

Grafica

Anno Accademico 2018-2019

Esercitazione 5

Matteo Berti, Matricola 889889

27 giugno 2019

Comandi da tastiera

Tasto	Azione
q	Terminazione dell'esecuzione
r	Raytracing
t	Visualizzazione raggio
p	Hard/soft shadow (toggle)

Note: il codice è stato generato e testato utilizzando le seguenti specifiche:

- OS: Debian GNU/Linux 9 (stretch)

- IDE: CLion 2018.3.4

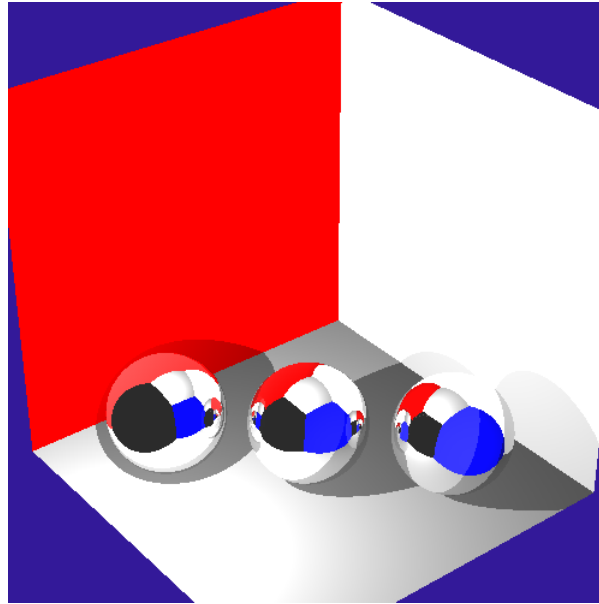
Nella configurazione dell'ambiente di esecuzione sono stati impostati gli argomenti **Program arguments:**

-size 500 500 -input ./cornell_box_reflective_spheres.obj -background_color 0.2 0.1 0.6 -num_bounces 4 e la directory contenente gli oggetti di scena **Working directory:** /path/Ray Tracing/Objs.

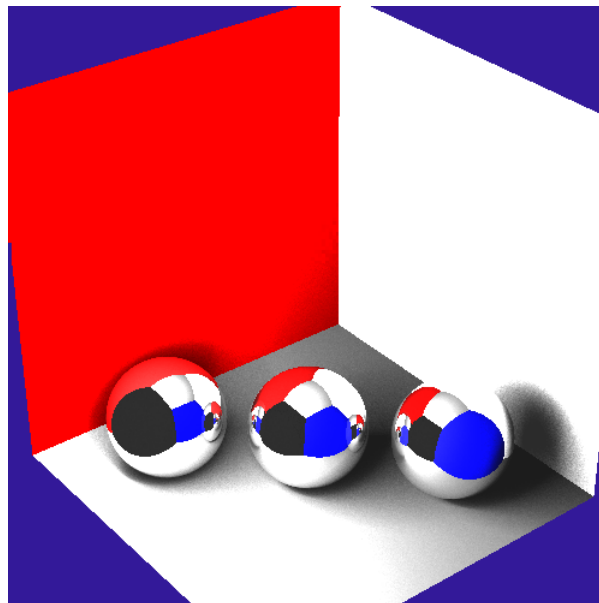
Parte I - Ray Tracing

Nel file **raytracer.cpp** sono state implementate le funzioni richieste.

- Hard shadows:** per l'hard shadow si considera la luce ad area presente come una luce puntiforme proveniente dal suo baricentro. Si calcola un raggio che dal punto analizzato arriva alla luce, ed effettuando il **CastRay** si osserva se colpisce qualche oggetto nel tragitto. In caso affermativo si calcola la distanza tra il punto e l'oggetto colpito e in base a questo viene assegnato un colore all'ombra.



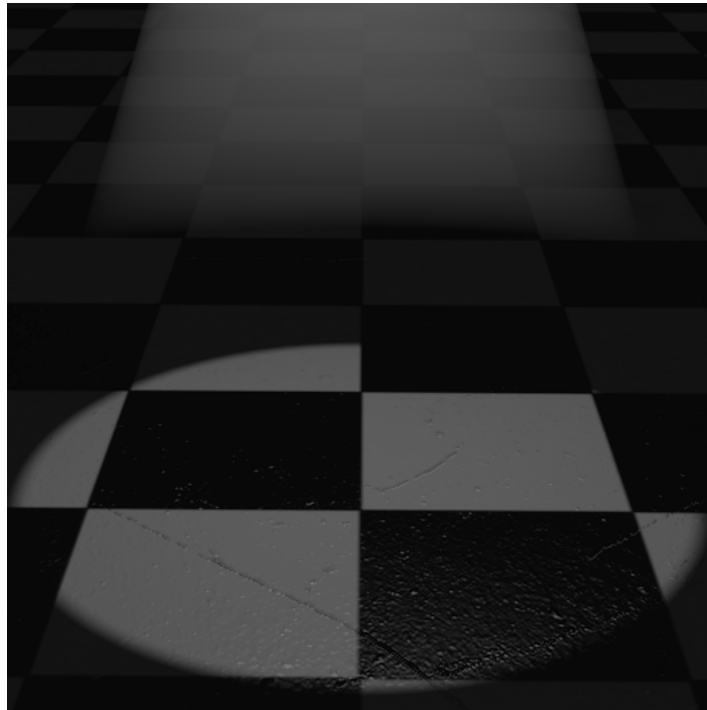
- b) **Refraction rays ricorsivi:** si calcola per prima cosa il **raggio riflesso** sul punto colpito in base alla **normale** alla superficie, da qui si genera un nuovo raggio in questa direzione e si richiama quindi ricorsivamente il **TraceRay** su questo nuovo raggio tante volte quanti sono i rimbalzi scelti dall'utente.
- c) **OPZIONALE: Soft shadows:** per la soft shadow si procede in modo simile all'hard shadow, ma invece che avere un'unica sorgente di luce puntiforme si calcola il colore di un pixel sommando i valori dell'ombra generata da sorgenti luminose provenienti da **500 punti random** all'interno della luce ad area (maggiore il numero di punti random, maggiore è la precisione della soft shadow).



