

# 177. Lezioni di scripting in LSL a Scriptlandia

Rosa Marincola  
rosamarincola@virgilio.it

## Premessa

Scriptlandia è stata creata il 7 dicembre 2012 ed è costituita da un piccolo arcipelago nel mondo virtuale 3D **edMondo**. È una realtà finalizzata a percorsi didattici sperimentali basati su applicazioni d'intelligenza artificiale che il MIUR e l'INDIRE conducono con docenti di alcune scuole d'Italia (<http://www.scuola-digitale.it/ed-mondo/progetto/info/>). Tra questi, la classe III A Sistemi Informativi Aziendali, dell'indirizzo Tecnico Economico dell'IIS "A. Guarasci" di Rogliano (Cs) sotto la mia guida in qualità di docente d'informatica, ha avviato un percorso innovativo sull'uso delle tecnologie in classe. In questo spazio protetto, gli utenti autorizzati e registrati, accedono mediante avatar che possono camminare, volare e teletrasportarsi da una regione all'altra. All'atto della consegna, **Scriptlandia** era solo una landa desolata e informe, ma già in poco tempo si è notevolmente evoluta e arricchita. L'arcipelago è stato "terraformato" su modello di tre isole delle Galápagos:

**Virtual Lab:** dove gli studenti svolgono le loro esercitazioni e sperimentazioni sotto la guida della loro docente d'informatica. In questo spazio la classe svolge attività di building 3D, ma soprattutto di programmazione (scripting) in LSL (Linden Scripting Language) in presenza e, talvolta a distanza.

**Agorà:** un luogo d'incontro con un anfiteatro in posizione centrale dove si può discutere, condividere esperienze, organizzare attività ed eventi, esporre lavori della classe di tipo interdisciplinare in mostre temporanee e permanenti.

**Library:** uno spazio dedicato a un percorso didattico d'informatica con materiali fruibili da tutti i visitatori. Si tratta di una biblioteca virtuale 3D molto insolita: è dominata dall'*Albero della cultura* (per gentile concessione del builder Giorgio Lo Forti alias Beduino Ruben). Esso, pur piantato in terreni infruttuosi (dominati dall'arroganza, dalla presunzione, dall'ignoranza, dalla superstizione, ecc.), se alimentato da elementi positivi (quali la costanza, la volontà, la ricerca, ecc.) riesce a crescere rigoglioso e a produrre le arti, la matematica, l'informatica e le altre scienze. Dal terreno intorno all'albero nascono funghi contenenti lezioni di script in LSL e fiori che rilasciano come contenuto dei problemi a carattere logico-matematico e algoritmico. I libri sono sostituiti da pannelli contenenti slides, links a siti esterni, timeline. Le mappe mentali e concettuali sono realizzate in 3D. Gli argomenti sviluppati in classe (secondo i programmi ministeriali) prendono forma nella realtà virtuale, come in un cantiere *in fieri*. Essi trattano elementi di storia dell'informatica, di programmazione, dell'architettura di un calcolatore, d'ingegneria del software, di reti di computer, di database, d'intelligenza artificiale, con un ampio spazio dedicato al Web.

Questa mia avventura nei mondi virtuali è iniziata nell'ottobre 2012, grazie al ricercatore INDIRE Andrea Benassi che ha presentato EdMondo in un seminario per tutor nei piani nazionali (da diversi anni svolgo l'attività di tutor in vari progetti: M@t.abel, Cl@ssi 2.0, Innovadidattica, DIDATEC Avanzato).

La mia formazione iniziale è stata assistita da un mentore d'eccezione: Salahzar Stenvaag a cui vanno i miei ringraziamenti per tutti i materiali online, le lezioni, i suggerimenti e i video (<https://www.youtube.com/user/salahzarstenvaag/videos?view=0>). Ho deciso così di condividere in rete questa serie di lezioni che ho preparato per i miei studenti sull'LSL

([http://wiki.secondlife.com/wiki/LSL\\_Portal](http://wiki.secondlife.com/wiki/LSL_Portal)), nell'auspicio che possano essere utili anche ad altre persone.



