



Politechnika Łódzka

Instytut Informatyki

Instrukcja do laboratorium

Fotorealistyczna Grafika Komputerowa

Laboratorium VI

Odbicia zwierciadlane, refrakcja, rekursywny Raytracing

Wydział Fizyki Technicznej, Informatyki i Matematyki Stosowanej

dr inż. Piotr Napieralski

mgr inż. Krzysztof Guzek

Łódź, 06.02.2012

Instytut Informatyki

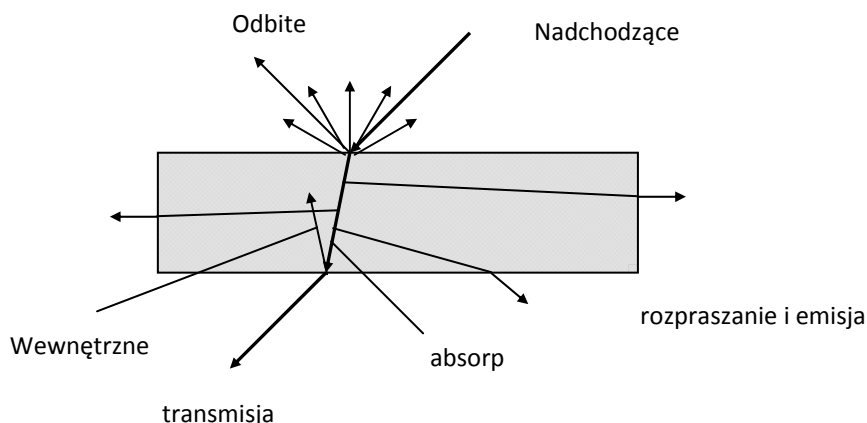
90-924 Łódź, ul. Wólczajska 215, budynek B9

tel. 042 631 27 97, 042 632 97 57, fax 042 630 34 14 email: napieral@ics.p.lodz.pl



1. Wstęp

Zachowanie światła po zetknięciu z materią zależy będzie od fizycznej charakterystyki światła jak i materiału z którego zbudowany jest obiekt, z którym owa interakcja zaszła. Przebieg możliwych zachowań światła przedstawia poniższa ilustracja (Rys.1).

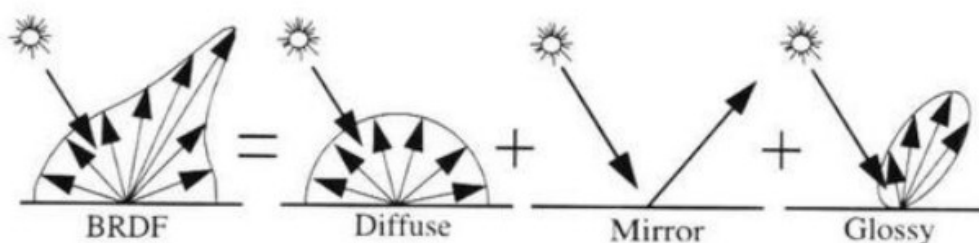


Rys.1 Interakcja światła i materii.

Światło, które pada na granicę dwóch ośrodków może podlegać trzem rodzajom zachowania: odbiciu, absorpcji lub transmisji. Oczywiście w naturalnych warunkach, część światła może zostać odbita, inna część ulec absorpcji a reszta przejść przez badaną powierzchnię. Zgodnie z zasadą zachowania energii:

$$\text{światło nadchodzące} = \text{odbite} + \text{absorbowane} + \text{transmitowane}$$

