

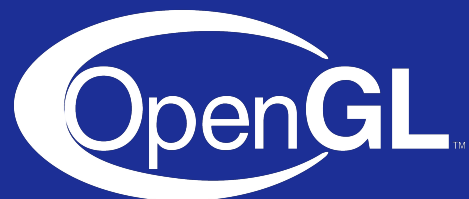
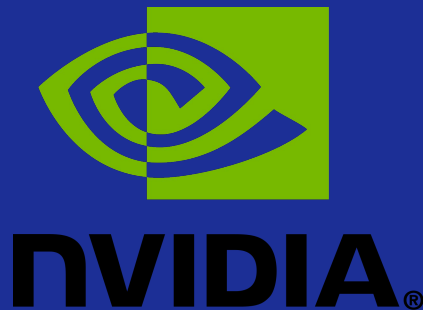
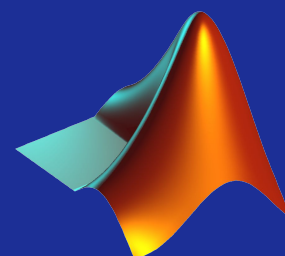


Grafika Komputerowa

Wykład 5

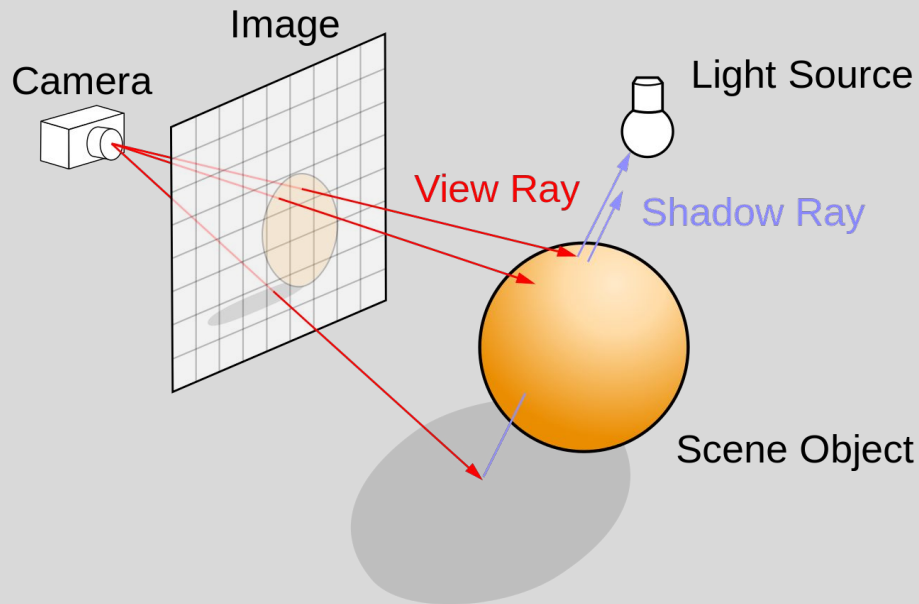
Potok Renderowania

Oświetlenie



mgr inż. Michał Chwesiuk

Ray tracing



- Podejście **śledzenia promieni** (ang. *ray tracing*) stosuje się w **grafice realistycznej**.
 - Śledzone są **promienie przechodzące przez piksele** obrazu wynikowego w celu **poszukiwania przecięć** z obiektami sceny.
 - Algorytm bardzo kosztowny z uwagi na drogą obliczeniowo operację **śledzenia promieni światła** i ich odbić.



