

Programmo Anch'io



Guida all'uso di

SCRATCH

Versione Docenti

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 didattico e tecnico di stimolo e supporto, rivolto ai docenti formatori che potranno effettuare, nell'ambito degli esercizi proposti, scelte specifiche per la realizzazione dei moduli didattici a carattere soprattutto laboratoriale da adattare all'età, alle conoscenze ed abilità degli studenti cui gli stessi moduli saranno rivolti.

L'efficacia del suddetto strumento, unitamente alle capacità didattico relazionali dei docenti, farà sì che attraverso il coinvolgimento degli studenti, gli stessi potranno avvicinarsi alla programmazione in modo semplice e piacevole ed acquisire le competenze necessarie per poter cimentarsi in maniera autonoma o di gruppo nella produzione di prodotti con caratteristiche pluridisciplinari con la fantasia, l'estro e la creatività che è propria dei nostri giovani, sempre più nativi digitali.

Il Presidente Dschola

Alfonso Lupo

La **Fondazione Cassa di Risparmio di Torino** è un ente non-profit interamente dedicato alla crescita e allo sviluppo economico, sociale e culturale del Piemonte e della Valle d'Aosta. Prende vita nel 1991 dalla Cassa di Risparmio di Torino, da cui ha ereditato lo spirito filantropico. E' persona giuridica privata, senza fine di lucro, dotata di piena autonomia gestionale, orientata a scopi di utilità sociale e di promozione dello sviluppo economico.

La Fondazione indirizza progetti e risorse in molteplici settori riconducibili a tre macro-aree di intervento: Arte e Cultura, Ricerca e Istruzione, Welfare e Territorio. Un impegno che in oltre 20 anni di attività si è tradotto in oltre 1 miliardo e 500 milioni di euro, distribuiti prevalentemente nel Nord Ovest d'Italia.

Convinta della centralità che la ricerca e l'istruzione rivestono per lo sviluppo del capitale umano dei cittadini di domani, la Fondazione CRT ha messo in campo in questi settori diversi progetti per sostenere il sistema scolastico del Piemonte e della Valle d'Aosta, per supportare i talenti, le politiche di placement e la ricerca accademica.

Il progetto "*Programmo anch'io*", proposto e realizzato dall'Associazione *Dschola*, si inserisce nell'ambito del progetto Diderot della *Fondazione CRT*, grazie al quale tutte le scuole primarie e secondarie del Piemonte e della Valle d'Aosta possono integrare i propri piani di offerta formativa con un ricco programma di lezioni, laboratori, spettacoli, mostre e visite didattiche.

Tali iniziative, la cui partecipazione è gratuita, consentono agli studenti di approfondire le materie tradizionali con metodologie innovative e di avvicinarsi a discipline che non sempre vengono inserite nei programmi curricolari.

www.fondazioneCRT.it



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 soluzioni 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 Associazioni 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](https://creativecommons.org/licenses/by/3.0/it/).