Parte II - Digital Art

La scena si compone di varie **mesh** e **nurbs** di cui alcune **riflettenti** e altre in **trasparenza** con rifrazione, una spot light illumina la scena.

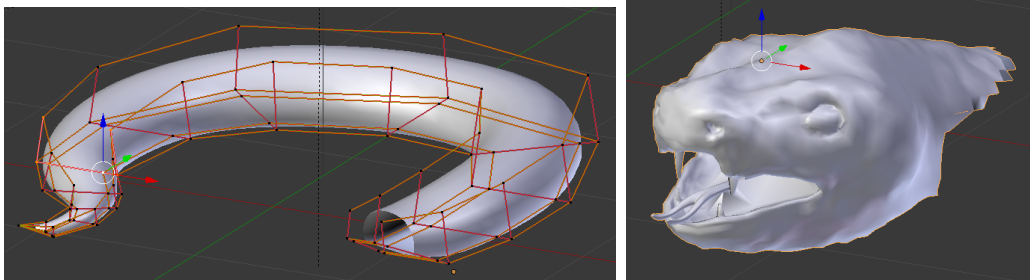
- **Pavimento:** il pavimento è costituito da un **cubo** appiattito e ridimensionato in modo opportuno. Vi sono state applicate due texture: una con il pattern di una scacchiera, e l'altra con dei graffi. La prima è stata posizionata e rifinita utilizzando **UV Unwrap** e **UV Editing**, come già visto nell'esercitazione 4. La seconda invece nella sezione Influence è stato deselezionato il colore e selezionata la normale a 0.5, per creare **bump mapping**. Infine al materiale è stata attribuita una leggera riflettività di 0.5 (questo processo è spiegato in dettaglio di seguito).



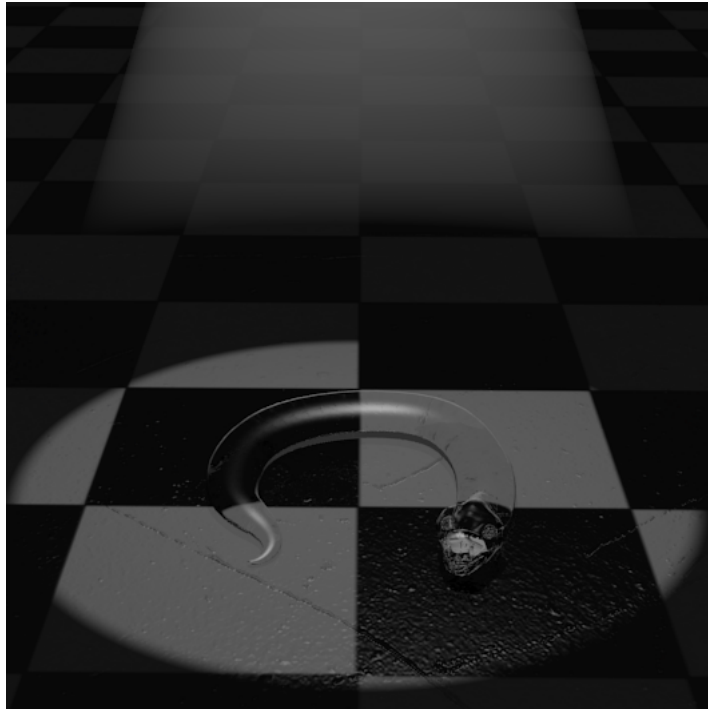
- **Testa:** la testa è composta da una **mesh** importata nel formato **.obj**. Nel tab: Material è stata abilitata con la spunta la sezione **Mirror**, si è impostata una riflettività di 0.9 di colore grigio chiaro, Fresnel: 0.0 e Depth: 6, quest'ultimo parametro indica il numero di rimbalzi che il raytracer effettua sulla superficie per tracciare gli oggetti adiacenti. Gli occhi sono due sfere poste dentro alla mesh della testa il cui materiale ha colore rosso di intensità 1.0, inoltre nel tab **Shading** è stato impostato il campo Emit a 2.0, questo fa sì che il materiale emetta luce.