Lezione di Salahzar

### Lezione 1: far apparire un messaggio in chat e una scritta su un prim

//Primo script con la terza A istituto tecnico economico S.I.A. IIS "A. Guarasci" di Rogliano (Cosenza) in LSL (Linden Scripting Language)  
 //tutto quello che è preceduto da doppio/ è un commento, quindi non sarà eseguito

//rezzare un cubo, cambiare il nome all'oggetto e la texture, nella scheda contenuto cliccare su Nuovo Script e aprire la finestra  
 //attenzione l'LSL è case sensitive, distingue le maiuscole dalle minuscole default

```
{
//siamo nello stato iniziale di esecuzione dello script, tutte le parentesi
che si aprono, si devono chiudere
state_entry()
{
//quando si salva o si ripristina lo script il prim (cubo o quello che
avete rezzato) comunica (dice) nella chat (canale 0) la frase scritta tra
apici
llSay(0, "Primo script di Rosa con la classe terza");
//questa funzione crea una scritta fluttuante sul prim di colore blu
llSetText("Marincola Rosa", <0,0,1>,1);
}
touch_start(integer count)
{
//al tocco, cioè quando si clicca sul prim, sulla chat compare la scritta
tra apici
llSay(0, "Hello World!");
}
}
```

//**Esercizio:** rezza una sfera, cambia nome e texture a piacere. Fai in modo che nello stato iniziale dica in chat "Benvenuto a Scriptlandia", che sulla sfera ci sia la scritto "Esercizio di script di ..." (metti il tuo nome) e quando si clicca sulla sfera venga mostrato in chat un messaggio di saluto.

## Lezione 2: Colorare in modo random un prim

```
//Lezione commentata tratta dal Blog di Salahzar
//http://alicorsi.wordpress.com/author/salahzar/
//RGB è il nome di un modello basato sui tre colori rosso (Red), verde
(Green) e blu (Blue).
//Con questo script imparerai a dare un colore di default ad un prim e a
cambiarlo in modo casuale ad ogni tocco.
//Rezza un cubo
default
{
    state_entry()
    {
//messaggio in chat
        llSay(0, "Clicca e cambio di colore");
//nello stato iniziale il prim è blu, prova a modificare le 3 componenti
dei colori
        llSetColor(<0,0,1>,ALL_SIDES);
//scritta flutuante sul prim, il simbolo \n serve per andare a capo
llSetText("Colori random \n se mi tocchi cambio colore",<0,0,1>,1 );
    }
    touch_start(integer total_number)
    {
// ad ogni tocco, cambia il colore (in modo casuale) a tutte le facce del
prim
        llSetColor(<llFrاند(1),llFrاند(1),llFrاند(1)>, ALL_SIDES);
// la funzione llFrاند crea un numero random da zero al numero contenuto
nel suo parametro.
// quindi llFrاند(1) restituisce un numero casuale tra zero e 1.
//Prova a sostituire ALL_SIDES con un numero intero da 1 a 6 (cioè il
numero di una faccia del cubo), cosa succede?
    }
}
//Esercizio: rezza una piramide, riutilizza e adatta lo script precedente in modo da eliminare la
scritta in chat, modifica la scritta flutuante in modo che sia su 3 righe e fai in modo che ad ogni
tocco, ciascuna faccia sia colorata in modo casuale.
//Suggerimento: la funzione llRound(...) tronca la parte decimale di un numero quindi
llRound(2.7) restituisce 2
// sostituisci ALL_SIDES con llRound(llFrاند(5.9))
//avrà un colore casuale per ciascuna delle 5 facce della piramide.
```



Lezione per i docenti in Welcome area

### Lezione 3: Come dare un URL per far visitare siti, video e altre risorse nel web browser

//Rezza un prim, e seleziona una texture (meglio un'immagine del sito che vuoi far visitare o crea una scritta e importala nell'inventary.

```
default
{
    state_entry()
    {
        llSetText("Cliccami per visitare \n sito della III
A ITC \n di Rogliano",<1,1,1>,1);
    }
    touch_start(integer count)
    {
        //questa istruzione apre una finestra con cui chiunque clicca
        llDetectedKey(0)
        //può visitare il sito di cui si è scritto l'URL
        llLoadURL(llDetectedKey(0),"Sito della III A ITC di
Rogliano","http://iiaitc.webs.com");
    }
}
```

//**Esercizio:** ricerca un buon sito sull'architettura di un computer (possibilmente con un simulatore per l'assemblaggio di un PC.