Ray tracing - przykłady

Ray tracing

Potok renderowania
geometrii

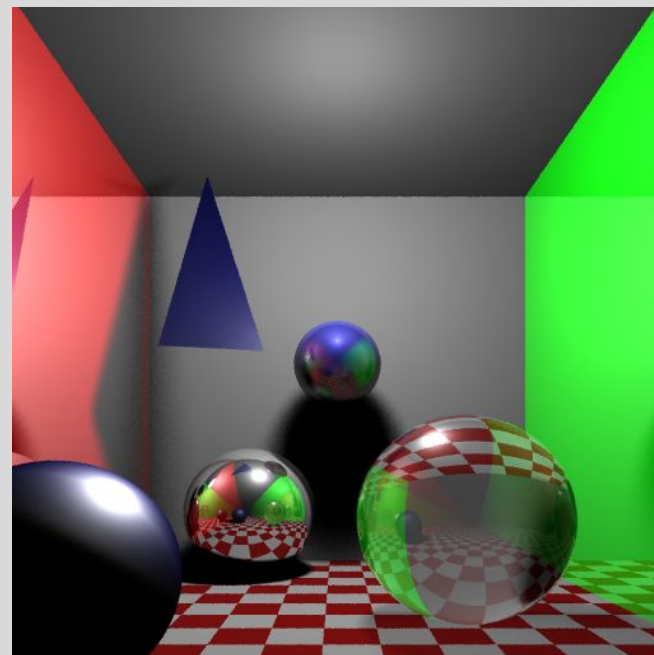
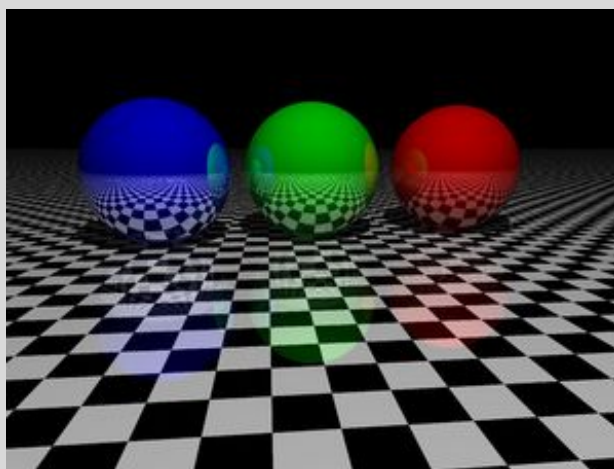
Test głębokości