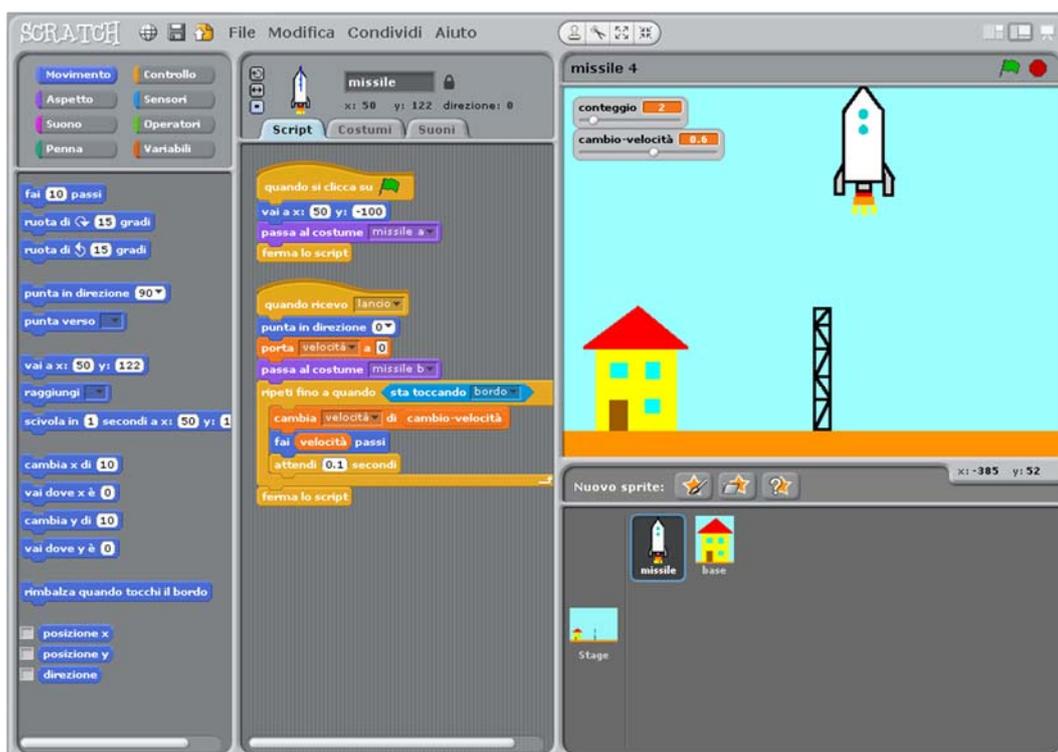
Scratch: la programmazione per gioco

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, insieme 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 fundamentalmente 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.

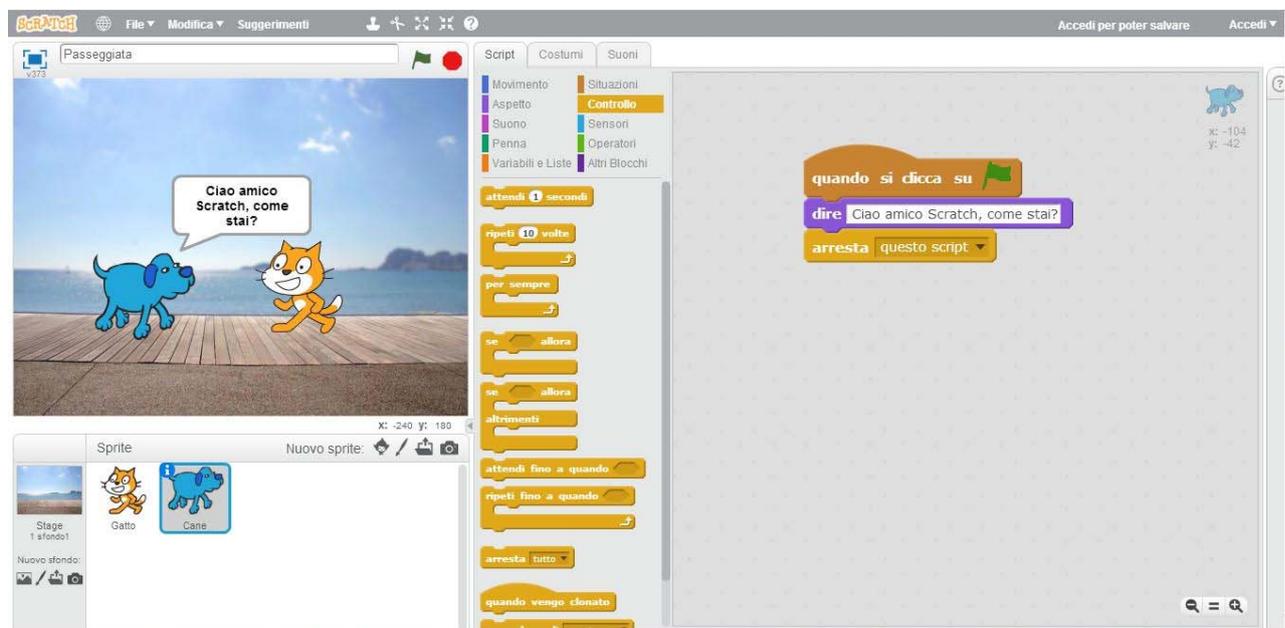
Lo **sprite** è l'oggetto posto nello stage a cui vengono associati uno o più script per farlo muovere, interagire, calcolare, giocare, e altro ancora. Oltre allo sprite raffigurante il gatto, esistono molti altri sprite già disponibili nel sito, oppure è possibile crearne dei nuovi attraverso un apposito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

Input-output

Il programma richiede in input il nome dell'utente ed attende una sua risposta che viene poi fornita in output facendola precedere dalla stringa "Ciao " unita a risposta attraverso il blocco *unione di...e...* *Risposta* è una variabile d'appoggio predefinita che contiene l'ultimo input effettuato.