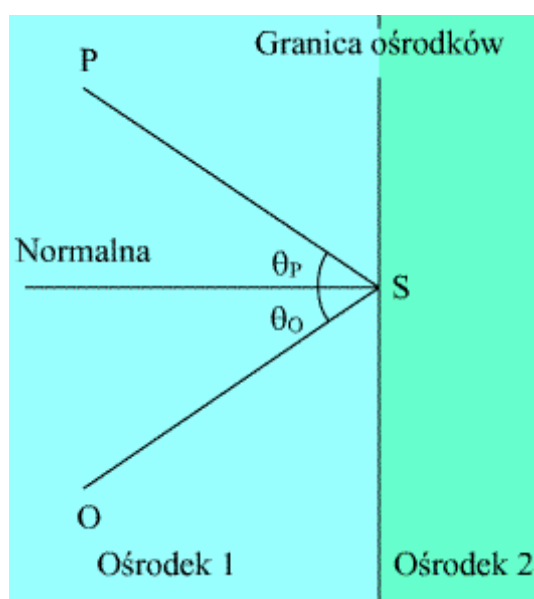
W przypadku obiektów zbudowanych z materiału matowego, większość światła zostaje rozproszona (ang. diffuse), czyli rozchodzi się równomiernie we wszystkich kierunkach. Dla obiektów połyskliwych (ang. glossy), część światła zostanie odbita w konkretnym kierunku. Jeśli odbite światło trafi bezpośrednio do obserwatora następuje rozbłysk (ang. highlight). Te dwa zachowania zostały już przez nas zaimplementowane za pomocą modelu Phong'a. Dotychczas zajmowaliśmy się jedynie takimi obiektami. Istnieje jeszcze trzecia możliwość odbicia światła - całkowite odbicie światła w jednym kierunku (Rys.2). Mówimy wówczas o materiałach lustrzanych lub inaczej zwierciadlanych.



Rys.2 Schematyczny rozkład odbicia światła.

2. Odbicie zwierciadlane

Obiekty o właściwościach lustrzanych odbijają promienie światła, dzięki czemu możemy na ich powierzchni obserwować fragmenty otoczenia, w którym się znajdują. Odwzorowanie tego elementarnego zjawiska w syntetycznym środowisku jest niezbędne dla zachowania realizmu obrazu. Odbicie zwierciadlane może mieć miejsce na gładkiej powierzchni oddzielającej dwa różne ośrodki np. powietrze i lustro wody. Dla uproszczenia obliczeń przyjmujemy, że wszystkie odbicia są lambertowskie, tzn. kąt odbicia jest równy kątowi padania. Zgodnie ze schematem (Rys.3) promień świetlny P zwany promieniem padającym (może on nadejść od obserwatora w przypadku promienia pierwotnego lub od innego obiektu w przypadku promienia wtórnego) pada w punkcie S na granicy ośrodków i odbija się jako promień odbity O.



Rys.3 Schemat odbicia promienia

Odbicie lustrzane zależy od pozycji obserwatora i będzie się zmieniało w zależności od jego położenia. By otrzymać powierzchnie symulującą zachowanie lustra należy wyprowadzić z punktu przecięcia promień odbity (wtórny) w celu znalezienia obiektu odbijającego się zwierciadlanie w punkcie przecięcia. Równanie pozwalające obliczyć kierunek promienia odbitego zostało już zaimplementowane na potrzeby składowej Specular równania Phong'a i ma postać:

$$R = I - 2 \cdot N \cdot (N \bullet I)$$

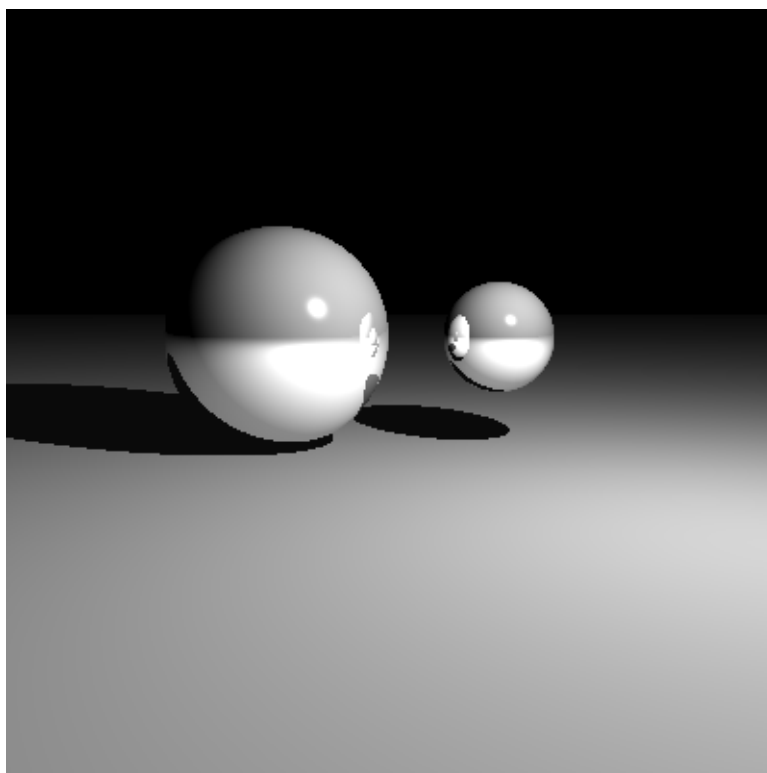
gdzie:

R - jest poszukiwanym wektorem kierunkowym promienia odbitego

I - jest wektorem kierunkowym nadejścia promienia

N - jest normalną w punkcie przecięcia.

Mając kierunek odbitego promienia powinniśmy mieć także jego punkt początkowy. Jest nim oczywiście punkt przecięcia obiektu z nadchodzącym promieniem. Dla tak utworzonego nowego promienia szukamy przecięcia z innymi obiektami geometrycznymi w scenie i dla najbliższego z nich obliczamy kolor piksela według modelu Phong (chyba że najbliższe znalezione przecięcie będzie również z obiektem lustrzanym, wówczas wysyłamy rekurencyjnie kolejny promień odbity). Rezultaty wielokrotnego odbicia zwierciadlanego możemy zaobserwować na Rys.4.



Rys.4 Dwie sfery o powierzchniach zwierciadlanych odbijające się wzajemnie.

3. Refrakcja

Część wiązki świetlnej, która nie uległa odbiciu przechodzi przez płaszczyznę graniczną i rozchodzi się dalej w nowym ośrodku. Powstały w ten sposób promień określany jest promieniem załamanym. Zjawisko przechodzenia światła przez powierzchnię rozgraniczającą dwa różne ośrodki nazywamy refrakcją (załamaniem). Refrakcja polega na zmianie kierunku rozchodzenia się fali związanej ze zmianą jej prędkości. W konsekwencji powoduje to zmianę długości fali przy zachowanej stałej częstotliwości. Fakt, że światło może rozchodzić się w danym ośrodku świadczy o jego przezroczystości.

Prawo fizyki opisujące zmianę kierunku biegu promienia światła przy przejściu przez granicę między dwoma ośrodkami przezroczystymi o różnych współczynnikach załamania. Prawo odkrył holenderski astronom i matematyk Willebrord Snell w 1621 roku. Prawo to, zwane prawem Snella mówi, że promień padający i załamany oraz prostopadła padania (normalna) leżą w jednej płaszczyźnie, a kąty spełniają zależność:

$$\frac{\sin \theta_1}{\sin \theta_2} = \frac{n_2}{n_1}$$

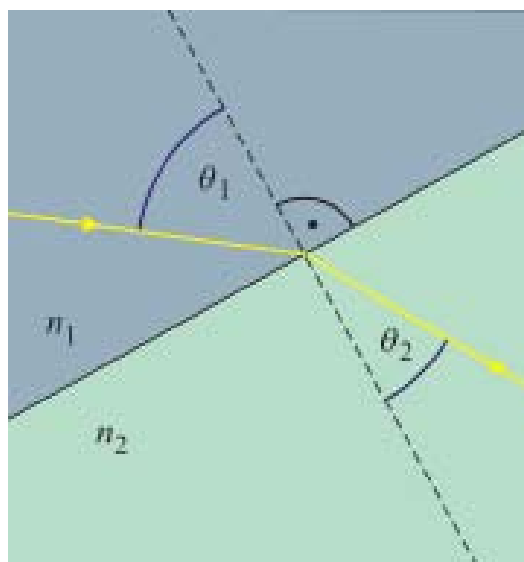
Gdzie:

n_1 - współczynnik załamania światła ośrodka pierwszego,

n_2 - współczynnik załamania światła ośrodka drugiego,

θ_1 - kąt padania, kąt między promieniem padającym a normalną do powierzchni granicznej ośrodków,

θ_2 - kąt załamania, kąt między promieniem załamanym a normalną.



Rys.5 Schemat załamania wiązki światła padającej na płaską powierzchnię rozgraniczającą dwa ośrodki.

Współczynnik załamania (ang. index of refraction) charakterystyczny dla danego materiału, jest równy stosunkowi prędkości światła v w danym ośrodku do prędkości światła c w próżni. Tabela.1 podaje współczynniki załamania dla wybranych ośrodków optycznych. Jak łatwo zauważyć współczynnik załamania dla danego ośrodka rośnie wraz z gęstością.