//Crea o trova un'immagine pertinente, poi rezza un cartello che permetta di visitare il sito.



Prime costruzioni in edmondo

## Lezione 4: Come rilasciare un testo scritto in una notecard

//Rezza un prim, seleziona una texture (meglio un'immagine importata nell'inventario).

//Crea una notecard nel tuo inventario e scrivi dentro il messaggio che vuoi che gli altri leggano.

//Trascina la notecard nell'inventario del prim e crea un nuovo script

//Risorsa consigliata

<http://secondtechnologies.blogspot.it/2010/09/lldetectedname-e-lldetectedkey.html>

```
default
{
state_entry()
{
    llSetText("Clicca per avere \n una notecard",<1,1,1>,1);
}
    touch_start(integer total_number)
    {
// Prende la notecard presente nell'inventario del prim
(llGetInventoryName) e la cede (llGiveInventory) a chi clicca
(llDetectedKey(0))
        llGiveInventory(llDetectedKey(0),
llGetInventoryName(INVENTORY_NOTECARD, 0));
    }
}
```

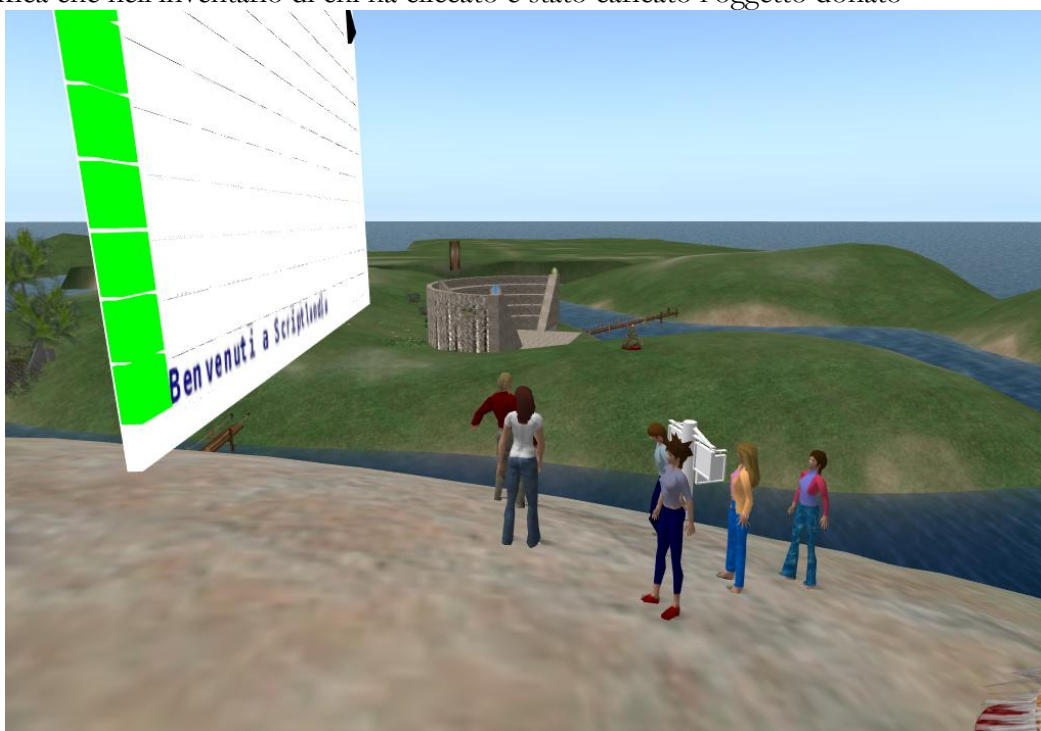
//**Esercizio:** invece di una notecard inserisci nell'inventario di un prim un oggetto di nome ad esempio “Margherita” (trascinalo dal tuo inventario nel contenuto del prim).