Oświetlenie lokalne i
globalne

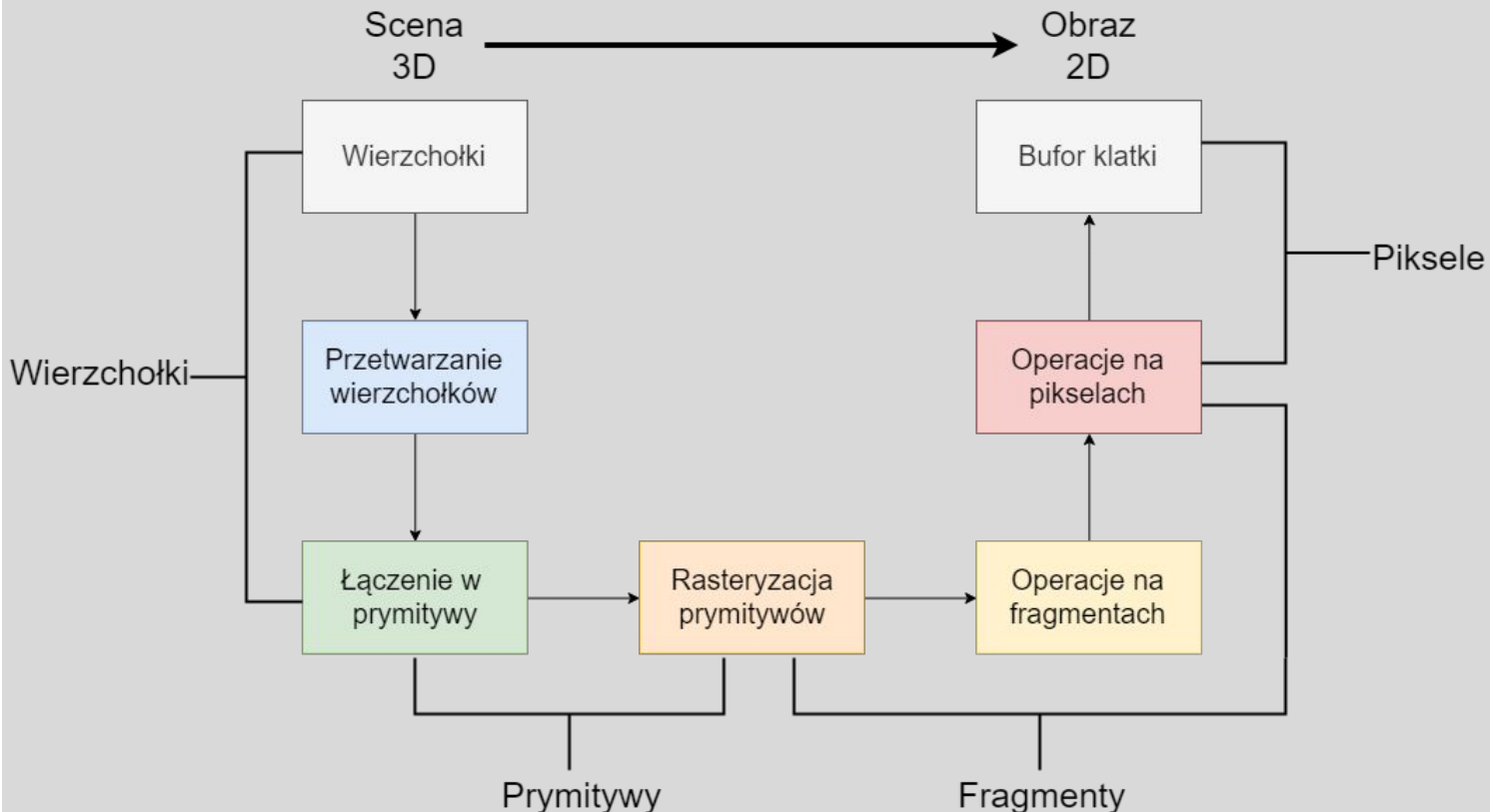
Cieniowanie

Wektory normalne

Model oświetlenia



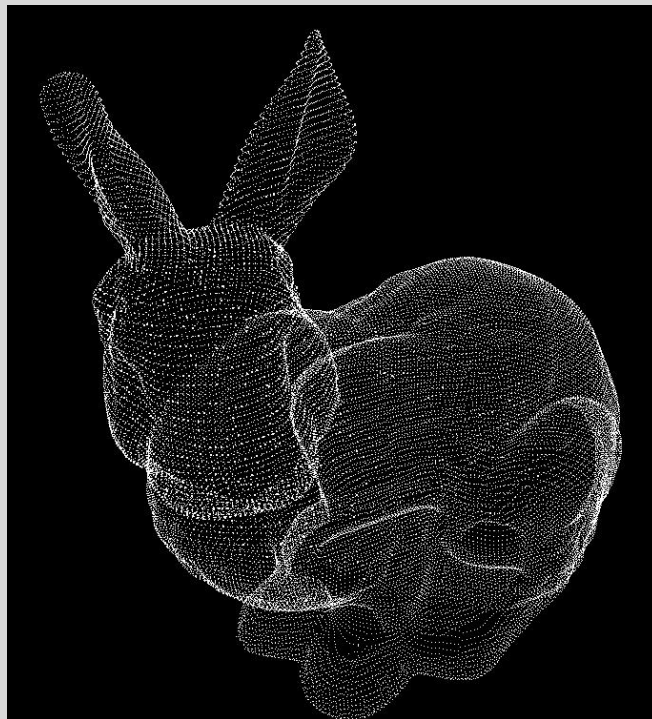
Potok renderowania geometrii



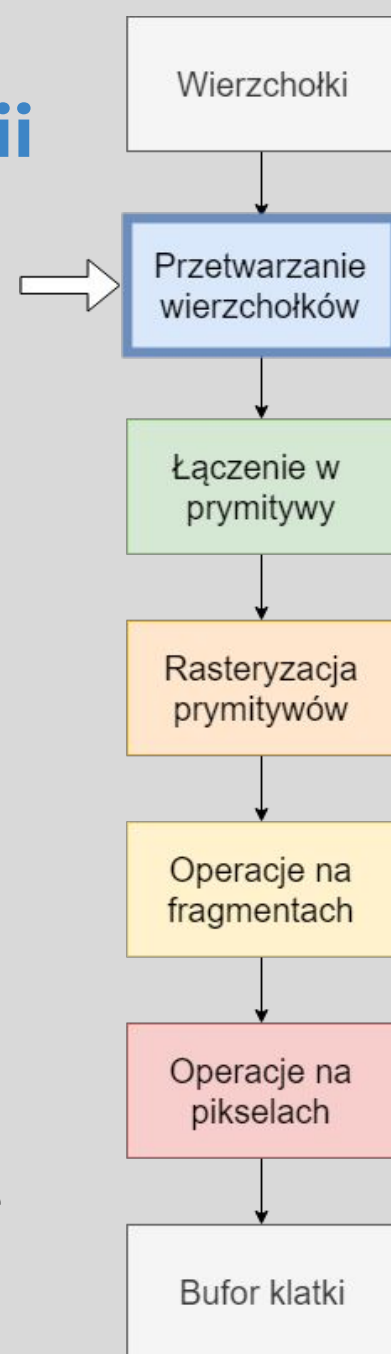
- **Potok renderowania geometrii** (ang. *rendering pipeline*) stosujemy w **grafice czasu rzeczywistego**.
- Podejście w pewnym stopniu odwrotne do ray tracingu, zaczynając od geometrii **rzutujemy** ją na obraz.



Potok renderowania geometrii



- Każdy obiekt w scenie składa się z **wierzchołków**.
- Atrybuty tych wierzchołków (np. położenie) poddawane są **przekształceniom**:
 - **Transformacje** **geometryczne** (macierz modelu)
 - **Morphing, symulacja ruchu** itp.