Ośrodek	n	Ośrodek	n
próżnia	1 (ściśle)	szkło typowe (kron)	1,52
powietrze (0°C, 1 atm)	1,00029	chlorek sodu	1,54
woda (temp. 20°C)	1,33	polistyren	1,55
aceton	1,36	dwusiarczek węgla	1,63
alkohol etylowy	1,36	ciężkie szkło (flint)	1,65
roztwór cukru (30%)	1,38	szafir	1,77
kwarc topiony	1,46	bardzo ciężkie szkło	1,89
roztwór cukru (80%)	1,49	diamant	2,24

Tabela.1 Współczynniki załamania światła dla próżni i wybranych ośrodków optycznych

Implementacja zjawiska refrakcji będzie wymagała rozszerzenia definicji materiału o dodatkowe pole przechowujące wartość współczynnika załamania światła.

Należy zaznaczyć, że prawo Snella opisuje zależności geometryczne jakie zachodzą między kierunkami promieni w sposób kompletny jedynie dla ośrodków jednorodnych. W przypadku wspomnianych wcześniej powierzchni anizotropowych promień świetlny może rozdzielać się na dwa promienie. Powstałe tak zjawisko nazywamy dwójłomnością.

Promień załamany tworząc z normalną kąt 90 stopni, zaczyna się ślizgać po powierzchni rozgraniczającej dwa ośrodki. Kąt padania, przy którym zachodzi taka sytuacja nazywany jest kątem granicznym. Przy wzroście kąta padania promienia powyżej wartości kąta granicznego, promień światła nie załamuje się i powstaje efekt całkowitego wewnętrznego odbicia.

Podobnie jak w przypadku odbicia zwierciadlanego, aby odwzorować zjawisko refrakcji należy poprowadzić z punktu przecięcia nowy promień. Kierunek tego promienia zależy będzie oczywiście od kąta padania względem wektora normalnego powierzchni w punkcie przecięcia (wektor normalny jest skierowany zawsze w stronę ośrodka z którego nadchodzi promień!) oraz współczynników załamania obu ośrodków. W celu obliczenia wektora kierunkowego promienia wtórnego możemy wykorzystać podstawowe tożsamości trygonometryczne oraz przedstawione prawo Snella:

Zależność pomiędzy kątem padania i kątem załamania promienia:

$$\lambda_s \sin \theta = \lambda_t \sin \phi$$

λ_s, λ_t - współczynniki załamania ośrodków

$$\sin^2 \phi + \cos^2 \phi = 1$$

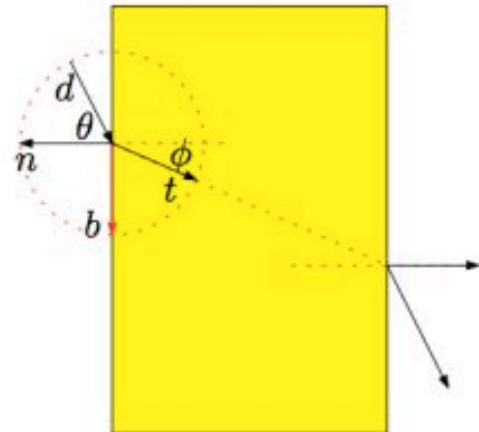
$$\cos^2 \phi = 1 - \frac{\lambda_s^2(1 - \cos^2 \theta)}{\lambda_t^2}$$