//Stai attento a scrivere esattamente il nome rispettando maiuscole-minuscole ed eventuali spazi, modifica la scritta sul prim,

//sostituisci la riga che dà l'oggetto dall'inventario

// llGiveInventory(llDetectedKey(0),"Margherita");

//verifica che nell'inventario di chi ha cliccato è stato caricato l'oggetto donato



Lezione su Virtual Lab

## Lezione 5: Come rilasciare un oggetto nell'inventory

//Questo script è la soluzione del precedente esercizio.

```
default
{
    state_entry()
    {
        llSetText("Clicca per avere \n un fiore\n
nell'inventario",<1,1,1>,1);
    }
    touch_start(integer total_number)
    {
        llGiveInventory(llDetectedKey(0),"Margherita");
    }
}
```

## Come rilasciare un oggetto ad un avatar particolare o ai membri di un gruppo

//Modifichiamo il codice in modo che se clicca un avatar particolare venga dato l'oggetto, altrimenti, ci sia un messaggio in chat

```
default
{
    state_entry()
    {
        llSetText("Regalo per XXX",<1,1,1>,1);
    }
    touch_start(integer total_number)
    {
        key avatar= llDetectedKey(0);
        //clicca sull'avatar a cui vuoi fare il regalo, poi su profilo e su copy
        key, copia il codice identificativo dell'avatar
        if (avatar=="26ecc3a5-9243-470e-b8d9-4afcacdecf59")
        {
            llGiveInventory(avatar,"Margherita");
        }
        else
        {
            llSay(0, "Mi dispiace, ma questo regalo
è solo per XXX!");
        }
    }
}
```

//Facciamo un'altra variante per dare un oggetto di nome "Margherita" solo ai membri di un gruppo.

```
default
{
    state_entry()
    {
        llSetText("Regalo per il mio gruppo",<1,1,1>,1)
    }
    touch_start(integer total_number)
    {
        key avatar= llDetectedKey(0);
        string nomeavatar=llDetectedName(0);
        llSay(0, "Mi ha toccato " + nomeavatar);
        if (llSameGroup(avatar))
```

```

        {
            llGiveInventory/avatar,"Margherita");
        }
else
    {
llSay(0, "Mi dispiace, ma non fai parte del mio gruppo!");
    }
}
}

```

## Lezione 6: Funzioni matematiche

//Per comprendere questa lezione devi conoscere: i tipi di dati, le variabili locali e globali. Di seguito segnalo alcune risorse.

[//http://wiki.secondlife.com/wiki/User:Michel\\_Lemmon/Script\\_Propedeutico#variabili\\_globali\\_e\\_locales](http://wiki.secondlife.com/wiki/User:Michel_Lemmon/Script_Propedeutico#variabili_globali_e_locales)

//Funzioni matematiche: [//http://wiki.secondlife.com/wiki/Category:LSL\\_Math](http://wiki.secondlife.com/wiki/Category:LSL_Math)

//Cos'è il Lag:

[//http://virtualworldsmagazine.wordpress.com/2012/09/27/come-sopravvivere-al-lag-tra-verita-e-leggende-metropolitane/](http://virtualworldsmagazine.wordpress.com/2012/09/27/come-sopravvivere-al-lag-tra-verita-e-leggende-metropolitane/)

```
integer handle;
```

```
default
```

```

{
    touch_start(integer count)
    {
        llSay(0, "Digita il numero di cui vuoi calcolare la
radice quadrata");
//la variabile globale "handle" acquisisce il valore ascoltato sul canale 0
cioè in chat
handle=llListen(0,"",llDetectedKey(0),"");
    }
//ora segue l'elaborazione di ciò che è stato ascoltato sul canale
zero
    listen(integer channel,string name,key id,string str)
    {
//I listeners creano lag, in particolare sul canale 0.
//Occorre limitare il numero di listeners al minimo possibile e rimuovere
l'handle non appena possibile con la funzione seguente
        llListenRemove(handle);
//"input" è una variabile locale, è un numero reale (virgola mobile) e
acquisisce il valore digitato in chat (il tipo stringa è convertito in
numero in virgola mobile)
        float input=(float)str;
//istruzione di controllo sul valore digitato
if (input<0)
    {
        llSay(0, "Errore, il radicando deve essere maggiore o
uguale a zero");
    }
else
    {
//se il numero è >=0 calcola e visualizza la radice quadrata, i dati sono
convertiti in stringhe
        llSay(0,"La radice quadrata di
"+(string)input +" è: "+(string)llSqrt(input));
    }
}
}

```

```
}
```

//**Esercizio:** modifica il codice in modo che calcoli il logaritmo di un numero

## Lezione 7: Impariamo a usare le finestre di dialogo

//handle è una variabile globale di tipo intero  
integer handle;

```
default
{
    state_entry()
    {
        lSetText("Domanda", <1,1,0>,1);
        handle=llListen(30,"",NULL_KEY,"");
    }
touch_start(integer total_number)
{
//id è una variabile locale di tipo chiave, essa identifica chi ha cliccato
sul prim
    key id=llDetectedKey(0);
//la finestra di dialogo richiede: la chiave, un testo (facoltativo), in
[...] la lista di ciò che scriverà su ciascun pulsante (nell'esempio
seguinte 4 pulsanti) e infine il canale su cui si ha il dialogo, in questo
caso 30
    llDialog(id,"Quante macchinine di Formula 1 ha Mario se sono
tutte Ferrari meno 3, sono tutte McLaren meno 2 ed ha anche una Williams?",
["3","4","5","6"],30);
}
listen(integer channel, string name, key id, string str)
{
llListenRemove(handle);
//a seconda dell'opzione selezionata, si avrà una risposta in chat
if (str=="3" ) llSay(0,"Risposta errata");
if (str=="4" ) llSay(0,"Risposta corretta N=1+(N-3)+(N-2)");
if (str=="5" ) llSay(0,"Risposta errata");
if (str=="6" ) llSay(0,"Risposta errata");
//lo script viene resettato/ripristinato per le esecuzioni successive
llResetScript();
}
}
```

//**Esercizio:** modifica il codice precedente in modo che ci sia un quesito a scelta multipla da te inventato, con 5 opzioni:a, b, c, d, e;  
//in corrispondenza ad ogni opzione un messaggio diverso in chat

## Lezione 8: Primi movimenti lineari: facciamo muovere un prim lungo i lati di un quadrato di lato 1m nel piano XY (orizzontale)

```
default
{
    touch_start(integer total_number)
    {
//Se vogliamo leggere in chat le coordinate assolute (della regione) del
punto di partenza usiamo llGetPos() che restituisce la terna <X,Y,Z>
        llSay(0, "Ora mi trovo " + (string)llGetPos());
//La funzione llSetPos(120, 42, 56) invece posiziona il prim nel punto di
coordinate assolute indicate <120, 42, 56>
```