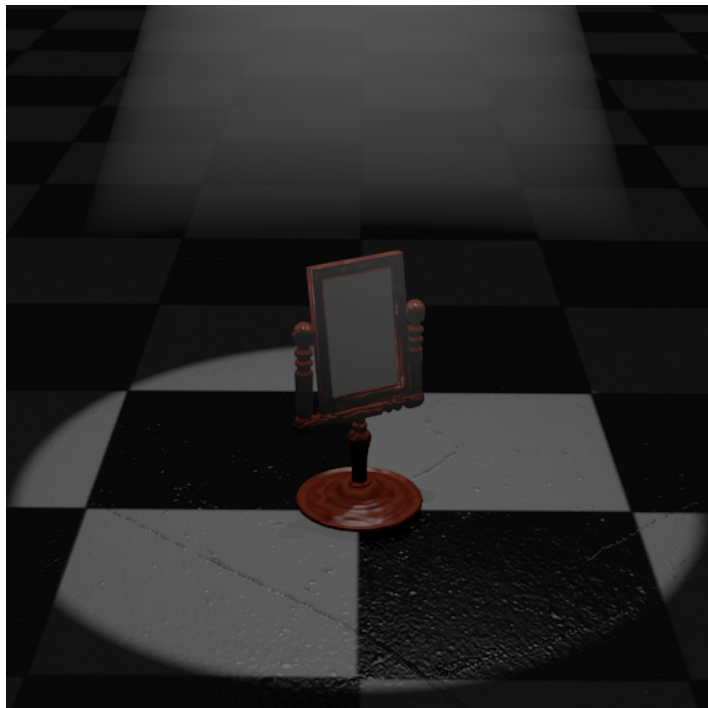
- **Serpente:** il serpente è composto da due elementi: la testa, importata come **mesh** da un oggetto *.obj*, e un corpo formato attraverso una superficie **nurbs** opportunamente modellata.



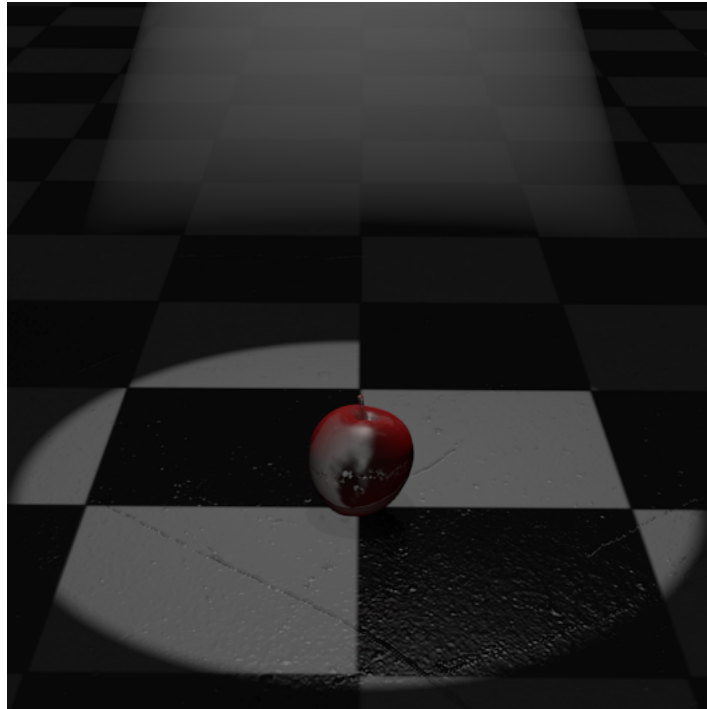
Per il materiale di questi due elementi è stata abilitata la sezione **Transparency**, selezionando il tab **Raytrace** e impostando Fresnel: 5.0, Blend: 1.5 ed IOR: 1.1, per dare un effetto leggermete ruvido al vetro. l'alpha è stato chiaramente impostato a 0.0. La profondità è sempre 6 punti come per i materiali riflettenti.



- **Specchio:** lo specchio è stato importato da un file *.obj*. Per il piano centrale è stato impostato un materiale riflettente, in modo del tutto simile a come è stato fatto in precedenza, per il supporto invece si è impostato un materiale legnoso, impostato come colore marrone e applicandovi la texture procedurale [Wood](#) completa sia di colore che di normale (bump mapping).



- **Mela:** anche la mela è una [mesh](#) importata tramite un oggetto *.obj*. Il materiale è trasparente simile al vetro, creato in modo simile a prima ma con valori di rifrazione differenti e con la particolarità che vi è stato impostato il colore rosso come diffusione, dando un effetto sfumato.



Posizionando questi elementi all'interno della scena ed aggiungendo altre due spot light di colore rosso e blu il risultato ottenuto è il seguente:

