

Programmazione con Snap4Arduino

Paola Giannini

Informazioni preliminari

- Accedete a

<https://orienta.dir.uniupo.it/course/view.php?id=43>

Con le vostre credenziali Facebook o gmail

Pwd: arduino2017

Trovate questi lucidi e un contenitore in cui salvare i vostri programmi.

Snap

- Ambiente di programmazione, con insieme di strumenti che aiutano nell'attività di programmazione:
 - Editor (in questo caso visuale, anche la sintassi è visuale)
 - Accesso alle librerie (vedrete l'estensione per Arduino)
 - Possibilità di eseguire parti di codice (script)
- Organizzazione del codice:
 - Script associati con Sprites e Sfondi
 - Multithreading (più script in esecuzione allo stesso tempo)
 - Attivazione da eventi (tastiera, sprite entra in contatto con un colore di sfondo, bandiera, ec.....). Possibilità di generare eventi (broadcast o mirati ad uno Sprite).
 - Interazione simile a una Graphical User Interface (GUI)

Per fare programmi in Snap

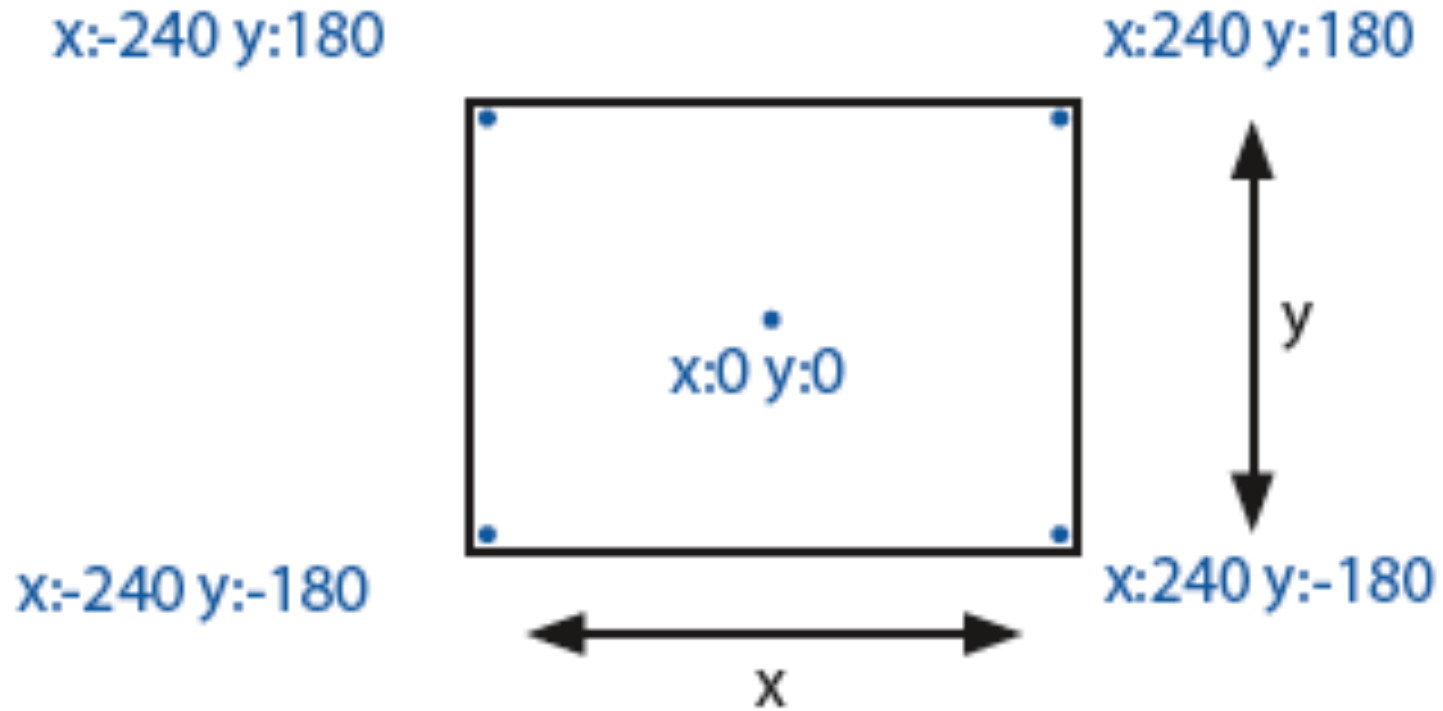
<https://snap.berkeley.edu/snap/snap.html>