$$t = b \sin \phi - n \cos \phi$$

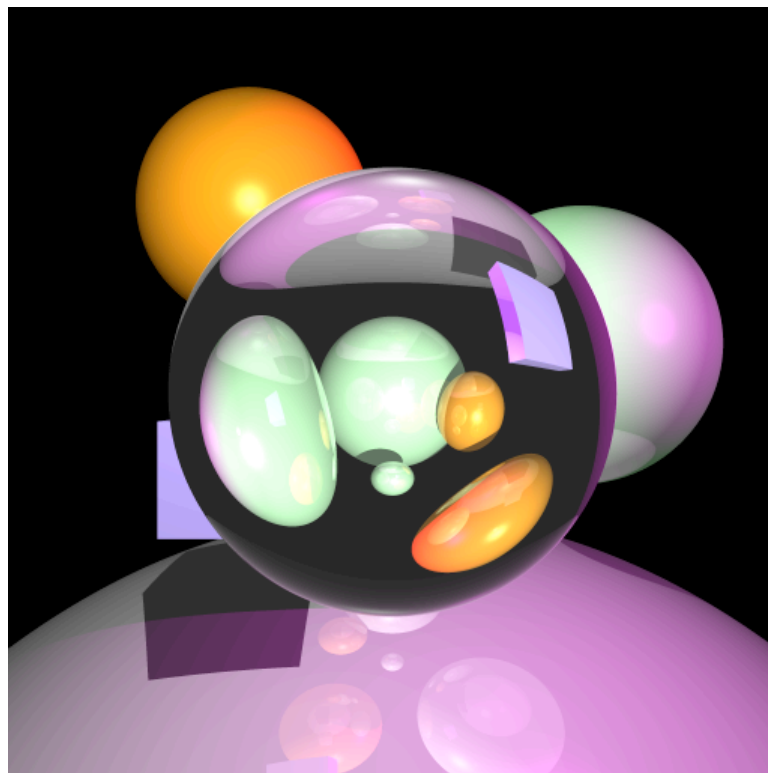
$$d = b \sin \theta - n \cos \theta$$

$$b = \frac{d + n \cos \theta}{\sin \theta}$$

$$\begin{aligned} t &= \frac{\sin \phi (d + n \cos \theta)}{\sin \theta} - n \cos \phi \\ &= \frac{\lambda_s (d + n \cos \theta)}{\lambda_t} - n \cos \phi \\ &= \frac{\lambda_s (d - n(d \cdot n))}{\lambda_t} - n \sqrt{1 - \frac{\lambda_s^2(1 - (d \cdot n)^2)}{\lambda_t^2}} \end{aligned}$$



W wyniku refrakcji obiekty otoczenia znajdujące się bezpośrednio za ośrodkiem uginającym promienie światła zostają zniekształcone (Rys.6). Odwzorowanie zjawiska refrakcji w środowisku 3D zdecydowanie podnosi poziom realizmu materiałów szklanych i cieczy.



Rys.6 Wyrenderowany obraz ze szklaną sferą powodującą refrakcję promieni.

4. Rekursywny Raytracing

Zaimplementowany do tej pory na laboratoriach klasyczny algorytm śledzenia promieni powinien wyglądać następująco:

1. Dla Każdego piksela obrazu (rzutni), z pozycji wirtualnej kamery wyprowadzany jest promień (lub kilka promieni w zależności od wybranego algorytmu próbkowania) przechodzący przez aktualnie rozpatrywany piksel obrazu.
2. Sprawdzane są kolizje danego promienia ze wszystkimi zdefiniowanymi w scenie prymitywami (sfera, płaszczyzna, trójkąt), a następnie ze wszystkich ewentualnie znalezionych punktów przecięcia promienia wybierany jest ten, który znajduje się najbliżej kamery.
3. Ze znalezionego w powyższy sposób punktu przecięcia prowadzone są promienie cienia do wszystkich źródeł światła występujących w scenie i sprawdzane jest czy przecinają one inne obiekty geometryczne sceny.
4. Jeśli promień cienia, prowadzony do danego źródła światła, przecina chociaż jeden nieprzezroczysty obiekt geometryczny to znaczy, że światło od tego źródła nie dociera bezpośrednio do rozpatrywanego punktu przecięcia i nie bierze udziału w obliczaniu koloru piksela. Jeśli promień cienia, prowadzony do danego źródła światła, nie przetnie na swej drodze żadnego innego obiektu geometrycznego sceny, wówczas obliczany jest kolor piksela dla natężenia tego źródła, zgodnie z określonym modelem oświetlenia (np. Phong).

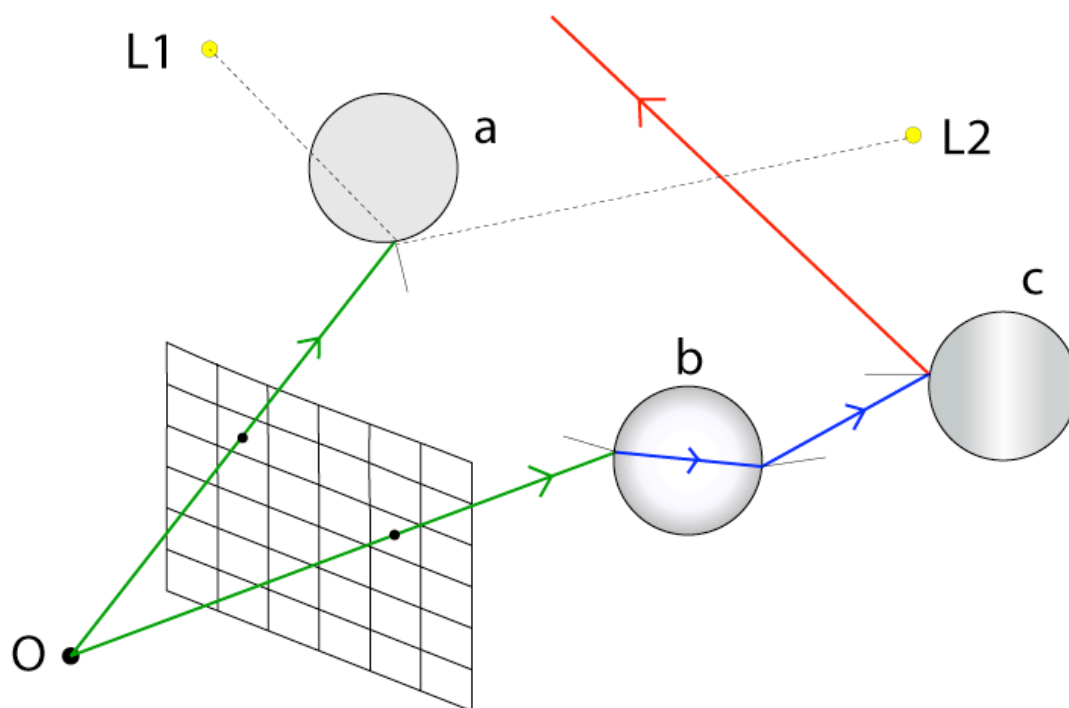
Aby uzyskać rekursywny raytracing, wystarczy po zaimplementowaniu funkcji obliczających odbicia zwierciadlane i refrakcję, dodać do powyższego algorytmu jeden dodatkowy warunek:

- 3a. Jeśli najbliższe znalezione przecięcie rozpatrywanego promienia nastąpiło z obiektem geometrycznym odbijającym lub załamującym światło, wówczas zamiast liczyć kolor piksela według modelu Phong, z punktu przecięcia wysyłany jest promień wtórny (odpowiednio, promień odbity lub promień załamany) i algorytm rekursywnie powtarza się od drugiego kroku.

Należy zdefiniować maksymalną dopuszczalną liczbę odbić promieni wtórnych, uwzględniając sytuację gdy w scenie znajduje się więcej niż jeden obiekt o właściwościach zwierciadlanych, aby nie doszło do nieskończonych odbić. Kolor piksela wyznaczany jest dopiero gdy promień wtórny trafi w obiekt rozpraszający światło (nie odbijający zwierciadlane i nie załamujący).

W rzeczywistości gdy światło pada na granicę dwóch ośrodków możliwe jest jednoczesne odbicie i załamanie, opisane za pomocą równań Fresnela. Dla uproszczenia obliczeń w naszym silniku renderującym będziemy przyjmować, że powierzchnia obiektu w

zależności od zdefiniowanego parametru materiału będzie albo całkowicie zwierciadlana albo całkowicie załamująca promienie (Rys.7).



Rys.7 Schemat rekursywnego śledzenia promieni.

Obserwator (kamera) znajduje się w punkcie **O**, na scenie umieszczone zostały dwa światła L1 i L2 oraz trzy obiekty:

- a** : nieprzezroczysty, rozpraszający, nie odbija i nie załamuje światła
- b** : przezroczysty, załamuje światło
- c** : zwierciadlany, odbija światło

Z punktu **O** wyprowadzane są promienie pierwotne (zielone). Pierwszy z rozpatrywanych promieni trafia w obiekt **a** - ponieważ jest on rozpraszający, więc nie są generowane żadne promienie wtórne, następuje obliczenie natężenia światła. Kreskami przerywanymi zaznaczono dodatkowe promienie cienia, służące określeniu widoczności światła w punktach przecięcia (sam obiekt (**a**) blokuje światło od źródła L1). Drugi promień pierwotny (zielony) trafia w obiekt **b** - ponieważ załamuje on światło, toteż z punktu przecięcia wypuszczany jest promień wtórny załamany (niebieski). Ten promień przechodzi przez ośrodek **b** i ponownie ulega załamaniu. Po opuszczeniu ośrodka **b** promień trafia w obiekt **c**, który odbija światło, zatem generowany jest kolejny promień wtórny - tym razem odbity (czerwony), który jednak nie przecina się z żadnym obiektem na scenie i na tym kończy się analiza.

