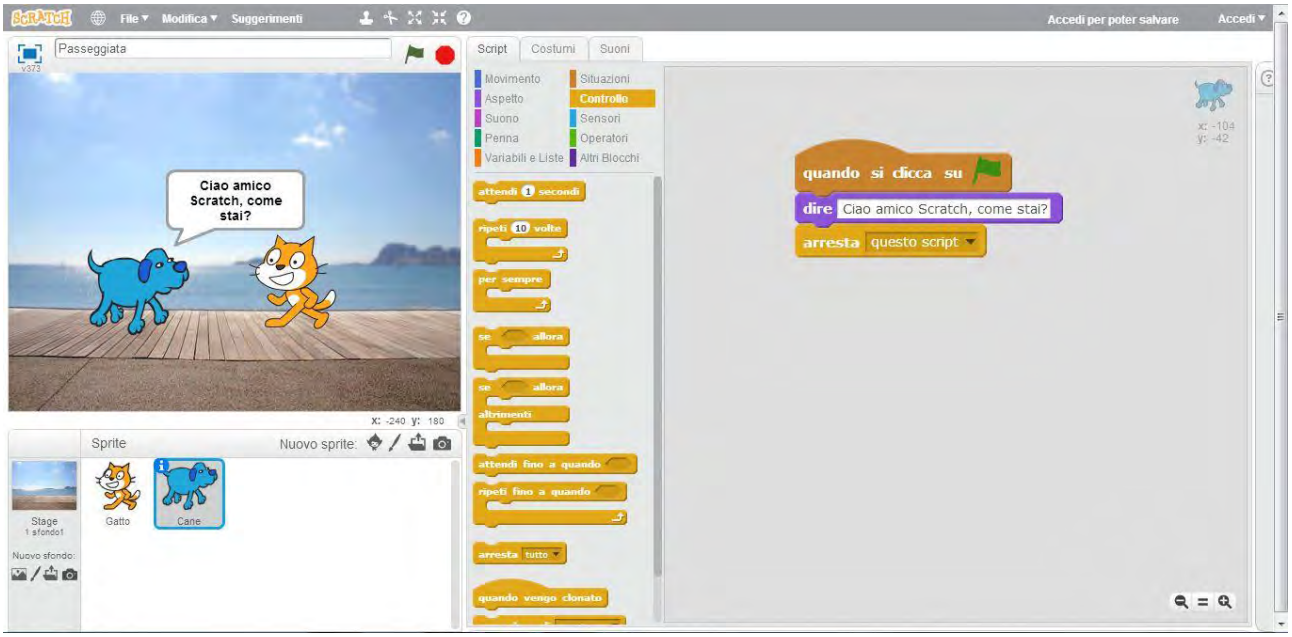
Lo **sprite** è l'oggetto posto nello stage a cui vengono associati uno o più script per farlo muovere, interagire, calcolare, giocare, e altro ancora. Oltre allo sprite raffigurante il gatto, esistono molti altri sprite già disponibili nel sito, oppure è possibile crearne dei nuovi attraverso un apposito **editor**.

Agendo su questi tre pulsanti, posti al di sotto dello stage, è possibile:

- scegliere uno sprite dalla libreria;
- disegnare un nuovo sprite;
- caricare uno sprite a sorpresa.

I **blocchi** sono suddivisi in 8 categorie di colore diverso in base al tipo di azione che effettuano. Nelle categorie **Movimento**, **Aspetto**, **Suono** si trovano i blocchi che permettono di applicare allo sprite le animazioni e i suoni desiderati; nella categoria **Penna** si trovano i blocchi per sviluppare programmi che permettono di scrivere e disegnare; i blocchi delle categorie **Controllo**, **Operatori**, **Variabili e Liste** permettono di codificare le principali strutture algoritmiche (se... allora... altrimenti, ripeti fino a quando..., ripeti... volte, ecc.) che agiscono sulle variabili: inoltre le stesse categorie hanno al loro interno i blocchi per la gestione degli eventi che servono a lanciare l'esecuzione degli script. Infine nella categoria **Sensori** troviamo i blocchi per testare il verificarsi di certe situazioni.

Scratch versione 2.0



Questa è la videata dell'ambiente di sviluppo di **Scratch 2.0** a cui si accede via web all'indirizzo http://scratch.mit.edu. Oltre alla barra dei menu in alto, dove si trovano i soliti comandi di apertura e salvataggio dei file, l'ambiente è suddiviso fondamentalmente in tre aree diverse:

- a sinistra della videata troviamo lo **stage**, il palcoscenico, l'area in cui gli sprite interagiscono durante l'esecuzione del programma, che parte di solito quando si fa clic sul pulsante con la bandierina verde, posto sopra lo stage; gli sprite sono dei disegni o delle immagini, presi tra quelli messi a disposizione da Scratch o creati da noi, con cui si interagisce quando è lanciato il programma e che eseguono i blocchi (le istruzioni) che costituiscono lo script (il programma);
- al centro troviamo l'elenco dei **blocchi** che servono a codificare il programma, suddivisi in 10 categorie caratterizzate ciascuna da un diverso colore; sono in pratica i mattoncini che si utilizzano per costruire il programma impilandoli nella sequenza desiderata;
- alla sinistra, infine, vi è l'area per programmare, detta **area Script**, in cui trasciniamo e impiliamo con la tecnica del drag and drop i blocchi che servono alla codifica del programma.

Lo **sprite** è l'oggetto posto nello stage e a cui vengono associati uno o più script per farlo muovere, interagire, calcolare, giocare, e altro ancora. Oltre allo sprite raffigurante il gatto, esistono molti altri sprite già disponibili nel sito, oppure è possibile crearne dei nuovi attraverso un apposito **editor**.

Agendo su questi quattro pulsanti, posti al di sotto dello stage, è possibile:

- scegliere uno sprite dalla libreria;
- disegnare un nuovo sprite;
- caricare uno sprite da file;
- creare un nuovo sprite dalla webcam.

I **blocchi** sono suddivisi in 10 categorie di colore diverso in base al tipo di azione che effettuano. Nelle categorie **Movimento**, **Aspetto**, **Suono** si trovano i blocchi che permettono di applicare allo sprite le animazioni e i suoni desiderati; nella categoria **Penna** si trovano i blocchi per sviluppare programmi che permettono di scrivere e disegnare; i blocchi delle categorie **Controllo**, **Operatori**, **Variabili e Liste** permettono di codificare le principali strutture algoritmiche (se... allora... altrimenti, ripeti fini a quando..., ripeti... volte, ecc.) che agiscono sulle variabili; la categoria **Situazioni** contiene i blocchi per la gestione degli eventi che lanciano l'esecuzione degli script; nella categoria **Sensori** troviamo i blocchi per testare il verificarsi di certe situazioni; infine, nella categoria **Altri Blocchi** troviamo raccolti i nuovi blocchi che possiamo creare associando loro uno script in modo da gestire i sottoprogrammi.

Per chi non volesse lavorare on line, esiste anche la versione di Scratch 2.0 scaricabile per essere utilizzata **offline**, sia per Mac, Windows e alcune versioni di Linux. Il download può essere effettuato al link http://scratch.mit.edu/scratch2download. Questa versione potrà essere aggiornata man mano che verranno aggiunte nuove funzionalità, attraverso il comando *Controlla gli aggiornamenti* contenuto nella barra dei menu sotto la voce *File*.

Acquario

Prerequisiti

Saper disegnare con il computer (utilizzo di Paint): Scratch ha al suo interno uno strumento per disegnare simile a Paint.

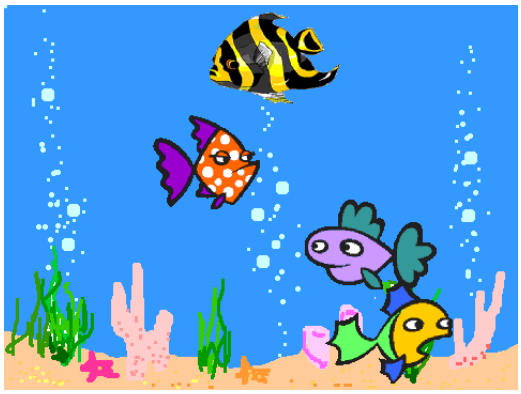
Esercizio

Progettare un acquario con alcuni pesci che si muovono avanti ed indietro.

Concetti di programmazione veicolati

L'iterazione infinita.

Soluzione



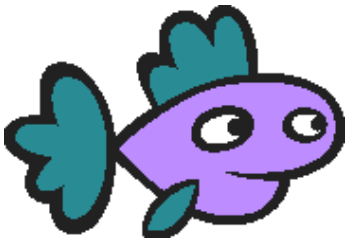
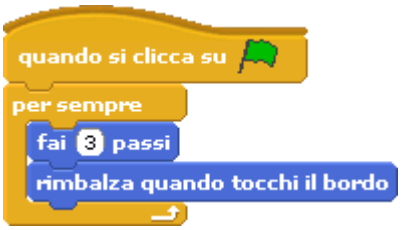
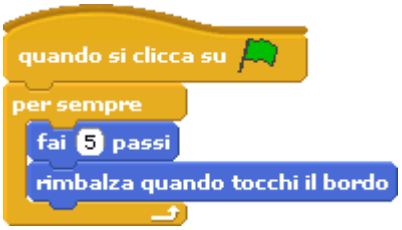
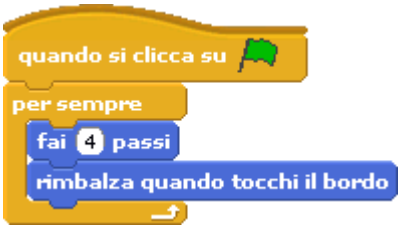
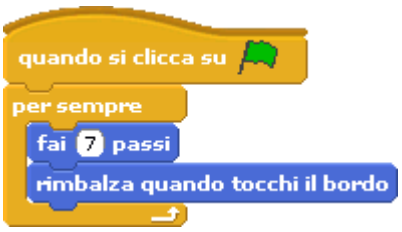
È importante far capire come si può cambiare il comportamento degli sprite in base ai tre pulsanti rispettivamente:

- può ruotare,
- voltati solo a destra e a sinistra (opzione da selezionare in questo progetto)
- non ruotare

Per inserire uno sfondo al progetto occorre selezionare

e poi dalla scheda Sfondi selezionare lo sfondo desiderato importandolo dalle immagini nella libreria già a disposizione o da una immagine desiderata.