Potok renderowania geometrii

Ray tracing

Potok renderowania geometrii

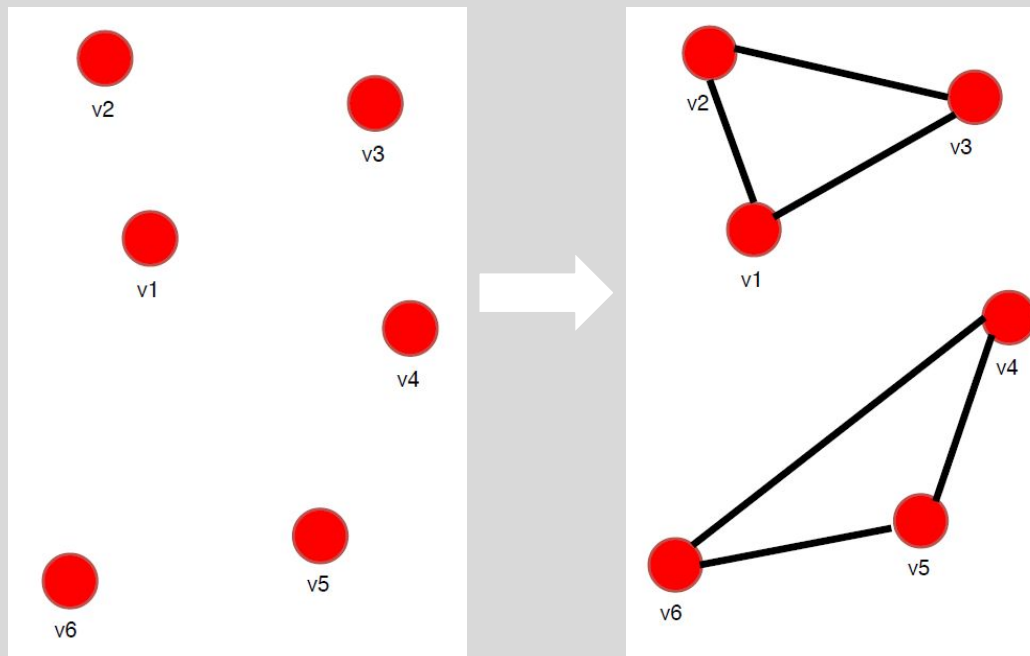
Test głębokości

Oświetlenie lokalne i globalne

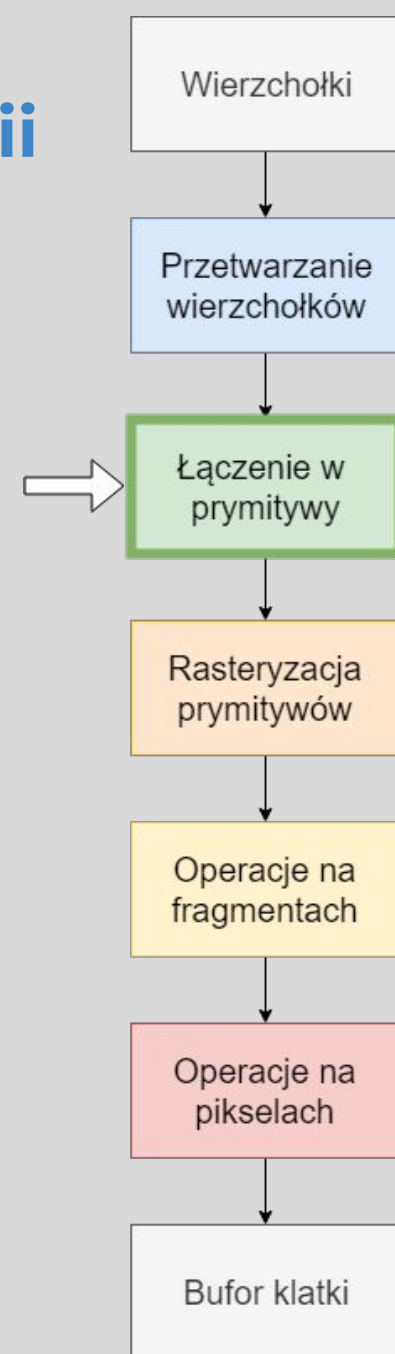
Cieniowanie

Wektory normalne

Model oświetlenia



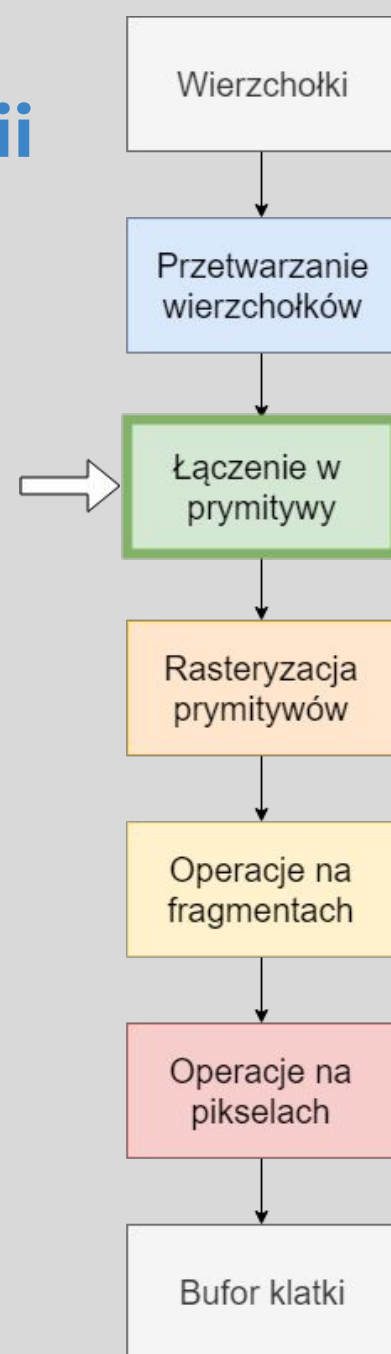
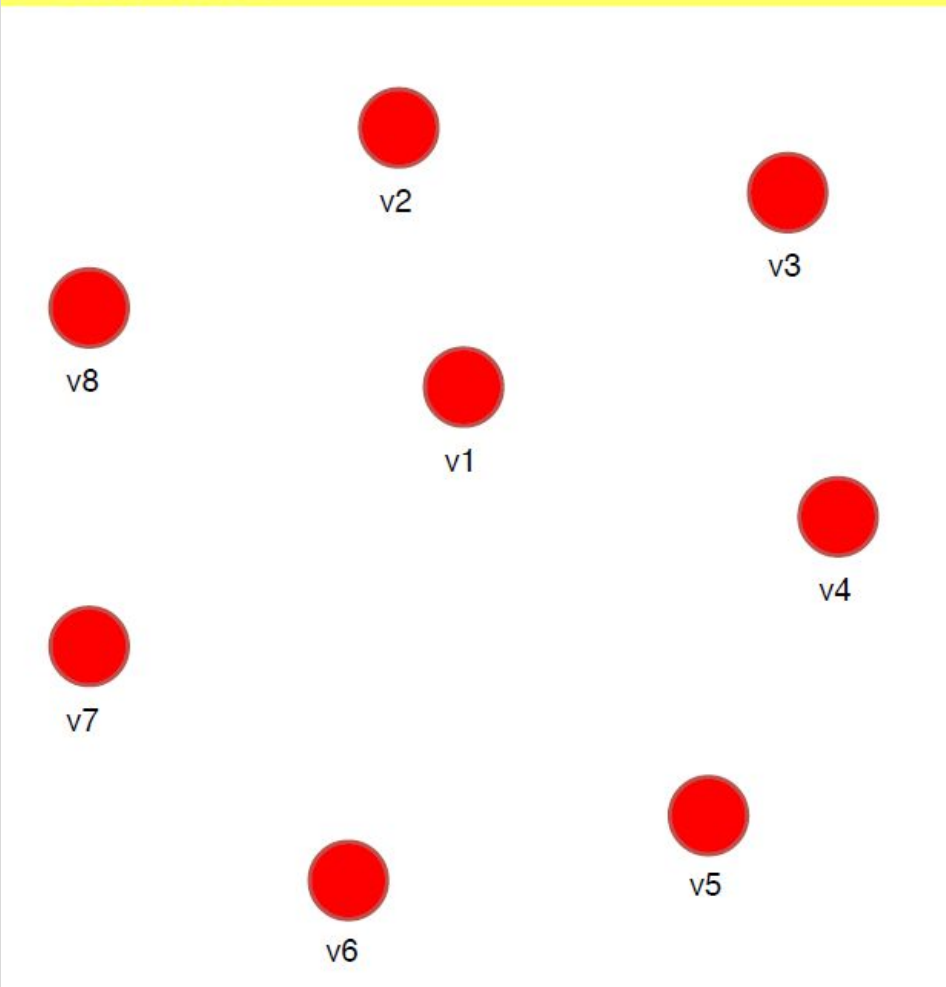
- Grupy wierzchołków **łączone są w prymitywy**, określone przez twórcę modelu.
- Na przykład :
 - Linie
 - Trójkąty
 - Czworokąty