L'ambiente di programmazione

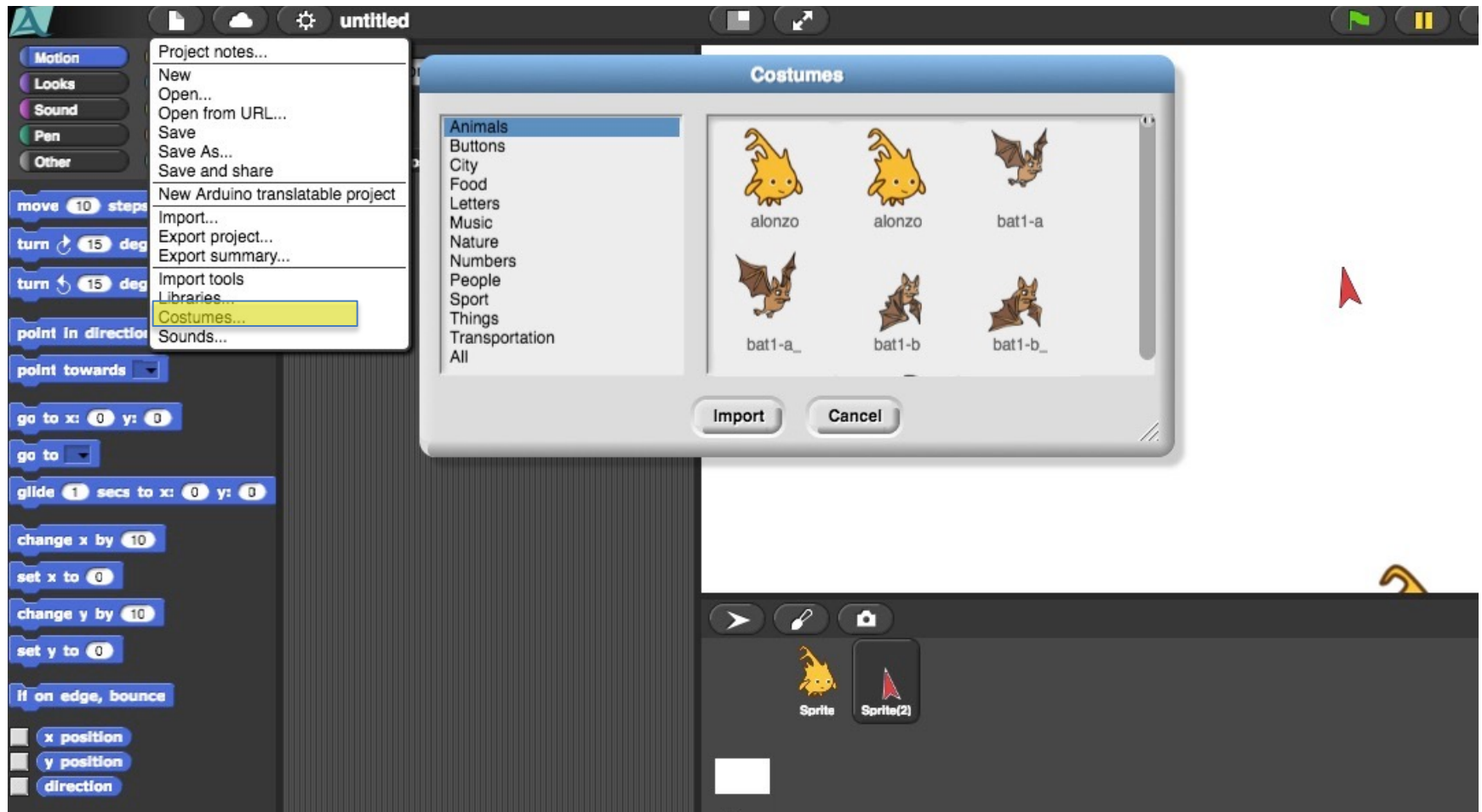
The image shows the Snap4Arduino programming environment. The interface is divided into several main sections:

- Area da cui si selezionano istruzioni:** Located on the left side, it contains a vertical menu of categories such as Movimento, Controllo, Aspetto, Sensori, Suono, Operatori, Penna, Variabili, and Altro. Below this menu is a large area where various code blocks are displayed, including 'fal 10 passi', 'ruota di 15 gradi', 'punta in direzione 90', and 'scivola in 1 secondi a x: 0 y: 0'.
- Area di scripting:** The central workspace where code blocks are assembled. It features a 'Sprite' panel at the top with a 'trascinabile' checkbox. Below it are tabs for 'Script', 'Costumi', and 'Suoni'. A script is currently visible, starting with 'when clicked', followed by 'pen down', a 'repeat 4' loop containing 'move 100 steps' and 'turn 90 degrees'.
- Pannello degli sprite:** Located at the bottom right, it shows a 'Sprite' icon and a 'Stage' icon.
- Stage:** The main canvas area on the right, currently blank, where the visual output of the program is rendered.

Lo stage



Area Sprite/Stage (può essere programmato anche lo stage)



Associati con Sprite/Stage ci sono le aree: Script, Costumi, Suoni



Elementi fondamentali di un linguaggio di programmazione

- Snap e' un linguaggio "visuale" imperativo, la computazione e' fatta eseguendo azioni su uno stato). I componenti di base sono:
 - Le istruzioni (azioni possibili)
 - Le espressioni (calcoli necessari per ottenere valori)
 - Le variabili (i contenitori di valori che rappresentano lo stato della computazione e che possono essere modificati)

Istruzioni elementari (fanno qualcosa che modifica lo stato del programma)

- In Scratch le Istruzioni si chiamano Blocchi in aggiunta a leggere/scrivere, definire e/o modificare variabili abbiamo istruzioni che agiscono sugli sprite:



- Forma caratteristica delle istruzioni, che possono essere messe in sequenza (il ; del C, C++, Java)

Istruzioni condizionali

- Condizionali: if then e if then else



- Attivazione basata su eventi



Iterazione

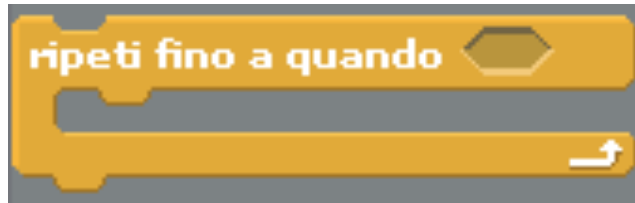
- Ripetizione (infinita):



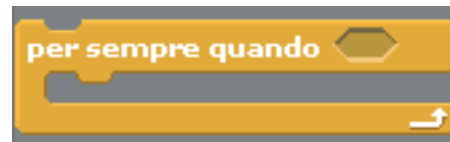
- For:



- Ripetizione condizionata: se la condizione è vera abbandona la ripetizione altrimenti esegue le istruzioni del ciclo:



- Attesa condizionale e ripetizione attesa:



Espressioni (calcolare un valore)

- Le espressioni ritornano un valore, ad esempio nel blocco Fai 10 passi o Attendi fino a quando si possono inserire espressioni:

tasto del mouse premuto



ritornano vero o falso (booleane)

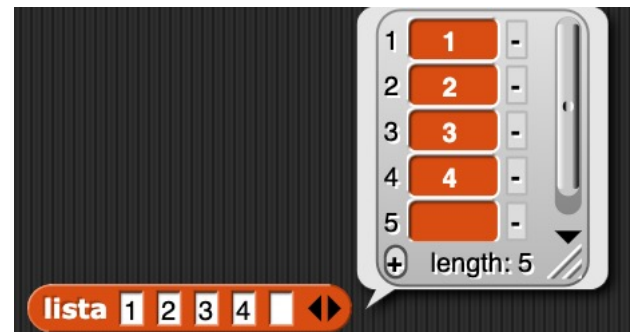
posizione x



ritornano un numero

Variabili

- Locali allo Sprite o globali a tutti gli Sprite
- Possono denotare valori numerici, stringhe (operazione di concatenazione, ecc..) e/o
- Liste (aggiungere, rimuovere, cercare)



Multi-Threading

- Thread: gruppo di istruzioni che può essere eseguito in parallelo ad altri thread.
- In C, C++ si programmano usando le primitive del sottostante sistema operativo, altri linguaggi quali Java, C# ecc.. sono implementati (Jvm o .Net) attraverso macchine virtuali multithreaded

Multi-threading in Scratch

- In Scratch i Thread si chiamano Script iniziano con un blocco arrotondato, ad esempio



si attiva quando si preme la bandiera verde (in genere inizio del programma).

Se ci sono più Script (Thread) con questo blocco di inizio, vengono attivati tutti quando si clicca sulla bandiera verde.

Eventi

- Gli Script comunicano fra loro:
 - generando eventi e
 - rispondendo ad eventi (handling)
- Comunicazione inter/intra Sprite:



Esercizio 1: Realizziamo un gioco

Gioco: Dobbiamo indovinare il numero che uno sprite ha pensato!

Come realizziamo questo gioco?

Algoritmo:

- 1) Lo sprite "pensa" un numero (diciamo fra 0 e 30)
- 2) Lo sprite ci "dice" di dirgli quale numero crediamo abbia pensato e noi gli "rispondiamo" un numero
- 3) Lui ci "dice" se il nostro numero e'
 - maggiore,
 - minore o
 - ugualeal numero che ha pensato
- 4) Se e' uguale abbiamo vinto altrimenti ritorniamo al passo 2)

(Un po' più complicato: Prima di pensare il numero lo sprite ci chiede un valore minimo e un massimo e pensa un numero in quell'intervallo)

Implementiamo l'algoritmo in snap

“pensa” => genera **un numero a caso** fra 0 e 30 e lo assegna ad un variabile **numero** (per ricordarselo!).

Le variabili sono la “memoria” del programma! Ci sono istruzioni **porta, cambia,..** per modificare le variabili

“chiede” => c'e' un'istruzione, **chiedi...**, che permette allo sprite di scrivere sullo stage un messaggio e crea anche un campo nel quale noi possiamo scrivere. Quando digitiamo un “a capo” quello che abbiamo scritto viene letto e assegnato alla variabile **risposta** (questa e' definite da Snap e la troviamo nei sensori)

“dice” => c'e' anche un'istruzione, **dire...**, che permette allo sprite di scrivere sullo stage un messaggio

Ripetizione e selezione

Le istruzioni 2, 3, 4 vengono ripetute.

Ci sono vari tipi di ripetizione.

Noi useremo **per sempre** perche' se voi continuate a dare la stessa risposta e questo non e' il numero il ciclo potrebbe andare avanti all'infinito "e oltre...."

Pero' se indovinate volete bloccare tutto. Per questo c'e' **ferma tutto**

In dipendenza del fatto che la risposta sia maggiore o minore o uguale dovete eseguire istruzioni diverse, per questo ci sono i condizionali, **se** e **se altrimenti**.

Esercizio 1 (lo script)

Abbiamo 2 variabili

numero (il numero da indovinare)

risposta (il numero che voi avete proposto)

Porta **numero** a numero casuale da 0 a 30

Ripeti per sempre:

chiedi un numero

se **risposta = numero**,

allora dire INDOVINATO e blocca tutto

se **risposta > numero**,

allora dire TROPPO ALTO

se **risposta < numero**,

allora dire TROPPO BASSO

Osservare e bloccare l'esecuzione degli Script

- Potete mettere in pausa l'esecuzione di uno script con il bottone centrale di



- Oppure controllare l'esecuzione delle istruzioni cliccando sui passi



- Viene "illuminata" l'istruzione eseguita

Esercizio 2: uno sprite che indovina il nostro numero

- Ci dice di pensare un numero fra 0 e 30 e poi inizia a dirci dei numeri, e ora siamo noi a dover dare l'informazione se quello che lui ci ha detto
 - È troppo alto
 - È troppo basso
 - È giusto
- Lo fa chiedendoci di inserire
 - 0 se il numero che ci ha detto è più piccolo di quello che abbiamo pensato
 - 1 se è uguale
 - 2 se è più grande
- Nel secondo caso dopo essersi congratulato con se stesso finisce il programma.
- Negli altri casi ritorna a proporci un numero
- (Potremo anche far sì che lo sprite capisca e ci dica se abbiamo BARATO, cioè ad un certo punto abbiamo dato una risposta sbagliata?)

Caratteristiche di Snap (non presenti in Scratch!)

- 1) Definizione di NUOVI Blocchi Istruzione (che possono essere anche parametrizzati)
- 2) Definizione di NUOVE Espressioni (che possono essere anche parametrizzate)

Esempio: scrivere un programma che disegna un quadrato a partire da una posizione x , y dello stage

Poi definiamo un blocco "quadrato".

Parametizziamolo (rispetto al lato del quadrato e alla posizione).

Poi facciamo un ciclo in cui lo chiamiamo 20 volte modificando lato e/o coordinate x e y .