Input-output con variabili

Il programma richiede in input prima il cognome dell'utente ed attende una sua risposta che viene poi assegnata alla variabile Cognome (creata attraverso il bottone *Nuova variabile*) e poi il nome dell'utente che viene assegnato alla variabile Nome (creata anch'essa attraverso il bottone *Nuova variabile*). Viene infine fornito in output l'unione della stringa "Ciao" con l'unione delle variabili Cognome e Nome.



Sequenza

Il programma richiede in input due addendi da sommare e quindi assegna alla variabile *Somma* la somma delle due variabili. Per eseguire l'operazione di somma viene usato il blocco *Operatore +*. Viene infine fornito in output il risultato richiesto.



Selezione se...

Il programma richiede in input il minuendo e il sottraendo e ne esegue la differenza solo se il primo è maggiore del secondo, per poi stamparne il risultato. Per effettuare il controllo viene usato il blocco *se...* in cui viene inserita la condizione realizzata attraverso il blocco *Operatore >*. Per eseguire l'operazione di sottrazione viene usato il blocco *Operatore -*.



Selezione se... altrimenti...

Il programma richiede in input due valori e ne esegue la differenza se il primo è maggiore del secondo altrimenti ne esegue la somma. Per effettuare il controllo viene usato il blocco *se... altrimenti..* in cui viene inserita la condizione realizzata attraverso il blocco *Operatore >*. Viene infine fornito in output il risultato richiesto.



Iterazione *ripeti fino a quando...*

Il programma richiede in input un valore e lo accetta solo se è positivo. Mediante il blocco *ripeti fino a quando...* si implementa un ciclo (da cui esce quando la condizione è vera) in cui viene inserita la condizione realizzata attraverso il blocco *Operatore >*.



Iterazione ripeti... volte

Il programma esegue per un numero definito di volte la richiesta di un input e quindi lo somma ai valori precedenti. Mediante il blocco *ripeti... volte* si implementa un ciclo col contatore che esegue il nucleo per un numero definito di volte.



Perimetro e area quadrato

Il programma calcola il perimetro e l'area di un quadrato dopo aver richiesto in input la lunghezza del suo lato. Viene usata una iterazione *ripeti fino a quando...* per il controllo dell'input.

```
quando si clicca su [bandierina]
  porta Lato a 0
  ripeti fino a quando [Lato > 0]
    chiedi [Inserisci la lunghezza del lato di un quadrato] e attendi
    porta Lato a [risposta]
  fine
  porta Perimetro a [Lato * 4]
  dire [unione di Il perimetro del quadrato è e Perimetro] per 2 secondi
  porta Area a [Lato * Lato]
  dire [unione di L'area del quadrato è e Area]
  ferma lo script
```

Area del triangolo

Il programma calcola l'area di un triangolo dopo aver richiesto in input la lunghezza della sua base e della sua altezza. Vengono usate due iterazioni *ripeti fino a quando...* per il controllo degli input e due *Operatori * e /* per calcolo dell'area.



Calcolo sconto

Il programma calcola lo sconto ed il prezzo scontato dopo aver richiesto in input il prezzo di partenza e la percentuale di sconto da applicare. Mediante l'*Operatore e* è possibile implementare l'operatore logico AND.

```

quando si clicca su
  porta Prezzo a 0
  ripeti fino a quando Prezzo > 0
  chiedi Inserisci il prezzo da scontare e attendi
  porta Prezzo a risposta
  porta %Sconto a 0
  ripeti fino a quando (%Sconto > 0) e (%Sconto < 100)
  chiedi Inserisci la % di sconto e attendi
  porta %Sconto a risposta
  porta Sconto a Prezzo * %Sconto / 100
  dire unione di Lo sconto è di euro e Sconto per 2 secondi
  porta Prezzo a Prezzo - Sconto
  dire unione di Il prezzo scontato è di euro e Prezzo
  ferma lo script
  
```

Calcolo sconto in base al prezzo

Il programma calcola lo sconto ed il prezzo scontato dopo aver richiesto in input il prezzo di partenza ed aver calcolato la percentuale di sconto da applicare in base al prezzo. Vengono usati due blocchi di selezione *se...altrimenti ...* annidati.



Ordinamento

Il programma ordina, in ordine alfabetico crescente, tre nomi ricevuti in input mediante una serie di blocchi di selezione *se...altrimenti...* annidati.

```

quando si clicca su [bandierina]
chiedi [Inserisci il primo nome] e attendi
porta [Nome1] a [risposta]
chiedi [Inserisci il secondo nome] e attendi
porta [Nome2] a [risposta]
chiedi [Inserisci il terzo nome] e attendi
porta [Nome3] a [risposta]
se <Nome1 < Nome2 e Nome1 < Nome3>
  se <Nome2 < Nome3>
    dire [Nome1] per 2 secondi
    dire [Nome2] per 2 secondi
    dire [Nome3] per 2 secondi
  altrimenti
    dire [Nome1] per 2 secondi
    dire [Nome3] per 2 secondi
    dire [Nome2] per 2 secondi
  altrimenti
    se <Nome2 < Nome3>
      se <Nome1 < Nome3>
        dire [Nome2] per 2 secondi
        dire [Nome1] per 2 secondi
        dire [Nome3] per 2 secondi
      altrimenti
        dire [Nome2] per 2 secondi
        dire [Nome3] per 2 secondi
        dire [Nome1] per 2 secondi
    altrimenti
      se <Nome1 < Nome2>
        dire [Nome3] per 2 secondi
        dire [Nome1] per 2 secondi
        dire [Nome2] per 2 secondi
      altrimenti
        dire [Nome3] per 2 secondi
        dire [Nome2] per 2 secondi
        dire [Nome1] per 2 secondi
ferma lo script
  
```

Calcolatrice

Il programma effettua l'operazione tra due operandi in base all'operatore selezionato mediante una serie di blocchi di selezione *se...altrimenti...* annidati.



Conta voti

Il programma conteggia il numero di voti sufficienti ed insufficienti inseriti in input sino a quando non viene inserito uno zero come fine input che viene controllato attraverso una iterazione *ripeti fino a quando...*



```
quando si clicca su [bandierina]
  porta Sufficienti a 0
  porta Insufficienti a 0
  porta Voto a 
  ripeti fino a quando [Voto = 0]
    chiedi Inserisci un voto e attendi
    porta Voto a risposta
    se [Voto > 5]
      cambia Sufficienti di 1
    altrimenti
      cambia Insufficienti di 1
  fine
  dire unione di Il numero di voti sufficienti è e Sufficienti per 2 secondi
  dire unione di Il numero di voti insufficienti è e Insufficienti
  ferma lo script
```

Indovina il numero

Il programma implementa il gioco in cui il computer genera un numero segreto e l'utente lo deve indovinare in un numero massimo di tentativi. Per generare il numero segreto si usa il blocco *numero a caso tra...e*. Il programma suggerisce all'utente se il suo tentativo è maggiore o minore del numero segreto. Nel caso l'utente raggiunga il numero massimo di tentativi senza indovinare, allora il programma dichiarerà chiuso il gioco. La ripetizione indefinita permette di ripetere il gioco sino a quando viene indovinato il numero segreto o sono finiti i numeri di tentativi a disposizione. Mediante l'Operatore *o* è possibile implementare l'operatore logico OR.

```

quando si clicca su [bandierina]
  porta numero segreto a numero a caso tra 1 e 50
  porta cont a 0
  porta tentativo a 0
  porta numero tentativi a 5
  ripeti fino a quando << tentativo = numero segreto o cont = numero tentativi >>
    chiedi Inserisci un tentativo e attendi
    porta tentativo a risposta
    cambia cont di 1
    se << tentativo = numero segreto >>
      dire Hai indovinato per 2 secondi
      dire unione di Numero tentativi e cont
    altrimenti
      se << tentativo > numero segreto >>
        dire Maggiore! per 2 secondi
      altrimenti
        dire Minore! per 2 secondi
      se << cont = numero tentativi >>
        dire Hai perso! per 2 secondi
        dire unione di Il numero segreto è e numero segreto
  ferma lo script
  
```

Media di N numeri

Il programma calcola la media di N numeri in input usando il blocco di iterazione *ripeti...* volte con cui le istruzioni che costituiscono il nucleo del ciclo vengono ripetute per N volte.



Calcolo il quadrato

Il programma calcola il quadrato di un numero N in input andando a sommare i primi N numeri dispari a partire da 1. Viene usato il blocco di iterazione *ripeti... volte* con cui le istruzioni che costituiscono il nucleo del ciclo vengono ripetute per N volte.

```
quando si clicca su [bandierina]
  porta N a 0
  ripeti fino a quando N > 0
    chiedi [Di che numero vuoi calcolare il quadrato?] e attendi
    porta N a risposta
  porta Somma a 0
  porta Dispari a 1
  ripeti N volte
    cambia Somma di Dispari
    cambia Dispari di 2
  dire [unione di Il quadrato è e Somma]
  ferma lo script
```