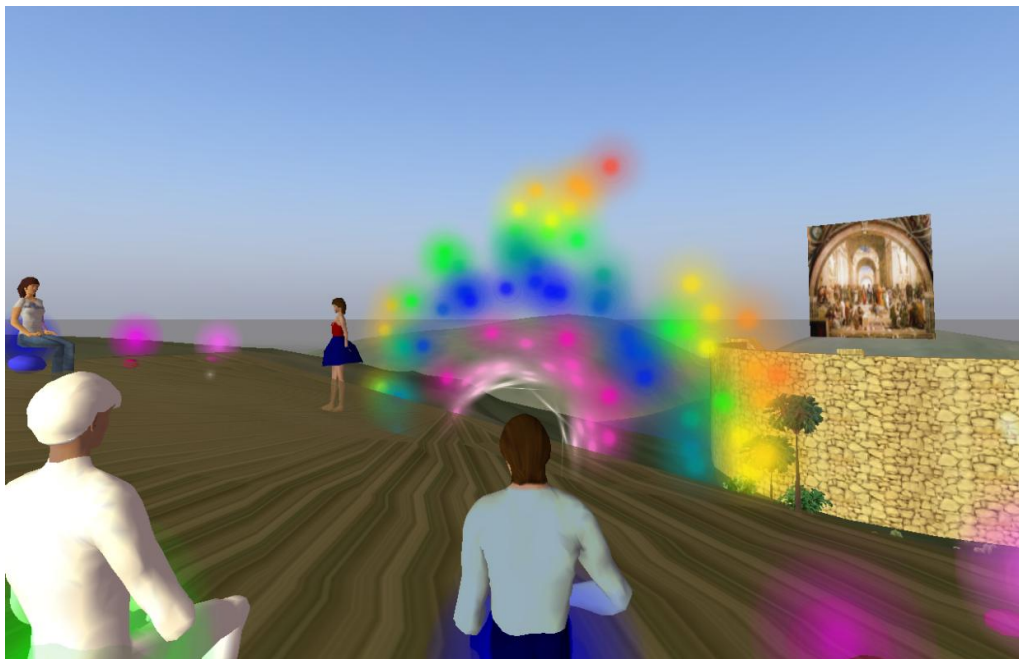


```
//nel nostro primo caso, alla sua posizione si aggiunge la terna < 0, 1, 0
> che lascia immutata X e Z e incrementa di 1 solo la Y
    llSetPos( llGetPos() + < 0, 1, 0 > );
//poi incrementa di 1 la X;
    llSetPos( llGetPos() + < 1, 0,0 > );
    llSetPos( llGetPos() + < 0, -1, 0 > );
    llSetPos( llGetPos() + < -1, 0,0 > );
}
}

//Modifichiamo il codice in modo che per 4 volte il prim descriva un
quadrato,
//ma ad ogni giro (ciclo) il lato abbia la misura del contatore i (indice)

default
{
touch_start(integer total_number)
{
integer i;
//per 4 volte (i va da 1 a 4 compreso, esegue il blocco di istruzioni in
parentesi {}), e incrementa di 1 il contatore i++ sarebbe i=i+1, quindi la
misura del lato aumenta di 1m.
    for(i=1;i<5;i++)
    {
        llSetPos( llGetPos() + < 0, i, 0 > );
        llSetPos( llGetPos() + < i, 0,0 > );
        llSetPos( llGetPos() + < 0, -i, 0 > );
//nel il terzo vertice dice le coordinate assolute del punto in cui si
trova
        llSay(0, "Ora mi trovo " + (string)llGetPos());
        llSetPos( llGetPos() + < -i, 0,0 > );
    }
}
}
```

//**Esercizio:** rezza una sfera e scrivi il codice in modo che ad ogni tocco, rimbalzi per 6 volte all'altezza costante di 0.5m.



Visione di alcuni effetti speciali

## Lezione 9: Prime rotazioni

```
//Prima di iniziare, leggi alcune pagine:
//http://virtualworldsmagazine.wordpress.com/2012/06/23/facciamoci-un-sistema-solare-in-casa-secondlife-o-opensim-esempio-di-un-progetto-astronomico/
//http://wiki.secondlife.com/wiki/LlTargetOmega

default
{
state_entry()
{
//con seguente funzione il prim ruota sempre attorno all'asse Z una volta
al secondo (vedi terna <0,0,1> ),
//la velocità di rotazione è PI rad/sec cioè 180° al secondo,
//1.0 è la forza che viene data allo spin, questo numero è un valore che ha
effetto quando il prim è fisico, e va settato diverso da 0 per i prims non
fisici).
//http://secondtechnologies.blogspot.it/2010/09/rotazione-di-un-prim-con-lltargetomega.html
//la rotazione dipende dal viewer del client, cioè non tutti vedono la
stessa rotazione

llTargetOmega(<0,0,1>,PI,1.0);
}
}

//Esercizio: modifica i parametri e osserva le diverse rotazioni.
//Crea un link-set con due prim e inserisci il codice nel prim radice, cosa osservi?
```