Gli script da progettare sono i seguenti:

Sprite	Codice
	
	
	
	

Attività da far svolgere in autonomia

Fare disegnare un pesce con lo strumento di disegno interno di Scratch.

Fare cercare il disegno di un pesce in Internet, scaricarlo, importarlo in Scratch e modificarlo opportunamente con lo strumento di disegno incorporato. Ad esempio eliminare lo sfondo e ruotarlo orizzontalmente.

Esercizio svolto

http://scratch.mit.edu/projects/20099140/

Conversazione

Prerequisiti

Saper distinguere le categorie a cui appartengono i blocchi.

Esercizio

Progettare una semplice conversazione tra uno sprite e l'utente.

Concetti di programmazione veicolati

Gestione dell'input/output; istruzione condizionale.

Soluzione

Per fornire in ingresso un valore viene visualizzata una casella di testo in cui scrivere, attraverso la tastiera, l'input desiderato che verrà salvato temporaneamente in risposta.

Attraverso una semplice istruzione di controllo è possibile personalizzare la risposta del gatto in modo da tenere conto del valore dell'età inserito.

Gli script da progettare sono i seguenti:

Sprite	Codice

Attività da far svolgere in autonomia

Far proseguire la conversazione inserendo nuove istruzioni condizionali per permettere di personalizzare ulteriormente le risposte in base agli input forniti.

Cambiare sprite e gestire la conversazione verso alcuni dei temi che si stanno studiando a scuola (ad esempio cercare il disegno di un Napoleone in Internet, scaricarlo, importarlo in Scratch e modificarlo opportunamente con lo strumento di disegno incorporato. La conversazione a questo punto potrà vertere sula vita del personaggio storico).

Esercizio svolto

http://scratch.mit.edu/projects/28969792/

Quiz con punteggio

Prerequisiti

Conoscere l'istruzione condizionale se… allora… altrimenti…

Esercizio

Progettare un quiz con il calcolo del punteggio realizzato.

Concetti di programmazione veicolati

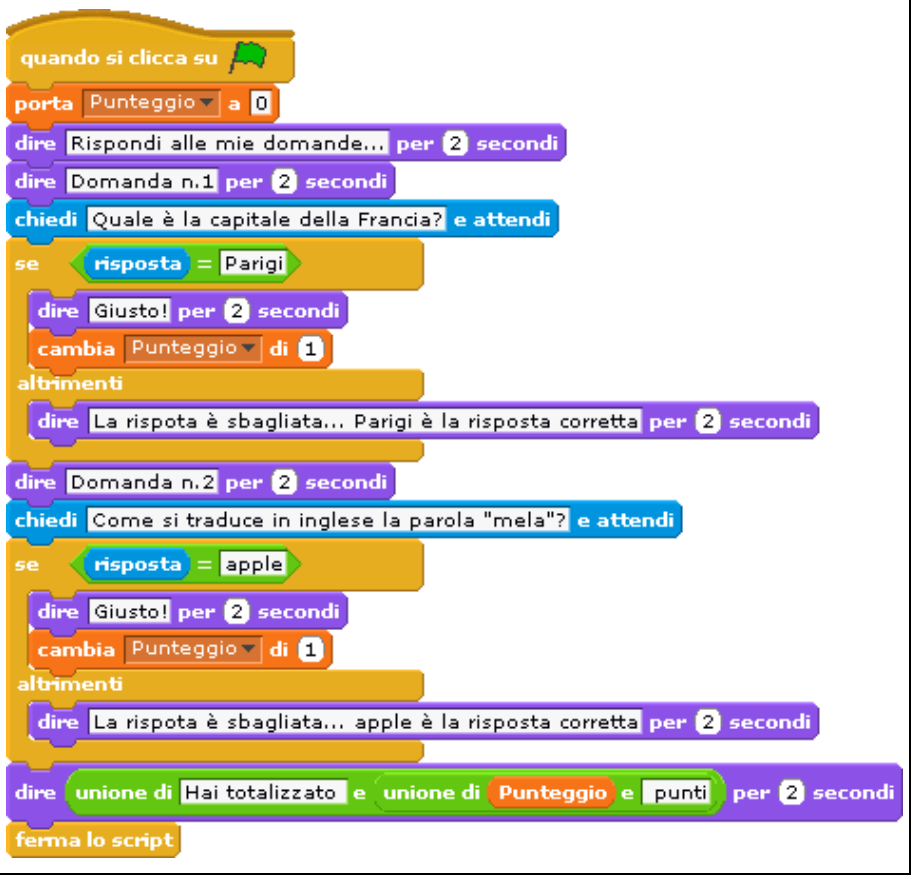
Istruzione condizionale; utilizzo di una variabile contatore.

Soluzione

Per calcolare il punteggio occorre dichiarare una variabile in modo da tenere memorizzato il conteggio delle risposte corrette:

attraverso i blocchi Porta e Cambia è possibile andare a modificarne il valore come nel progetto in cui la variabile viene posta a zero all'inizio e cambiata di una unità ogni volta che viene data la risposta corretta.

Gli script da progettare sono i seguenti:

Sprite	Codice
	

Attività da far svolgere in autonomia

Far proseguire il quiz aggiungendo altre domande su di uno specifico tema.

Decrementare il valore della variabile Punteggio di -1 nel caso di risposta errata.

In base al punteggio finale ottenuto, dare al termine un giudizio insufficiente, sufficiente, buono o ottimo in base al numero di risposte corrette date.

Esercizio svolto

http://scratch.mit.edu/projects/28952868/

Costumi con ragazzo avanti e indietro

Prerequisiti

Conoscenza istruzione condizionale.

Esercizio

Progettare un'animazione simulando una camminata.

Concetti di programmazione veicolati

Uso dei costumi degli sprite; gestione delle frecce della tastiera.

Soluzione

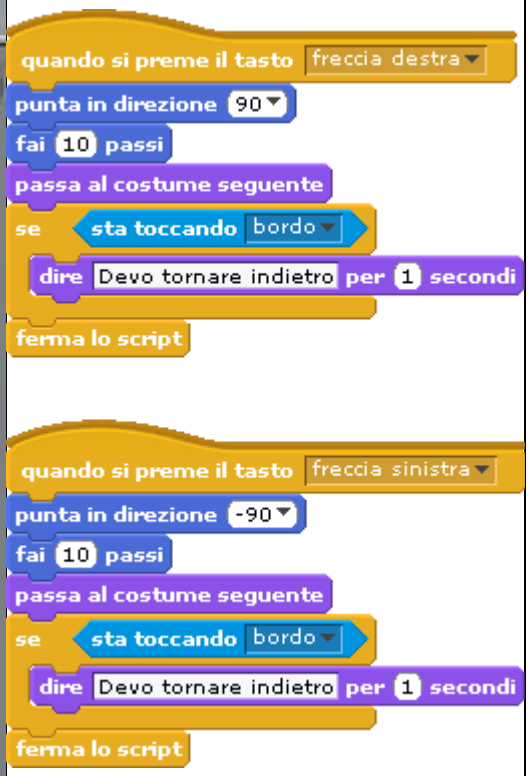
Uno sprite può aver associato dei costumi che gli permettono di cambiare aspetto simulando quindi una animazione. I costumi possono essere importati dalla libreria di sprite già impostata o essere disegnati col programma di disegno presente in Scratch.

In questo caso il progetto andrà in esecuzione alla pressione dei tasti freccia Destra e Sinistra in modo che lo sprite si muova verso destra o sinistra verso passando in automatico al costume seguente e creando l'effetto della camminata.

Verrà segnalato dallo stesso sprite quando verrà toccato il bordo.

In questo caso sono due gli script associati allo stesso sprite.

Gli script da progettare sono i seguenti:

Sprite	Codice
	

Attività da far svolgere in autonomia

Usare un altro sprite e programmare le quattro frecce destra, sinistra, alto e basso per far muovere lo sprite nelle quattro direzioni.

Usare in modo opportuno il blocco Punta in direzione provando diversi valori dell'angolo di rotazione.

Esercizio svolto

http://scratch.mit.edu/projects/28953774/

Drago sputafuoco

Prerequisiti

Conoscere come programmare i tasti freccia della tastiera.

Esercizio

Un gatto ed un drago si muovono sullo stage. Il gatto si muove attraverso la pressione dei tasti freccia presenti sulla tastiera, il drago lo insegue automaticamente e cerca di bruciare il gatto con la sua fiamma.

Concetti di programmazione veicolati

La programmazione ad eventi; la ripetizione in attesa di un evento; l'inizializzazione e il concetto di variabile.

Soluzione

Si utilizza il cronometro per contare quanto tempo resiste il gatto agli attacchi del drago. Serve inizializzare il cronometro: ovvero azzerarlo. Serve copiare in continuazione il valore della variabile di sistema cronometro in una variabile chiamata tempo per poter fermare il cronometro nel momento in cui il drago riesce a bruciare il gatto.

Gli script da progettare sono i seguenti:

Sprite	Codice

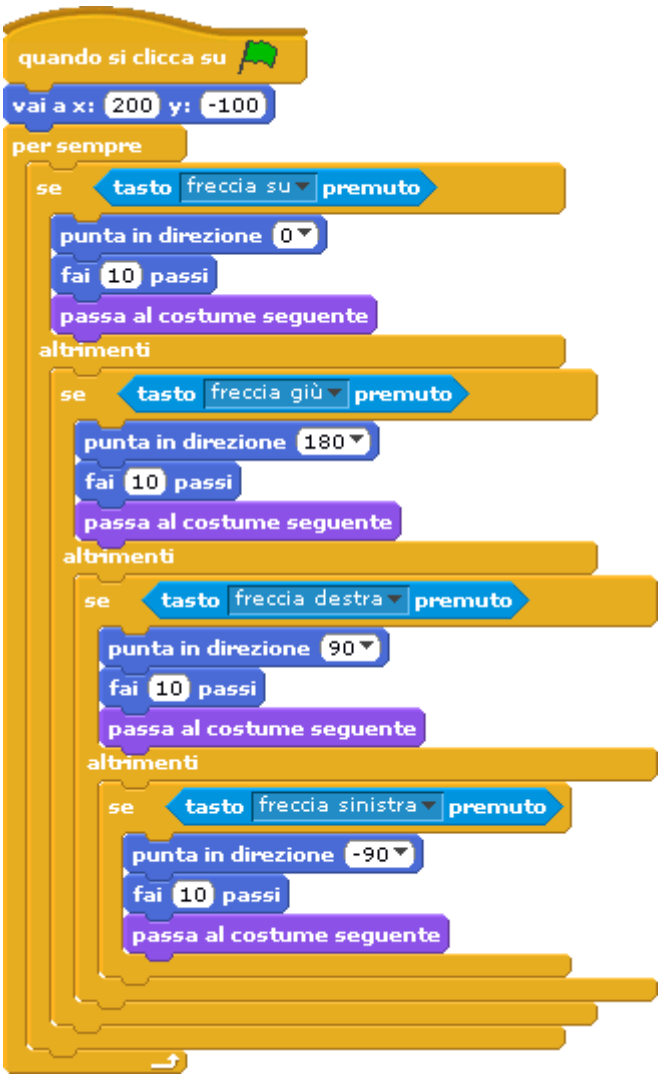
Sprite	Codice

Esercizio svolto

http://scratch.mit.edu/projects/19841887/

Variante

Una variante che rende più reattivo il movimento del gatto è la seguente. Qui si può nuovamente spiegare il costrutto di se… allora… e di se… allora… altrimenti…

Sprite	Codice
	

Esercizio svolto

http://scratch.mit.edu/projects/19841915/

Un semplice Paint

Prerequisiti

Saper distinguere le categorie a cui appartengono i blocchi.

Esercizio

Progettare un semplice Paint per disegnare a mano libera con tre colori diversi.

Concetti di programmazione veicolati

Gestione degli eventi.

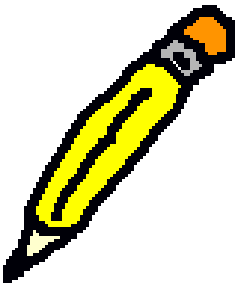
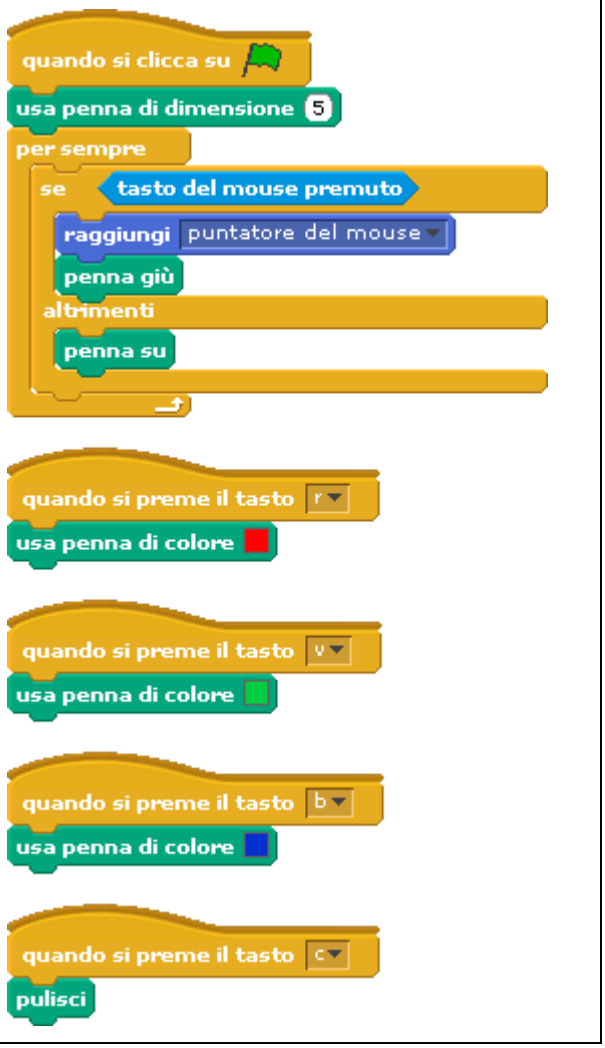
Soluzione

Quando viene premuto il tasto del mouse una matita virtuale, guidata dai movimenti dell'utente con il mouse, permette di scrivere e disegnare usando il colore rosso, verde e blu.

Il cambio di colore avviene alla pressione dei tasti "r", "v" e "b" come mostrato sullo stage.

Per cancellare il tutto basta premere il tasto "c".

Gli script da progettare sono i seguenti:

Sprite	Codice
	

Esercizio svolto

http://scratch.mit.edu/projects/20265276/

PacMan

Prerequisiti

Saper programmare le frecce della tastiera e conoscere l'iterazione infinita.

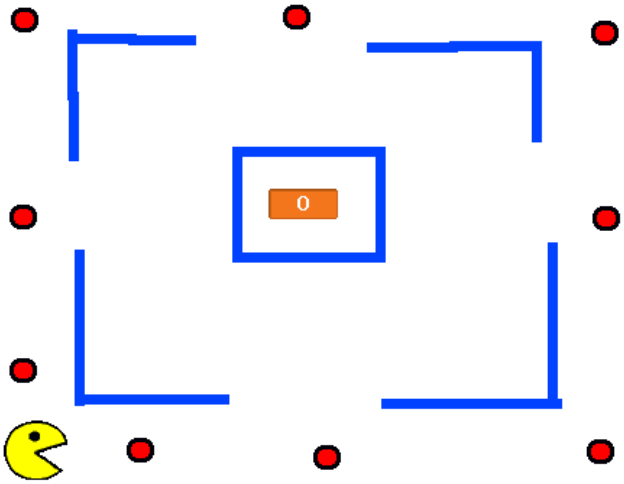
Esercizio

Progettare la base del gioco di Pacman.

Concetti di programmazione veicolati

Interazione tra sprite; gestione di più processi concorrenti.

Soluzione



Lo sprite Pacman presenta due costumi che gli permettono di animarsi quando si muove premendo le quattro frecce.

Viene usata una variabile punteggio che viene azzerata a inizio gioco e aumentata di una unità ogni volta che Pacman mangia un pallino.

Attraverso i blocchi Mostra e Nascondi è possibile agire sugli sprite facendoli comparire e sparire a piacimento.

Sono usati due sensori: il primo

serve a far ritornare PacMan all'inizio del gioco nel caso tocchi il labirinto; il secondo

serve a far nascondere il pallino come se Pacman lo avesse mangiato.

Gli script da progettare sono i seguenti:

Sprite	Codice

Sprite	Codice

Esercizio svolto

http://scratch.mit.edu/projects/28954982/

Pong

Prerequisiti

Saper gestire le frecce della tastiera.

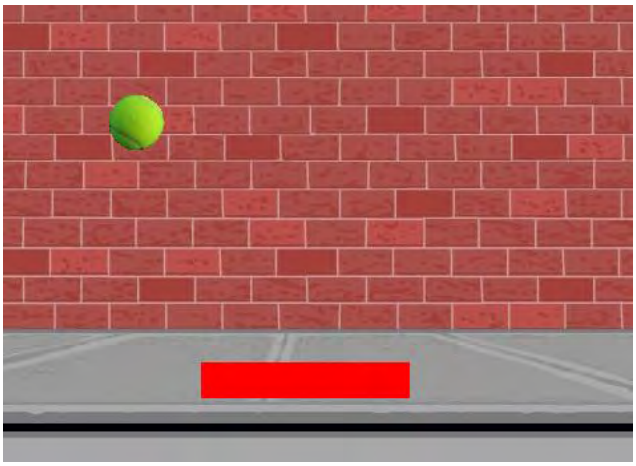
Esercizio

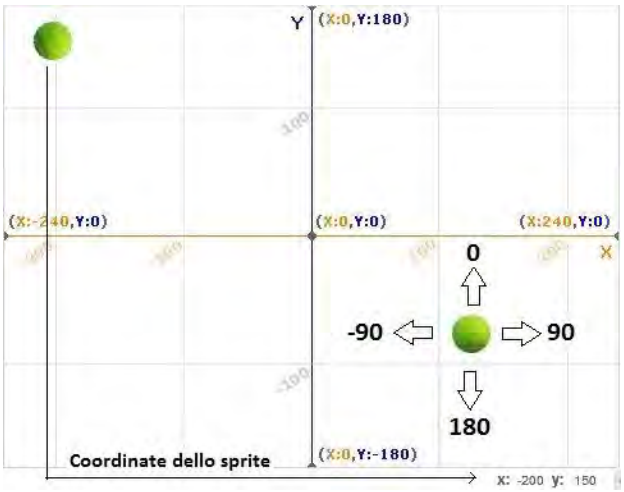
Il classico gioco del ping pong. Occorre colpire con la paletta una palla da tennis evitando che vada a colpire la linea nera di fine gioco.