W programie powinna zostać zaimplementowana klasa Sceny. W kontekście metody śledzenia promieni, sceną nazywamy zbiór obiektów oraz źródeł światła biorących udział

w tworzeniu naszego wirtualnego świata. Scena pozwala nam uporządkować światła i obiekty, dając możliwość w łatwy i elegancki sposób przeprowadzić algorytm przecinania obiektów i znajdowania pożądaných punktów przecięcia. Scena składać się będzie z kolekcji, tablic lub innych form, pozwalających zapamiętać złożoną strukturę danych, by móc te dane porównywać odzyskiwać i dokonywać na nich określonych operacji. Forma implementacji sceny jest dowolna. Powinna zawierać jednakże dwie listy: listę obiektów i listę światel. Metody powinny pozwalać dodać do sceny obiekty jak i światła. Powinna istnieć możliwość sprawdzenia ilości światel i obiektów w scenie.

Wraz z wprowadzeniem sceny powinniśmy zaimplementować osobną klasę oznaczającą przecięcie, będzie ona oczywiście zawierać współrzędne punktu przecięcia oraz referencję do przecinanego obiektu. Dodatkowe pole które powinna zawierać to pole oznaczające odległość przecięcia od początku promienia. Dyktowane jest to potrzebą określenia który obiekt w scenie znajduje się bliżej a który dalej.