## Lezione 10: Alcune funzioni per gestire le rotazioni di un prim

```
//Prima devi leggere attentamente
http://wiki.secondlife.com/wiki/Rotation/it

//Con le nuove funzioni che useremo, la rotazione appare la stessa a tutti
i client

//Rotazione di 90° intorno asse Z

default
{
touch_start(integer total_number)
{
//Con questa istruzione leggiamo in chat il valore della funzione
llGetRot()
llSay(0, (string)llGetRot());
//Essa restituisce un quaternion come a quattro numeri, tre dei quali
rappresentano
//la direzione verso la quale un oggetto è rivolto ed un quarto che
rappresenta l'inclinazione sinistra o destra dell'oggetto intorno a questa
direzione.

//delta e rot sono variabili locali di tipo rotation
rotation rot = llGetRot();

//La funzione llEuler2Rot converte il vettore contenuto in <> da radianti
in una rotazione che viene assegnata alla variabile delta
//DEG_TO_RAD = 0.01745329238f; è una costante di tipo float che quando
moltiplicata per un angolo in gradi dà l'angolo in radianti.
```

```
//90 * DEG_TO_RAD converte 90° in radianti, cioè PI/2
//<0,0,90 * DEG_TO_RAD>, in questo modo ci sarà una rotazione di 90° solo
attorno all'asse Z
```

```
rotation delta = llEuler2Rot (<0,0,90 * DEG_TO_RAD>);
//Ora si ha la composizione dei due quaternioni: a quello iniziale si
applica quello che effettua la rotazione di 90° secondo l'asse Z
rotation rot = delta * rot;
//llSetRot() posiziona il prim secondo i valori calcolati della rotazione
rot
llSetRot (rot);
}
}
```

//**Esercizio:** ora combiniamo insieme ciò che è stato fatto finora per gestire alcuni movimenti base attraverso una finestra di dialogo.

//Completa lo script seguente e testalo inserendolo nel contenuto di un prim in modo che sia radice di un link-set (con un'auto)

```
rotation delta;
rotation rot;
integer handle;

default
{
state_entry()
{
llSetText("movimenti base", <1,1,0>,1);
handle=llListen(30,"",NULL_KEY,"");
}
touch_start(integer total_number)
{
key id=llDetectedKey(0);
llDialog(id,"Inizia a gestire i movimenti:X+1 significa che
incrementa di un metro la coordinata X, mentre -30°z indica una rotazione
attorno all'asse z di 30° in senso orario", ["X+1","Y+1","Z+1","X-1","Y-
1","Z-1","30°x","-30°x","30°y","-30°y","30°z","-30°z"], 30);
}
listen(integer channel, string name, key id, string str)
{
llListenRemove(handle);
rotation rot = llGetRot();
if (str=="X+1")
{ llSetPos( llGetPos() + < 1, 0,0 > );}
if (str=="30°x")
{
delta = llEuler2Rot (<30 * DEG_TO_RAD,0,0>);
rot = delta * rot;
llSetRot(rot);
}
}
//scrivi le traslazioni e le rotazioni mancanti indicate sui pulsanti della
finestra di dialogo
.....
.....
.....
llResetScript();
}
}
```

## Lezione 11: Alcuni operatori logici

//Prima di affrontare la lezione, devi conoscere gli operatori logici, sono importantissimi in programmazione:

[//http://it.wikipedia.org/wiki/Tabella\\_della\\_verit%C3%A0](http://it.wikipedia.org/wiki/Tabella_della_verit%C3%A0)

[//http://wiki.secondlife.com/wiki/LSL\\_Operators](http://wiki.secondlife.com/wiki/LSL_Operators)

//Rezza un cubo, sarà il nostro dado con cui facciamo un gioco molto semplice:

//(1)se viene generato il numero 1 oppure 5, chi ha toccato vince

//(2)se esce un numero  $2 <= n <= 4$ , il giocatore perde.

//(3)Se esce il 6, il giocatore deve ritentare

//Scriviamo la condizione 2) nella forma  $n > 1$  AND  $n > 5$ , (ma potevamo scriverlo in altro modo equivalente... prova!)

```
default
{
touch_start(integer total_number)
{
//scrivi l'istruzione per colorare le facce in modo casuale (Lezione 2)
//n è un numero casuale da 1 a 6
integer n=(integer)(llFrnd(6)+1);
//il numero estratto viene visualizzato con una scritta sul cubo
llSetText( (string)n, <1,1,1>, 1);

if (n==1 || n==5)      {llSay(0, "Hai vinto, è uscito "+ (string)n);}
//1)
if (n>1 && n<5)      {llSay(0, "Hai perso, è uscito "+(string)n);} //2)
if (n==6)             {llSay(0, "Devi ritentare, è uscito "+(string)n);} //3
}
}
```

//**Esercizio:** modifica le regole del gioco (le condizioni di vittoria o di sconfitta con gli operatori).

//Fai in modo venga estratto un numero a caso da 1 a 20, poi il cubo rimbalzi e rotei in aria un numero di volte pari al numero estratto.

//Con un programma di grafica puoi "disegnare" le sei facce del cubo e applicare le diverse texture per rendere realistica la simulazione.



Library

## Lezione 12: Istruzione ciclica col controllo in testa

```
//Rezza una sfera e applicale la texture di un pallone
//In questa lezione utilizziamo l'istruzione ciclica precondizionale
//http://it.wikipedia.org/wiki/Algoritmo\_iterativo
//http://wiki.secondlife.com/wiki/User:Michel\_Lemmon/Script\_Propedeutico

default
{
    touch_start(integer total_number)
    {
        integer n=(integer) (llFrاند(10)+1);

        //n è un numero casuale da 1 a 10
        //vogliamo che al tocco, il pallone rimbalzi da terra
        //finquando è vera la condizione che il numero n sia positivo e diverso da
        3

        vector position = llGetPos();

        while (n!=3 && n>0)
        {

            llSetPos( llGetPos() + <0, 0, 0.1*n > );
            // con il prodotto 0.1*n facciamo in modo che i balzi non siano tutti
            uguali...prova il codice e vedrai!

            llSetPos( llGetPos() + <0, 0, -0.1*n > );
            //decrementiamo n di 1 ad ogni ciclo
                n--;

        }
    }
}
```

## Istruzione ciclica col controllo in coda

```
//Se hai testato più volte lo script precedente, avrai osservato che alcune
volte la palla non eseguiva alcun balzo.
//Questo si verifica se il numero generato in modo casuale è 3, cioè quando
la condizione iniziale è falsa al tocco, il ciclo non viene eseguito.
//Per fare in modo che il blocco di istruzioni, venga eseguito almeno una
volta,
//possiamo utilizzare l'istruzione ciclica postcondizionale

default
{
    touch_start(integer total_number)
    {
        integer n=(integer) (llFrاند(10)+1);
        //n è un numero casuale da 1 a 10
        do
            {
                llSetPos( llGetPos() + <0, 0, 0.1*n > );
                llSetPos( llGetPos() + <0, 0, -0.1*n > );

            }
        n--;
    }
    while(n>0);
    //il blocco di istruzioni viene ripetuto finquando n è positivo, n ad ogni
    ciclo è decrementato di 1, quando n=0 non verrà più eseguito
}
```

}

//**Esercizio:** rezza un cilindro e riduci l'altezza in modo che sembri una moneta,  
//applica sulle due facce le texture testa e croce.  
//Riutilizza i codici precedenti e modificali per simulare il lancio  
//di una moneta che rotei un numero casuale di volte in aria.



Vista su due isole dell'arcipelago

## Conclusioni

In questa prima serie di lezioni, ho trattato tipi di dati semplici, istruzioni operative e di controllo e alcune funzioni base in LSL, col solo evento touch\_start. Le lezioni sono state pensate per studenti di 15-16 anni che studiano informatica come disciplina curricolare, dunque come applicazione della teoria anche in un ambiente virtuale per sperimentare l'animazione di oggetti 3D in linguaggio LSL.

La pubblicazione di questo lavoro vuole essere un aiuto all'apprendimento graduale e divertente per quanti hanno un primo approccio con questo linguaggio di scripting. Certo la programmazione richiede applicazione, tempo, approfondimento progressivo e tanta creatività, ma i mondi virtuali offrono più di altri ambienti, molti stimoli per affrontare un simile percorso di autoformazione a qualsiasi età!

“Il bambino che non gioca non è un bambino, ma l'adulto che non gioca ha perso per sempre il bambino che ha dentro” (Pablo Neruda).

Rosa Marincola