Concetti di programmazione veicolati

Movimento attraverso lo studio delle coordinate spaziali.

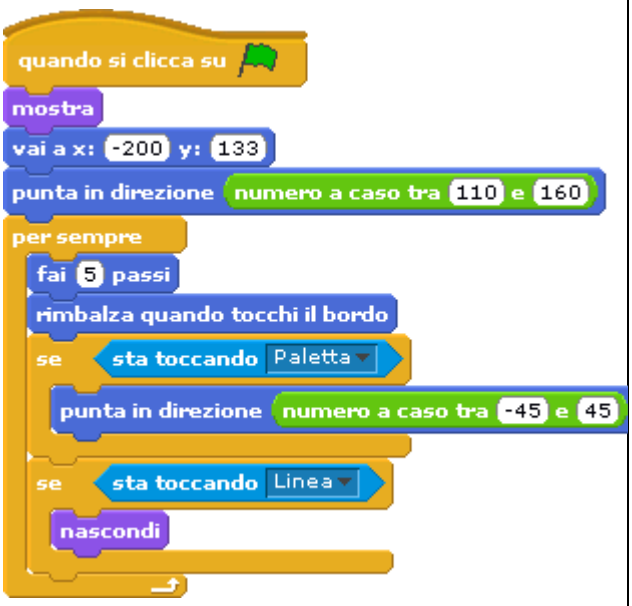
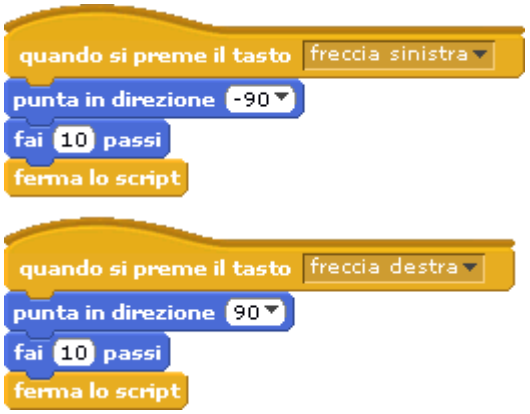
Soluzione





Per capire come viene generato casualmente l'angolo direzionale della pallina alla partenza e quando urta la paletta occorre fare riferimento al seguente schema. Mediante la generazione di un numero casuale è possibile ottenere quindi la direzione con cui dovrà muoversi la pallina.

Gli script da progettare sono i seguenti:

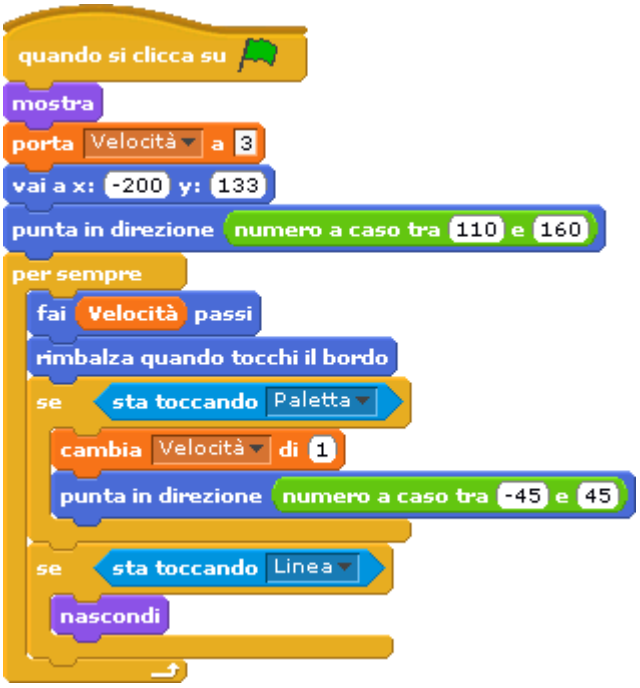
Sprite	Codice
	
	
	Non ha script associati

Esercizio svolto

http://scratch.mit.edu/projects/28955408/

Variante

Una variante che rende più reattivo il movimento della pallina consiste nel caricare la velocità della stessa pallina in una variabile che aumenterà il valore ogni volta che viene colpita dalla paletta aumentando così in automatico la sua velocità e rendendo il gioco più difficile.

Sprite	Codice
	

Esercizio svolto

http://scratch.mit.edu/projects/28955848/

Conversazione con messaggio

Prerequisiti

Saper distinguere le categorie a cui appartengono i blocchi.

Esercizio

tre sprite interagiscono tra di loro conversando.

Concetti di programmazione veicolati

Gestione e scambio dei messaggi tra processi; gestione del suono.

Soluzione

Mediante l'invio di messaggi e la gestione dell'evento attraverso il blocco cappello Quando ricevo un messaggio è possibile far comunicare e interagire tra di loro gli sprite per ottenere tutta una serie di situazioni come, nel nostro caso, una conversazione a tre. La procedura va sincronizzata in modo da rendere più reale l'azione.

Per la gestione dei suoni esistono dei blocchi che permettono di introdurre nel progetto suoni già registrati o di introdurne dei nuovi.

Gli script da progettare sono i seguenti:

Sprite	Codice
	quando si clicca su dire Ciao bel cagnone, come ti chiami? per 2 secondi produci suono miao e attendi la fine invia a tutti Frase al Cane e attendi dire Io sono Scratch il gatto per 2 secondi produci suono miao e attendi la fine dire E tu che voli alto nel cielo chi sei? per 2 secondi invia a tutti Frase al Pappagallo e attendi dire Bene, allora possiamo giocare! per 2 secondi ferma lo script
	quando ricevo Frase al Cane dire Mi chiamo Fido. E tu? per 2 secondi produci suono Dog1 e attendi la fine produci suono Dog1 e attendi la fine ferma lo script
	quando ricevo Frase al Pappagallo dire Ciao Scratch, io sono Cocorito per 2 secondi produci suono Duck e attendi la fine produci suono Duck e attendi la fine ferma lo script

Esercizio svolto

http://scratch.mit.edu/projects/28956244/

Fruit game

Prerequisiti

Saper disegnare con il computer (utilizzo di Paint): Scratch ha al suo interno uno strumento per disegnare simile a Paint. Saper cercare immagini in Internet e manipolare fotografie/immagini.

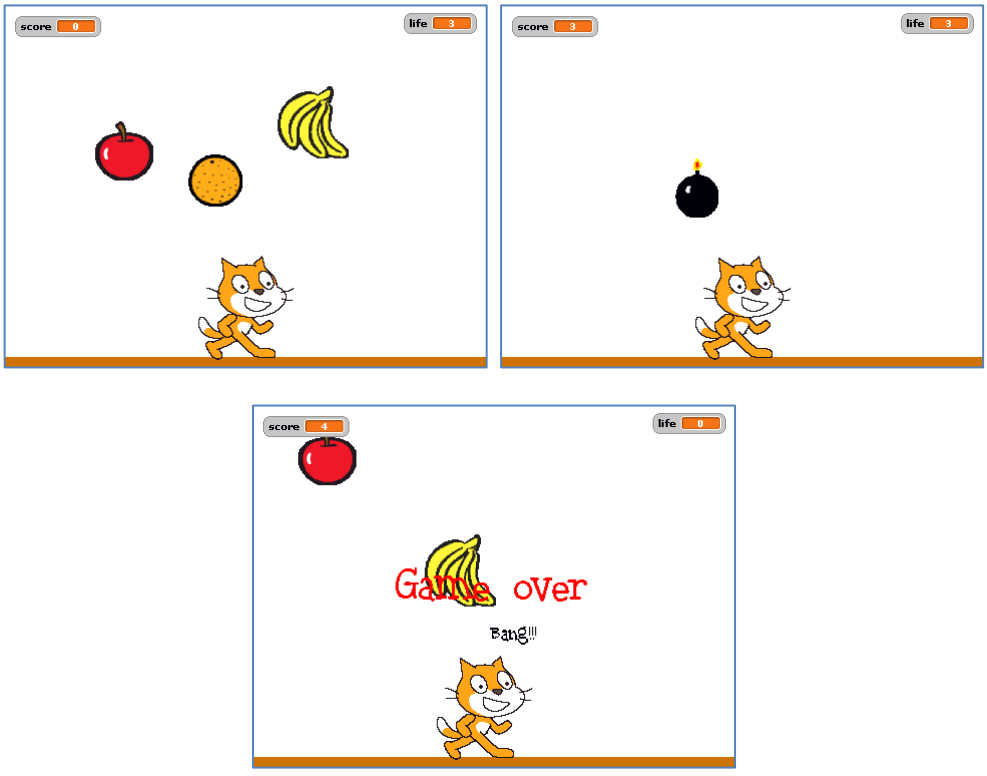
Esercizio

Il gatto deve mangiare la frutta che scende dal cielo ed evitare la bomba. Il gatto ha tre vite a disposizione.

Concetti di programmazione veicolati

Inizializzazione, sincronizzazione, uso delle variabili.

Soluzione



Con questo gioco si imparano ad usare diversi nuove istruzioni.

L'istruzione per generare un numero casuale per far si che i frutti cadano da posizioni sempre diverse e con tempi di comparsa differenti.

Anche la velocità di caduta degli oggetti è diversa, questa si può ottenere facendo compiere un numero diverso di passi ai diversi sprite.

L'istruzione ripeti fino a quando viene presentata con una condizione composta: *sto toccando il colore del terreno* oppure *sto toccando il gatto*. Questa condizione è soddisfatta quando o il frutto tocca il terreno oppure tocca il gatto ma potrebbe anche essere che vi sia la contemporaneità degli eventi ovvero che tocca il gatto ed il terreno contemporaneamente.