Trasformazione da base 10 a base 2

Il programma trasforma un numero positivo in input da base 10 a base 2 mediante il metodo delle divisioni successive.

```
quando si clicca su [bandierina]
  porta numero a 0
  ripeti fino a quando [numero > 0]
    chiedi [Dimmi un numero positivo] e attendi
    porta numero a risposta
  fine
  porta binario a ""
  porta quoziente a numero
  ripeti fino a quando [quoziente = 0]
    porta resto a [resto della divisione di quoziente diviso 2]
    porta binario a [unione di resto e binario]
    se [resto > 0]
      cambia quoziente di -1
    fine
    porta quoziente a [quoziente / 2]
  fine
  dire [binario]
  ferma lo script
```

Ricerca binaria

Il programma chiede all'utente di pensare un numero compreso tra 1 e 100, quindi applicando l'algoritmo di ricerca binaria tenta di indovinarlo. L'idea della ricerca binaria è quello di dividere a metà l'intervallo di ricerca. Chiamato N tale valore, verrà chiesto all'utente di rispondere alla domanda "Il numero che hai pensato è minore di N?". Se la risposta è affermativa si escludono dalla ricerca tutti i numeri superiore ad N, altrimenti, si escludono dalla ricerca tutti quelli inferiori ad N. Si ripete il procedimento fino a quando l'intervallo di ricerca si riduce al solo numero che era stato pensato dall'utente.

```

quando si clicca su 
  dire Pensa ad un numero compreso tra 1 e 100 per 2 secondi
  porta minimo a 1
  porta massimo a 100
  porta tentativi a 0
  ripeti fino a quando minimo = massimo
  cambia tentativi di 1
  porta numero a arrotonda (minimo + massimo) / 2
  chiedi unione di il numero che hai pensato è minore di e unione di numero e ? (s/n) e attendi
  se risposta = s o risposta = S
    porta massimo a numero - 1
  altrimenti
    porta minimo a numero
  dire minimo
  ferma lo script
  
```

Calcolatrice ad eventi

Il programma gestisce cinque processi di cui il primo associato allo sprite del gatto mentre gli altri associati agli sprite relativi ai quattro operatori. Quest'ultimi vengono attivati al verificarsi dell'evento *click* sullo sprite interessato all'operazione.

```

quando si clicca su [gatto]
chiedi Inserisci il primo numero e attendi
porta A a risposta
chiedi Inserisci il secondo numero e attendi
porta B a risposta
ferma lo script
  
```

```

quando si clicca su somma
porta Risultato a A + B
pensa Risultato
ferma lo script
  
```

```

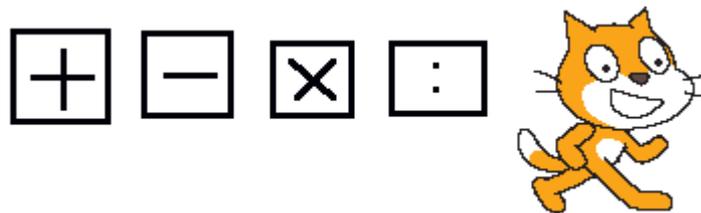
quando si clicca su Sottrai
porta Risultato a A - B
pensa Risultato
ferma lo script
  
```

```

quando si clicca su Moltiplica
porta Risultato a A * B
pensa Risultato
ferma lo script
  
```

```

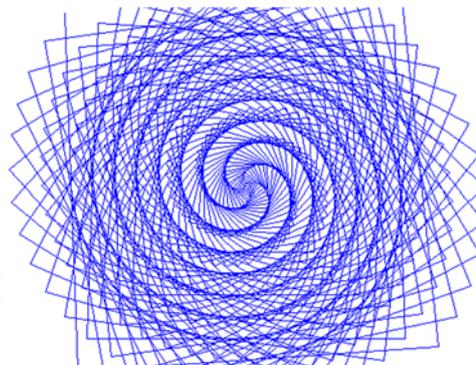
quando si clicca su Dividi
porta Risultato a A / B
pensa Risultato
ferma lo script
  
```



Spirograph

Lo *spirograph* è uno strumento composto da un insieme di ruote, anelli e barre tutti dotati di denti in grado di scorrere tra loro. Con questo strumento è possibile produrre dei disegni curiosi. Una versione digitale di tale strumento è possibile crearla utilizzando Scratch. È sufficiente posizionare lo sprite al centro dello schermo e progettare un ciclo che ripete per N volte un numero di passi crescente al termine del quale effettua una rotazione di un determinato numero di gradi.

```
quando si clicca su [bandierina]
  penna su
  pulisci
  vai a x: 0 y: 0
  penna giù
  porta passi a 0
  ripeti 360 volte
    fai passi passi
    ruota di 92 gradi
    cambia passi di 1
  ferma lo script
```



Testa o croce

Il programma simula il lancio di una monetina. Cliccando sulla monetina viene generato un numero casuale che corrisponderà al numero di rotazioni che la monetina effettuerà durante il lancio. Al costume numero 1 viene associato l'immagine della testa e alla numero 2 quella della croce.

```

quando si clicca su moneta
  porta giri a numero a caso tra 5 e 15
  ripeti giri volte
    passa al costume seguente
    attendi 0.1 secondi
  se numero costume = 1
    cambia testa di 1
  altrimenti
    cambia croce di 1
  ferma lo script
  
```

```

quando si clicca su
  porta croce a 0
  porta testa a 0
  porta numero lanci a 0
  ferma lo script
  
```



Glossario dei Blocchi

Qui di seguito è possibile visionare 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 sprite 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 valore fa riferimento alla dimensione originale dello sprite) |
| | Riporta la percentuale dell'attuale dimensione dello sprite |
| | Fa comparire lo sprite nello stage |
| | Fa scomparire lo sprite dallo stage |
| | Porta 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 dei 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 |

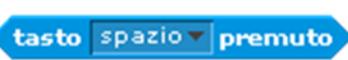
| | |
|---|---|
|  | Attende fino a quando la condizione indicata non risulta vera e solo allora esegue gli script associati |
|  | Ripete i blocchi inseriti al suo interno fino a quando la condizione continua ad essere vera |
|  | Ferma l'esecuzione dei blocchi ad esso collegati |
|  | Ferma l'esecuzione di tutti i blocchi del progetto |

Blocchi penna

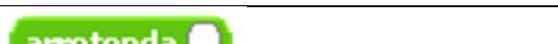
| BLOCCO | DESCRIZIONE |
|---|--|
|  | Cancella dallo stage tutto ciò che è stato disegnato con la penna e tutti i timbri |
|  | Appoggia la penna sullo stage: lo sprite mentre si sposta traccia una linea |
|  | Alza la penna dallo stage: lo sprite può spostarsi senza tracciare alcuna linea |
|  | Cambia 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 |
|  | Usa una penna delle dimensioni indicate |
|  | Riproduce l'immagine dello sprite sullo stage |

Blocco sensori

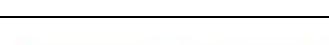
| BLOCCO | DESCRIZIONE |
|---|--|
|  | La condizione è vera se lo sprite sta toccando il puntatore del mouse, un altro sprite o il bordo dello stage. |
|  | La condizione è vera se lo sprite sta toccando il colore indicato (il colore viene scelto utilizzando il contagocce) |

| | |
|---|---|
|  | 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 0 al valore specificato |
|  | Riporta il risultato della funzione selezionata (abs, sqtr, sin, cos, tan, asin, acos, atan, ln, 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 |
|  | Apri un visualizzatore indicante il valore della variabile sullo stage |
|  | Nasconde il visualizzatore |
|  | Riporta tutti 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..... | 10 |
| Scratch versione 1.4 | 10 |
| Scratch versione 2.0..... | 12 |
| Input-output | 14 |
| Input-output con variabili | 15 |
| Sequenza | 16 |
| Selezione se..... | 17 |
| Selezione se... altrimenti..... | 18 |
| Iterazione ripeti fino a quando..... | 19 |
| Iterazione ripeti... volte | 20 |
| Perimetro e area quadrato..... | 21 |
| Area del triangolo | 22 |
| Calcolo sconto | 23 |
| Calcolo sconto in base al prezzo | 24 |
| Ordinamento..... | 25 |
| Calcolatrice | 26 |
| Conta voti..... | 27 |
| Indovina il numero | 28 |
| Media di N numeri..... | 29 |
| Calcolo il quadrato..... | 30 |
| Trasformazione da base 10 a base 2 | 31 |
| Ricerca binaria..... | 32 |
| Calcolatrice ad eventi..... | 33 |
| Spirograph | 34 |
| Testa o croce | 35 |
| Glossario dei Blocchi | 37 |

SCRATCH

Guida per i docenti



Quest'opera è distribuita con [licenza Creative Commons Attribuzione 3.0 Italia](https://creativecommons.org/licenses/by/3.0/it/).