Poniżej zamieszczono pseudokod funkcji obliczającej oświetlenie dla znalezionej punktu przecięcia za pomocą algorytmu rekursywnego raytracingu. Parametr liczba_odbić pozwala określić maksymalną głębokość rekurencji dla promieni odbitych i załamanych. Parametry px i py przechowują współrzędne x i y aktualnie przetwarzanego piksela. Funkcja zwraca obliczone natężenie.

```
Natężenie oświetl(scena, pozycja_kamery, punkt_przecięcia,
liczba_odbić, px, py)
{
    natężenie = utwórz nowe zerowe natężenie;

    // jeśli przecięcie należy do obiektu rozpraszającego
    // lub przekroczyliśmy dopuszczalną głębokość rekurencji
    if((punkt_przecięcia->obiekt jest rozpraszający) lub
    liczba_odbić<=0)
    {
        // testowanie widoczności światel
        for(wszystkie źródła w scenie)
        {
            źródło = pobierz bieżące źródło;
            promień_cienia = promień(punkt_przecięcia, źródło->pozycja);
            test = znajdź przecięcie z obiektami sceny(promień_cienia);
            if(test == null){
                natężenie += oblicz diffuse i specular Phong'a
                (punkt_przecięcia, źródło, pozycja_kamery);}
        }
        natężenie += oblicz Ambient;
    }
}
```

```

// jeśli przecięcie należy do obiektu lustrzanego lub załamującego
else if(liczba_odbić > 0)
{
    nadchodzący = Wektor(pozycja_kamery, punkt_przecięcia);

    if(punkt_przecięcia->obiekt jest lustrem){
        // wyznacz promień odbity według prawa odbicia
        wtórny = oblicz_promień_odbity(punkt_przecięcia,
        nadchodzący);}
    else if(punkt_przecięcia->obiekt jest refrakcyjny){
        // wyznacz promień załamany według prawa Snella
        wtórny = oblicz_promień_zalamany(punkt_przecięcia,
        nadchodzący);}

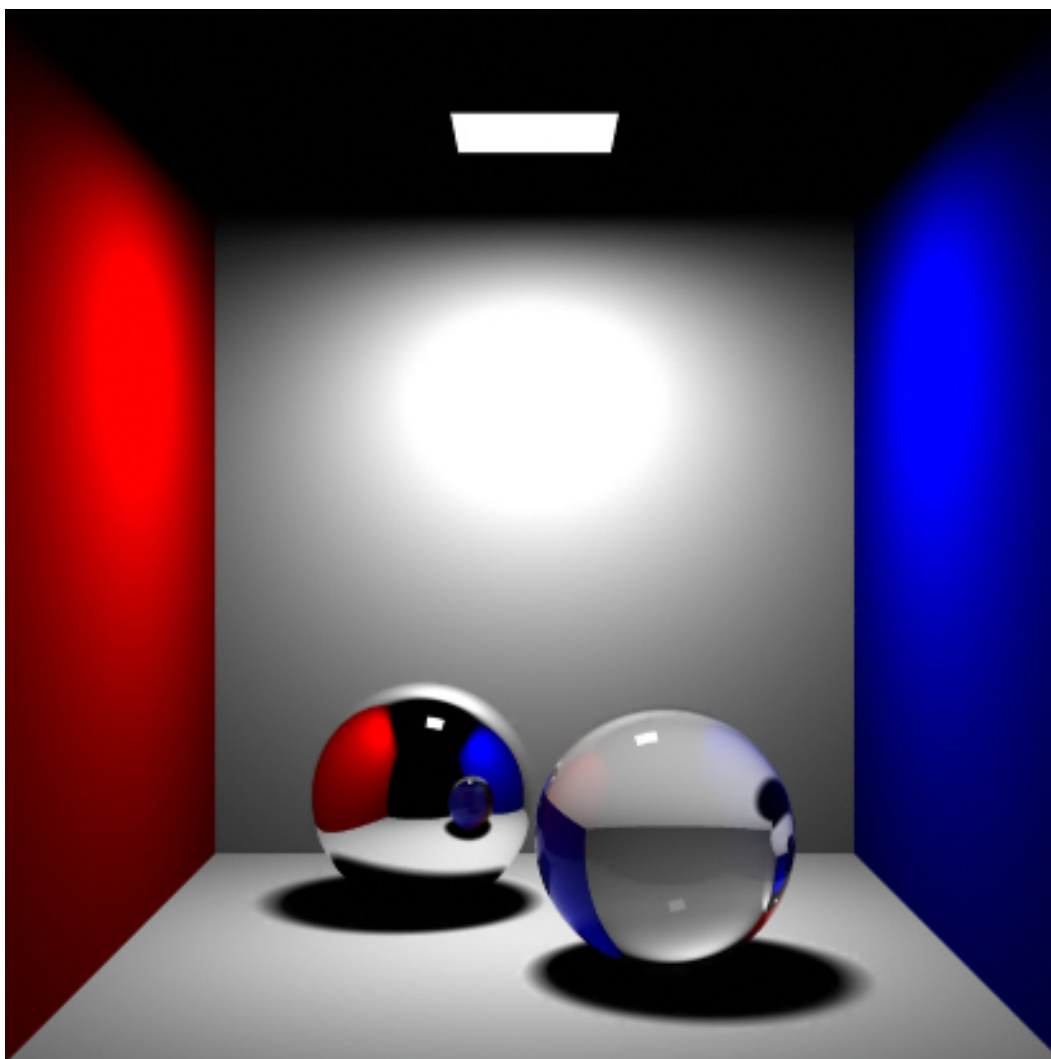
    test = znajdź_przecięcie_z_obiektami_sceny(wtórny);
    if(test != null){
        // jeśli promień wtórny przeciął obiekt sceny wywołaj
        // rekurencyjnie funkcję oświetl
        natężenie += oświetl(scena, pozycja_kamery, test,
        liczba_odbić-1, px, py);}
    else
        //jeśli nie dodaj kolor tła
}

return natężenie;
}

```

5. Zadanie

Proszę zaimplementować funkcje obliczające odbicie zwierciadlane i załamanie promienia, a następnie zmodyfikować algorytm raytracingu, tak aby uwzględniał rodzaj materiału obiektu z którym przecina się promień (rozpraszający, zwierciadlany lub refrakcyjny). Proszę wyrenderować wynik w postaci cornel box'a zawierającego dwie sfery: zwierciadlaną i refrakcyjną (Rys.8). Ściany cornel box'a powinny być rozpraszające (matowe).



Rys. 8 Obraz wynikowy wyrenderowany za pomocą algorytmu rekursywnego raytracingu.