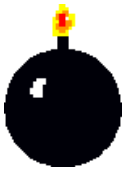
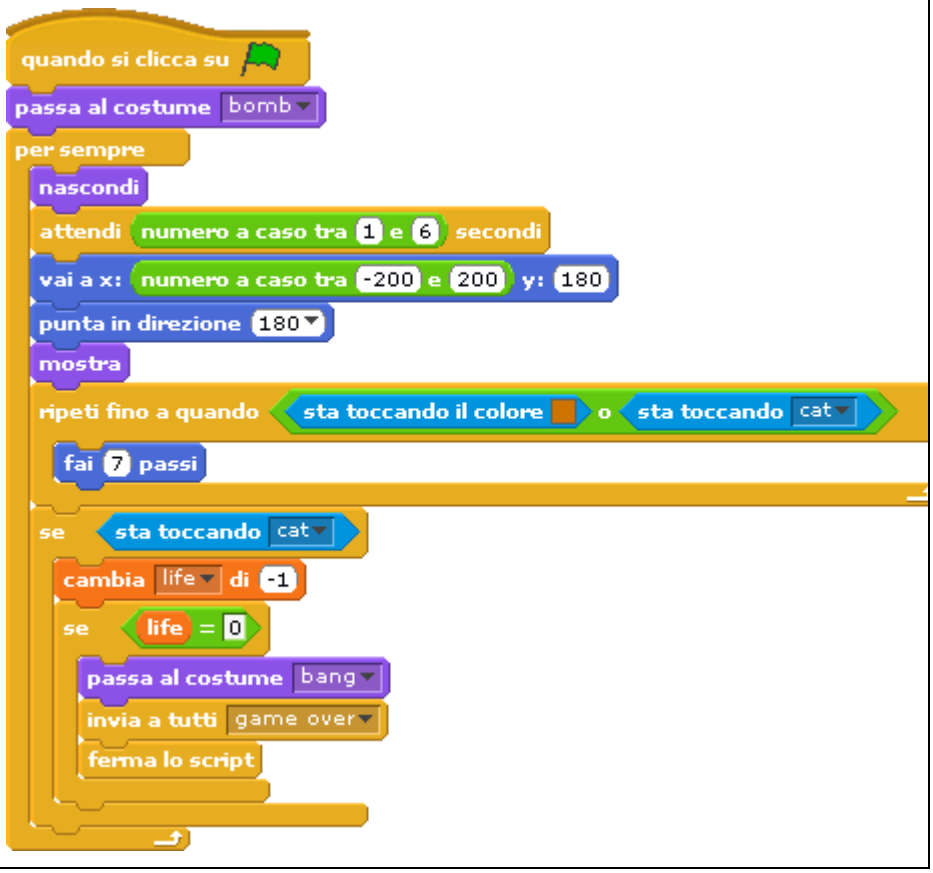
Quando il gatto perde tutte e tre le vite è necessario far comparire la dicitura *Game Over* e quindi fermare il gioco. Questo è possibile utilizzando una sincronizzazione tramite messaggio tra lo sprite bomba che lo invia e lo sprite Game Over che ne è in attesa.

Da notare infine l'utilizzo dell'istruzione Ferma tutto che permette di concludere tutti gli script in esecuzione, in particolare quelli del gatto e dei frutti.

Gli script da progettare sono i seguenti:

Sprite/Stage	**Codice**

Sprite/Stage	Codice

Sprite/Stage	Codice
	

Sprite/Stage	Codice
Game over	

Esercizio svolto

http://scratch.mit.edu/projects/20824302/

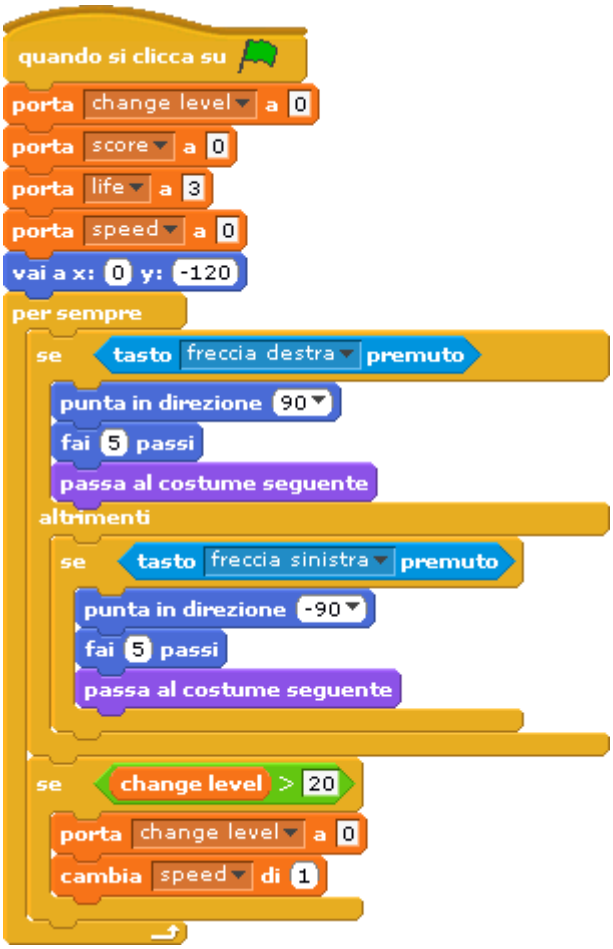
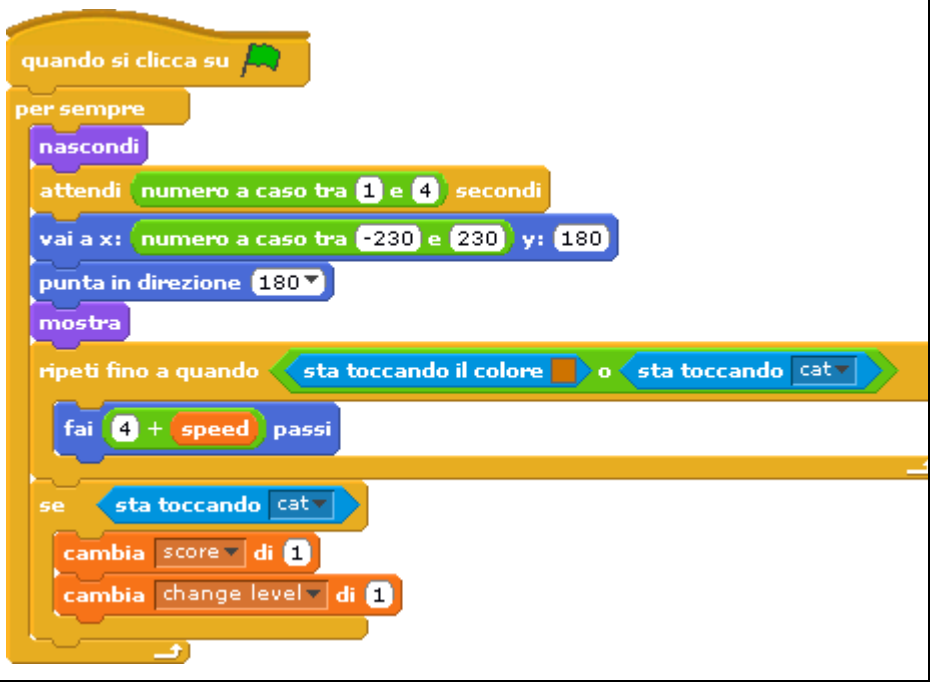
Variante

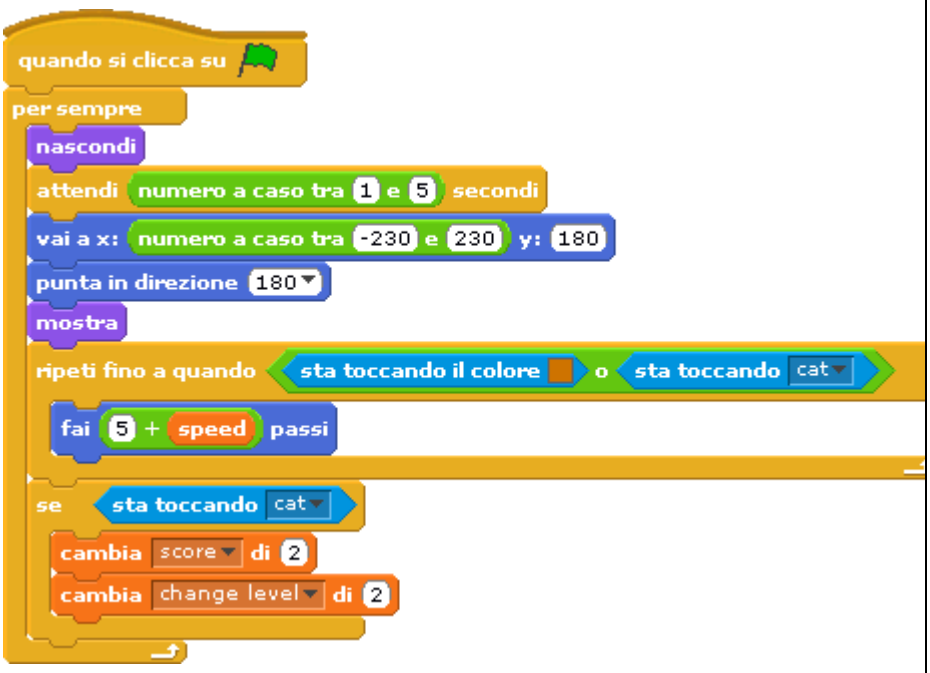
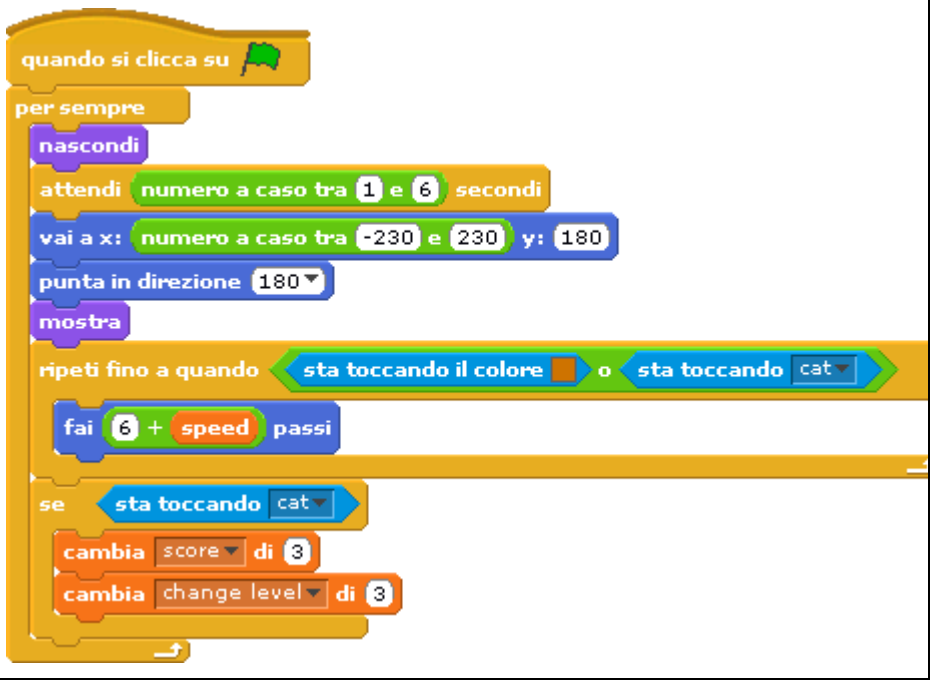
Si vuole far aumentare la velocità di caduta degli oggetti dal cielo man mano che aumenta il punteggio.

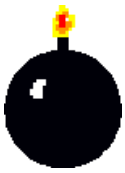
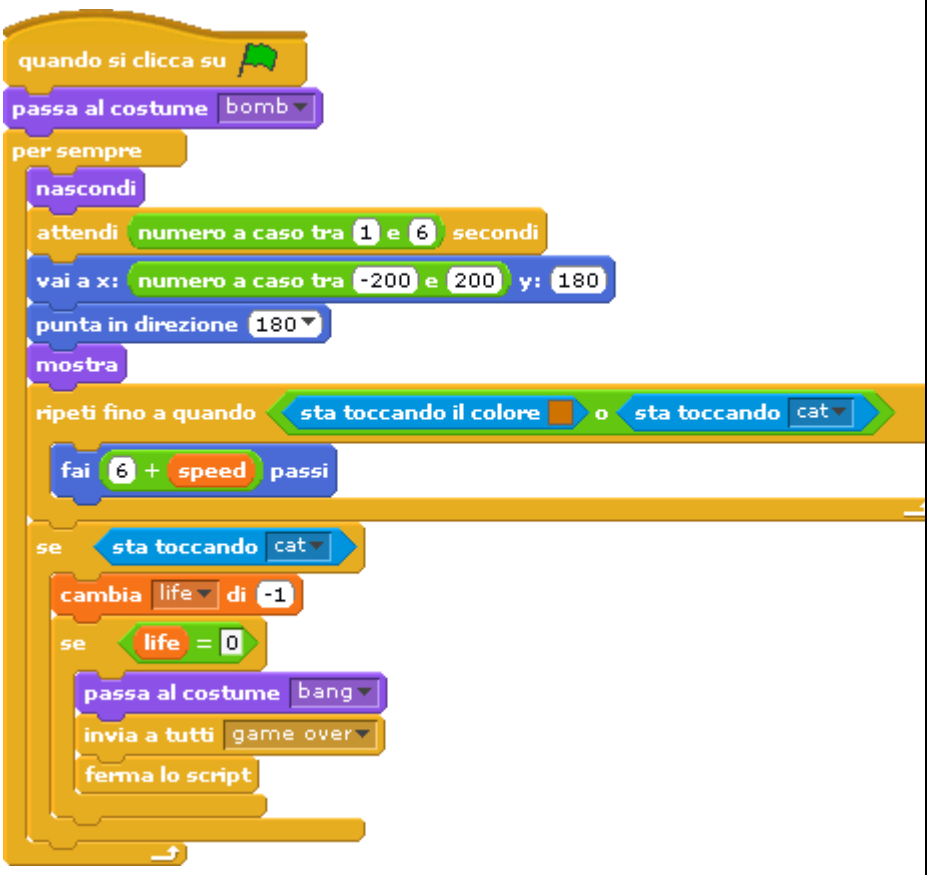
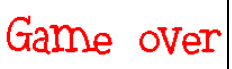
La regola che si vuole seguire è la seguente: ad ogni incremento di 20 punti deve essere aumentata di un passo la velocità di discesa degli oggetti.

Per farlo si utilizzano due variabili in più: speed e change-level. Questa viene incrementata con la stessa logica della variabile score ma ogni volta che il valore di change-level supera i 20 punti viene incrementata di 1 la variabile speed e la variabile change-level deve essere azzerata per poter ricominciare a contare i 20 punti necessari al nuovo incremento di velocità.

Operativamente la variabile speed fa aumentare la velocità degli oggetti perché viene aggiunta al numero di passi che normalmente compie l'oggetto ad inizio del gioco.

Sprite/Stage	Codice
	
	

Sprite/Stage	Codice
	
	

Sprite/Stage	Codice
 Bang!!!	
	

Esercizio svolto

http://scratch.mit.edu/projects/20824339/

Storytelling: Il lupo e l'agnello

Prerequisiti

Saper disegnare con il computer (utilizzo di Paint): Scratch ha al suo interno uno strumento per disegnare simile a Paint. Saper cercare immagini in Internet e manipolare fotografie/immagini.

Saper raccontare una storia

Esercizio

Raccontare la storia del lupo e dell'agnello:

Lungo la sponda di una riva sono venute a bere due bestie: il lupo più in alto e, molto più in basso, l'agnello: pasto eccellente per quello scannatore vorace

Il lupo vuol fare l'offeso.

- Perchè intorbidi l'acqua che sto bevendo?

E l'agnello tutto timoroso:

- Scusa, come posso fare ciò che tu mi rimproveri? Io bevo l'acqua che passa prima da te.

- Sei mesi fa hai parlato male di me- riprende il lupo.

- Ma se ancora non ero nato!

- Allora fu tuo padre a parlare male di me.

Così, lo afferra e lo sbrana.

Morale: la favola mostra che non c'è possibilità per un debole di difendersi contro un potente che si propone di fargli torto.

Concetti di programmazione veicolati

Inizializzazione, sincronizzazione.

Soluzione

Raccontare una storia con diversi personaggi (sprite) e sfondi (stage) implica saper gestire correttamente la sincronizzazione tramite messaggi.

Come si può vedere dalla soluzione proposta si deve innanzitutto procedere alla corretta collocazione iniziale dei personaggi e dello sfondo, non solo come posizione ma anche come visibilità.

Poi è tutto un lavoro di scambio di messaggi e di sincronizzazione sugli stessi da parte dei personaggi che popolano la storia.

Una possibile variante a questo esercizio è quello di registrare le voci dei personaggi e farle sentire al posto della visualizzazione dei fumetti.

Gli script da progettare sono i seguenti:

Sprite/Stage	Codice
Sfondo 	

Sprite/Stage	Codice
Sfondo 	quando ricevo inizio passa allo sfondo water-and-rocks
	quando ricevo inizio mostra dire Un lupo ed un agnello per 2 secondi dire spinti dalla sete per 2 secondi dire erano giunti allo stesso ruscello per 2 secondi dire Più in alto si fermò il lupo per 2 secondi invia a tutti lupo in scena dire molto più in basso si mise l'agnello per 2 secondi invia a tutti agnello in scena nascondi invia a tutti lupo1
	quando ricevo lupo in scena mostra vai a x: -142 y: 102
	quando ricevo agnello in scena mostra

Sprite/Stage	Codice
	quando ricevo lupo1 dire Perche' intorbidi l'acqua che sto bevendo? per 2 secondi invia a tutti agnello1
	quando ricevo agnello1 dire Scusa, come posso fare ciò che tu mi rimproveri? per 2 secondi dire Io bevo l'acqua che passa prima da te per 2 secondi invia a tutti lupo2
	quando ricevo lupo2 dire Sei mesi fa hai parlato male di me per 2 secondi invia a tutti agnello2
	quando ricevo agnello2 dire Ma se ancora non ero nato! per 2 secondi invia a tutti lupo3 attendi fino a quando sta toccando lupo ripeti 10 volte cambia effetto fantasma di 25 attendi 0.5 secondi nascondi invia a tutti morale

Sprite/Stage	Codice
Sfondo La favola mostra che non c'è possibilità per un debole di difendersi contro un potente che si propone di fargli torto.	

Esercizio svolto

http://scratch.mit.edu/projects/19855831/

Poligoni regolari

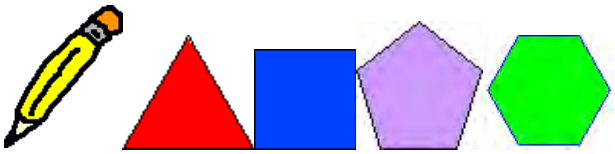
Prerequisiti

Saper disegnare con il computer (utilizzo di Paint): Scratch ha al suo interno uno strumento per disegnare simile a Paint, saper cercare immagini da Internet

Conoscenza dei poligoni regolari, angoli e angoli supplementari.

Esercizio

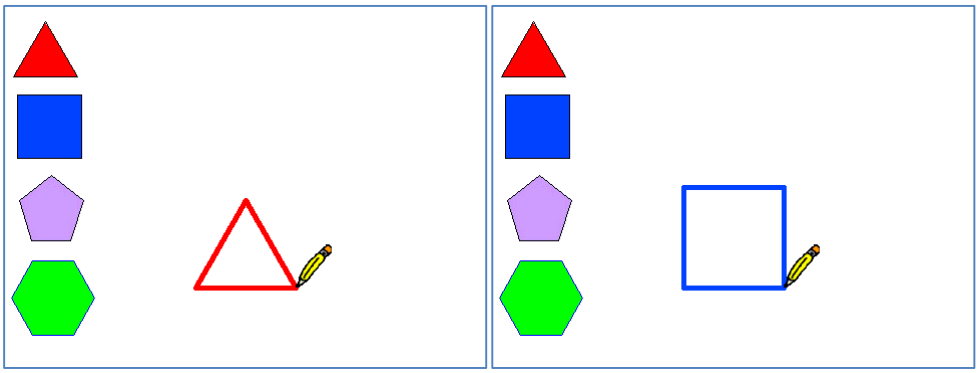
Utilizzare uno stage vuoto e cinque sprite: uno raffigurante una matita e altri quattro raffiguranti i poligoni regolari triangolo, quadrato, pentagono ed esagono. Scopo dell'esercizio è quello di far disegnare alla matita il poligono regolare su cui l'operatore cliccherà con il mouse.

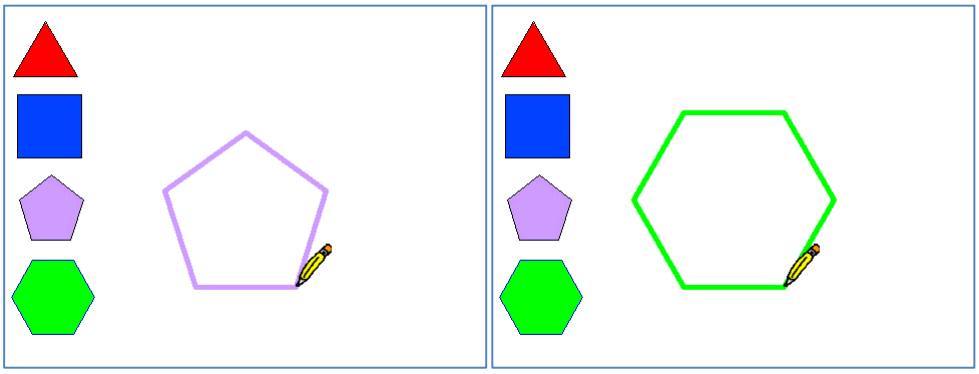


Concetti di programmazione veicolati

La sincronizzazione e la ripetizione.

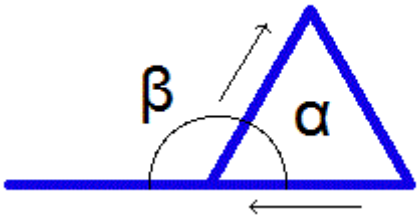
Soluzione



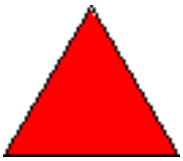
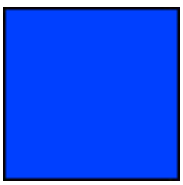


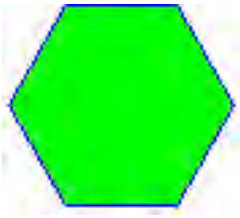
Agganciare alle conoscenze di geometria sui poligoni regolari, sugli angoli interni a tali poligoni ed ai relativi angoli supplementari il concetto di ripetizione per un determinato numero di volte.

Ad esempio nel triangolo equilatero gli angoli interni (α) sono tutti di 60° quindi il relativo supplementare(β) è di 120°, quindi la matita che traccia un lato del triangolo per poter tracciare il lato successivo deve effettuare una rotazione di 120° cioè della misura dell'angolo supplementare.



Gli script da progettare sono i seguenti:

Sprite	Codice
	
	

Sprite	Codice
	

Sprite	Codice

Esercizio svolto

http://scratch.mit.edu/projects/19841831/

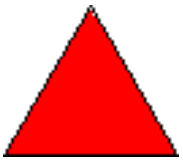
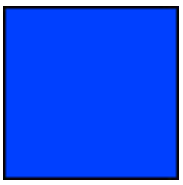
Variante

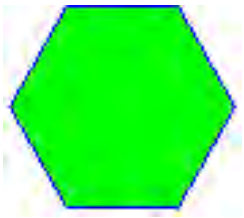
L'informatico cerca sempre una soluzione che faccia un minor uso di istruzioni possibile. Allora una soluzione interessante è quella di constatare che l'angolo supplementare nei poligoni regolari è uguale al risultato dell'operazione di 360 diviso il numero dei lati del poligono.

Per fare questo serve introdurre il concetto di variabile, ovvero di un contenitore che può contenere un valore, molte volte numerico ma non solo, che può cambiare nel tempo, man mano che le azioni si susseguono.

Servono per svolgere l'esercizio 2 variabili:

- una per contenere il numero dei lati del poligono che si desidera disegnare e che viene impostata dal click sul disegno di tale poligono;
- l'altra per impostare il codice del colore con cui si vuole disegnare il poligono.

Sprite	Codice
	
	

Sprite	Codice
	

Attività da far svolgere

Far generare alcuni poligoni con un numero superiore di lati. Attenzione sarà necessario diminuire la lunghezza del lato.

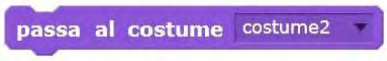
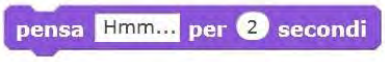
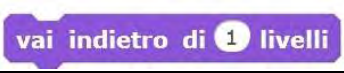
Esercizio svolto

http://scratch.mit.edu/projects/20098973/

Glossario dei Blocchi

Qui di seguito è possibile visionate tutti i blocchi e le loro funzionalità

Blocchi Movimento	
BLOCCO	**DESCRIZIONE**
	Muove lo sprite avanti (usando valori positivi) o indietro (usando valori negativi)
	Ruota lo sprite in senso orario per i gradi indicati
	Ruota lo sprite in senso antiorario per i gradi indicati
	Fa spostare lo spite in determinata direzione (0=su, 90=destra, -90= sinistra, 180=giù)
	Fa puntare lo sprite verso il puntatore del mouse o verso un altro sprite
	Sposta lo sprite alla direzione x e y indicata
	Fa spostare lo sprite dove si trova il puntatore del mouse o un altro sprite
	Fa spostare lo sprite nel punto indicato dalle x e dalle y facendolo scivolare nel tempo indicato.
	Modifica la posizione X dello sprite del valore specificato
	Porta lo sprite dove X è al valore specificato
	Modifica la posizione Y dello sprite del valore specificato
	Porta lo sprite dove Y è al valore specificato
	Fa ruotare lo sprite in direzione opposta quando tocca il bordo
	Riporta la posizione X dello sprite
	Riporta la posizione Y dello sprite
	Riporta la direzione dello sprite (0=su, 90=destra, -90= sinistra, 180=giù)

Blocchi Aspetto	
BLOCCO	**DESCRIZIONE**
	Cambia l'aspetto dello sprite passando al costume specificato
	Cambia l'aspetto dello sprite passando al costume successivo della lista dei costumi (se il costume è l'ultimo della lista, ritorna al primo)
	Riporta il numero attuale del costume dello sprite
	Mostra il messaggio specificato in un fumetto (effetto parola) per il tempo indicato
	Mostra un il testo specificato in un fumetto (effetto parola)
	Mostra il messaggio specificato in un fumetto (effetto pensiero) per il tempo indicato
	Mostra un il testo specificato in un fumetto (effetto pensiero)
	Cambia l'effetto grafico dello sprite del valore specificato (selezionare dal menù il tipo di effetto grafico desiderato)
	Porta un effetto grafico al valore specificato (la maggior parte degli effetti grafici ha valore tra 0 e 100)
	Rimuove tutti gli effetti grafici dello sprite
	Cambia la dimensione dello sprite della quantità specificata
	Porta la dimensione dello sprite alla percentuale specificata (il valora fa riferimento alla dimensione originale dello sprite)
	Riporta la percentuale dell'attuale dimensione dello sprite
	Fa comparire lo spirte nello stage
	Fa scomparire lo sprite dallo stage
	Port in primo piano lo sprite rispetto a quelli presenti sullo stage
	Muovi lo sprite indietro del numero di livelli indicato
	Cambia l'aspetto dello stage passando allo sfondo specificato

passa allo sfondo seguente	Cambia l'aspetto delle stage passando allo sfondo successivo
numero sfondo	Riporta il numero dello sfondo attuale dello stage

Blocchi suono	
BLOCCO	**DESCRIZIONE**
produci suono meow	Inizia a riprodurre il suono selezionato tramite il menu e passa SUBITO all'esecuzione del blocco successivo, anche se il suono non è terminato
produci suono meow e attendi la fine	Inizia a riprodurre il suono selezionato tramite il menu e passa ATTENDE IL TERMINE del suono per l'esecuzione del blocco successivo
arresta tutti i suoni	Arresta la riproduzione di tutti i suoni
suona tamburo 1 per 0.25 battute	Riproduce il suono di uno strumento a percussione, selezionabile dal menu, per il numero di battute specificato
suona nota 60 per 0.5 battute Do Medio (60)	Suona una nota musicale per il numero di battute indicato (occorre utilizzare la tastiera per selezionare una nota)
fai una pausa di 0.25 battute	Fa una pausa del numero di battute indicato
passa a strumento 1	Passa allo strumento specificato. Lo strumento verrà utilizzato dallo sprite per i successivi blocchi "suona nota"
cambia volume di -10	Cambia il volume dei suoni dello sprite del valore indicato (tra 0 e 100, valori positivi e negativi)
porta volume a 100 %	Porta al valore specificato il volume de suoni riprodotti dallo sprite
volume	Riporta il volume dei suoni riprodotti dallo sprite
cambia tempo di 20	Cambia la velocità dei suoni riprodotti per il valore specificato
porta tempo a 60 bpm	Porta il tempo dei suoni riprodotti dallo sprite al valore di battute al minuto specificato
tempo	Riporta il tempo dei suoni riprodotti dallo sprite (i valori sono in battute per minuto)

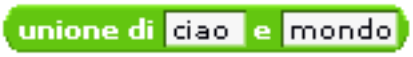
Blocchi Controllo	
BLOCCO	**DESCRIZIONE**
	Esegue lo script ad esso collegato quando si clicca sulla bandiera verde
	Esegue lo script associato quando viene premuto un tasto specificato
	Esegue lo script ad esso collegato quando si clicca su questo sprite
	Attende il numero di secondi indicato prima di riprodurre lo sprite successivo
	Esegue all'infinito i blocchi inseriti al suo interno
	Ripete il numero di volte indicato i blocchi inseriti al suo interno
	Invia un messaggio a tutti gli sprite (incluso se stesso) attivando l'attivazione di alcuni blocchi. Non attende l'esecuzione di tali blocchi prima di proseguire
	Invia un messaggio a tutti gli sprite (incluso se stesso) attivando l'attivazione di alcuni blocchi. Attende il termine dell'esecuzione di tali blocchi prima di proseguire.
	Esegue lo script ad esso collegato quando riceve un determinato messaggio
	Esegue un continuo controllo ed esegue i blocchi inseriti al suo interno quando si verifica la specifica condizione indicata.
	Se si verifica una determinata condizione indicata esegue i blocchi contenuti al suo interno

	Se sei verifica una determinata condizione esegue i blocchi inseriti nella parte superiore, altrimenti, quando tale condizione non si verifica, quelli della parte inferiore
	Attende fino a quando la condizione indicata non risulta vera e solo allora esegue gli script associati
	Ripete i blocchi inseriti al suo interno fino a quando la condizione continua ad essere vera
	Ferma l'esecuzione dei blocchi ad esso collegati
	Ferma l'esecuzione di tutti i blocchi del progetto

Blocchi penna	
BLOCCO	**DESCRIZIONE**
	Cancella dallo stage tutto ciò che è stato disegnato con la penna e tutti i timbri
	Appoggia la penna sullo stage: lo sprite mentre si sposta traccia una linea
	Alza la penna dallo stage: lo sprite può spostarsi senza tracciare alcuna linea
	Cambia il colore della penna. Il colore viene scelto con il contagocce
	Usa la penna del colore indicato (0 §= colore rosso; 100 = colore blu. Le tonalità rispecchiano i colori dell'arcobaleno)
	Cambia il valore indicato della luminosità della penna dello sprite
	Porta la luminosità della penna al valore specificato (0= molto scuro; 100=molto chiaro)
	Modifica la dimensione del tratto della penna
	Sa una penna delle dimensioni indicate
	Riproduce l'immagine dello sprite sullo stage

Blocco sensori	
BLOCCO	**DESCRIZIONE**
	La condizione è vera se lo sprite sta toccando il puntatore del mouse, un altro sprite o il bordo dello stage.
	La condizione è vera se lo sprite sta toccando il colore indicato (il colore viene scelto utilizzando il contagocce)
	La condizione è vera se il primo colore specificato sta toccando il secondo colore specificato (entrambi si scelgono con il contagocce)
	Visualizza la domanda indicata e una casella di testo in cui è possibile digitare una risposta. La conferma della risposta avviene dopo che è stato premuto il tasto “invio”.
	Memorizza l’ultima risposta data tramite il blocco “chiedi e...attendi”. Il valore è globale perché condiviso con tutti gli sprite.
	Riporta la posizione x del puntatore del mouse
	Riporta la posizione y del puntatore del mouse
	La condizione è vera quando il tasto del mouse viene premuto
	La condizione è vera quando il tasto specificato viene premuto
	Riporta la distanza dello sprite dal puntatore del mouse o da uno specifico sprite
	Riporta il valore, espresso in secondi, del cronometro. Il cronometro è sempre in esecuzione.
	Riporta il valore (espresso in secondi) del cronometro. Il cronometro è sempre in esecuzione.
	Riporta il valore relativo ad una specifica proprietà (posizione x, posizione y, direzione, numero, costume, dimensione e volume) di un determinato sprite
	Riporta il volume dei suoni catturati dal microfono del computer (il valore è compreso tra 0 e 100)
	La condizione risulta vera quando i suoni catturati dal microfono del computer superano il valore 30
	Riporta il valore del sensore specificato.

	La condizione è vera quando il pulsante è premuto o i contatti della Scratch Sensor Board sono connessi.

Blocchi Operatori	
BLOCCO	**DESCRIZIONE**
	Esegue un'addizione e genera un risultato
	Esegue una sottrazione e genera un risultato
	Esegue una moltiplicazione e genera un risultato
	Esegue una divisione e genera un risultato
	Sceglie un numero intero a caso compreso nell'intervallo specificato
	La condizione risulta vera quando il primo valore è inferiore al secondo
	La condizione risulta vera quando il primo valore è uguale al secondo
	La condizione risulta vera quando il primo valore è superiore al secondo
	La condizione è vera quando entrambe le condizioni sono vere
	La condizione è vera quando una delle due condizioni è vera
	Inverte il valore di una condizione da vero in falso e viceversa
	Unisce due stringhe alfanumeriche o valori
	Riporta la lettera che occupa la posizione specificata di una stringa o di un valore
	Riporta il numero di caratteri di una stringa alfanumerica o di un valore
	Riporta il resto della divisione dei numeri specificati
	Riporta il numero intero più vicino al numero o al valore specificato
	Riporta il risultato della funzione selezionata (abs, sqtr, sin, cos, tan, asin, acos, atan, In, log, e^,10^) applicata al numero specificato

Blocchi Variabili	
BLOCCO	**DESCRIZIONE**
	Riporta il valore della variabile
	Porta il valore della variabile selezionata al valore indicato
	Modifica il valore della variabile significata della quantità specificata
	Apre un visualizzatore indicante il valore della variabile sullo stage
	Nasconde il visualizzatore
	Riporta tuti gli elementi contenuti nella lista
	Aggiunge alla lista l'elemento indicato. L'elemento viene inserito sempre in ultima posizione
	Rimuove un elemento occupante una specifica posizione nella lista, il primo, l'ultimo o tutti gli elementi
	Inserisce un elemento specifico nella posizione indicata (specifica, prima, ultima o qualunque) in una determinata lista
	Sostituisce un elemento occupante una determinata posizione (specifica, prima, ultima o qualunque), in una specifica lista con l'elemento indicato
	Riporta l'elemento occupante una determinata posizione (specifica, prima, ultima o qualunque), nella lista selezionata
	Riporta il numero totale degli elementi presenti nella lista selezionata
	La condizione risulta vera quando la lista contiene esattamente il valore specificato

Indice

Introduzione 1
Autori 7
Scratch: la programmazione per gioco 11
Scratch versione 1.4 11
Scratch versione 2.0 13
Acquario 15
Prerequisiti 15
Esercizio 15
Concetti di programmazione veicolati 15
Soluzione 15
Attività da far svolgere in autonomia 16
Esercizio svolto 16
Conversazione 17
Prerequisiti 17
Esercizio 17
Concetti di programmazione veicolati 17
Soluzione 17
Attività da far svolgere in autonomia 18
Esercizio svolto 18
Quiz con punteggio 19
Prerequisiti 19
Esercizio 19
Concetti di programmazione veicolati 19
Soluzione 19
Attività da far svolgere in autonomia 20
Esercizio svolto 20
Costumi con ragazzo avanti e indietro 21
Prerequisiti 21
Esercizio 21
Concetti di programmazione veicolati 21
Soluzione 21
Attività da far svolgere in autonomia 22
Esercizio svolto 23

Drago sputafuoco 24
Prerequisiti 24
Esercizio 24
Concetti di programmazione veicolati 24
Soluzione 24
Esercizio svolto 26
Variante 27
Esercizio svolto 27
Un semplice Paint 28
Prerequisiti 28
Esercizio 28
Concetti di programmazione veicolati 28
Soluzione 28
Esercizio svolto 29
PacMan 30
Prerequisiti 30
Esercizio 30
Concetti di programmazione veicolati 30
Soluzione 30
Esercizio svolto 32
Pong 33
Prerequisiti 33
Esercizio 33
Concetti di programmazione veicolati 33
Soluzione 33
Esercizio svolto 34
Variante 35
Esercizio svolto 35
Conversazione con messaggio 36
Prerequisiti 36
Esercizio 36
Concetti di programmazione veicolati 36
Soluzione 36
Esercizio svolto 37
Fruit game 38

Prerequisiti 38
Esercizio 38
Concetti di programmazione veicolati 38
Soluzione 38
Esercizio svolto 42
Variante 43
Esercizio svolto 46
Storytelling: Il lupo e l'agnello 47
Prerequisiti 47
Esercizio 47
Concetti di programmazione veicolati 47
Soluzione 48
Esercizio svolto 52
Poligoni regolari 53
Prerequisiti 53
Esercizio 53
Concetti di programmazione veicolati 53
Soluzione 53
Esercizio svolto 56
Variante 57
Attività da far svolgere 58
Esercizio svolto 58
Glossario dei Blocchi 59

Guida per gli studenti

www.associazionedschola.it/programmoanchio

Quest'opera è distribuita con licenza Creative Commons Attribuzione 3.0 Italia.