Potok renderowania geometrii

POINTS



Ray tracing

Potok renderowania geometrii

Test głębokości

Oświetlenie lokalne i globalne

Cieniowanie

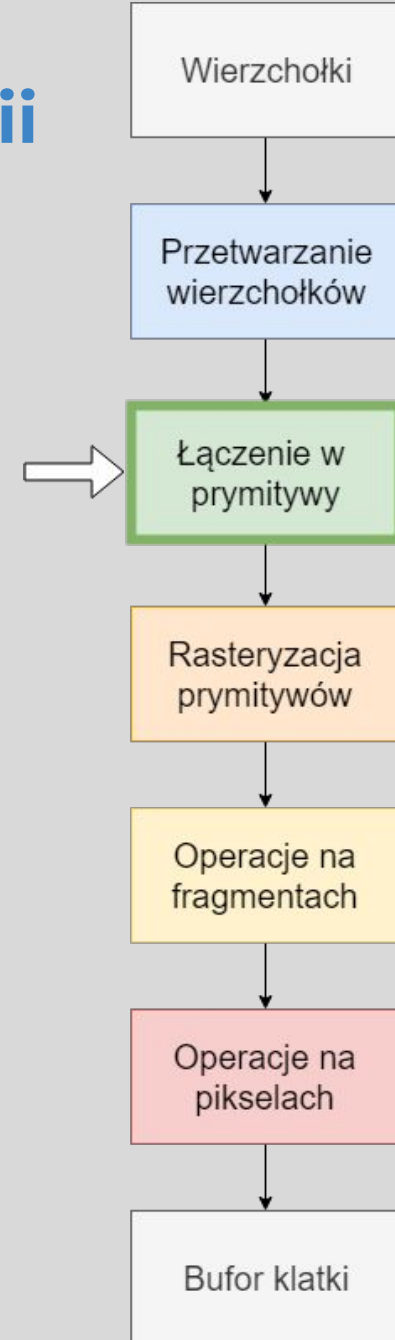
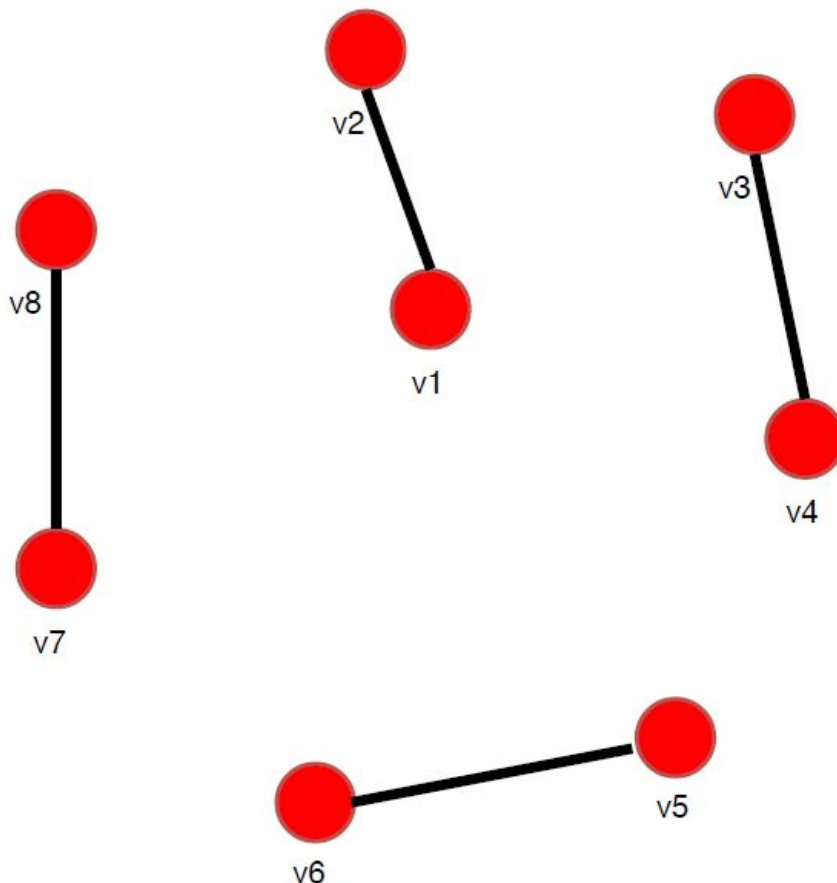
Wektory normalne

Model oświetlenia



Potok renderowania geometrii

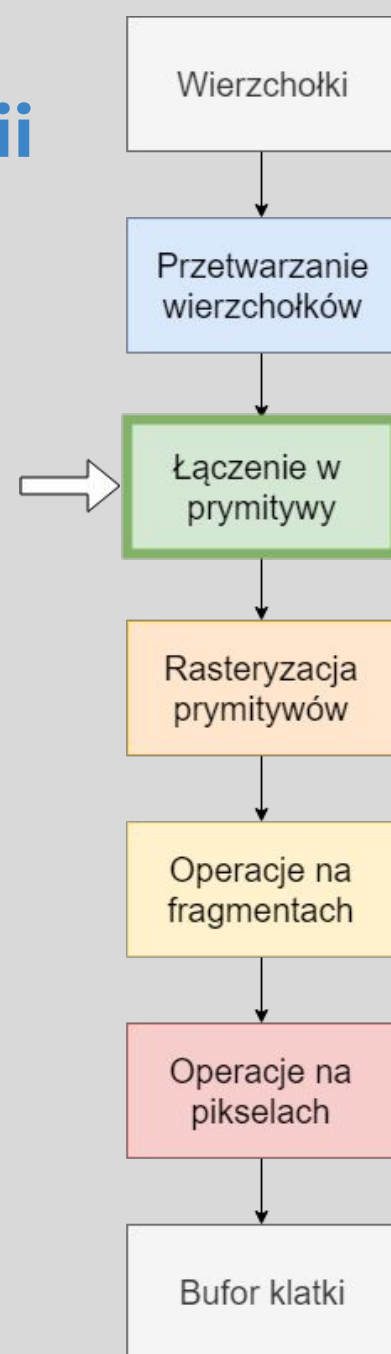
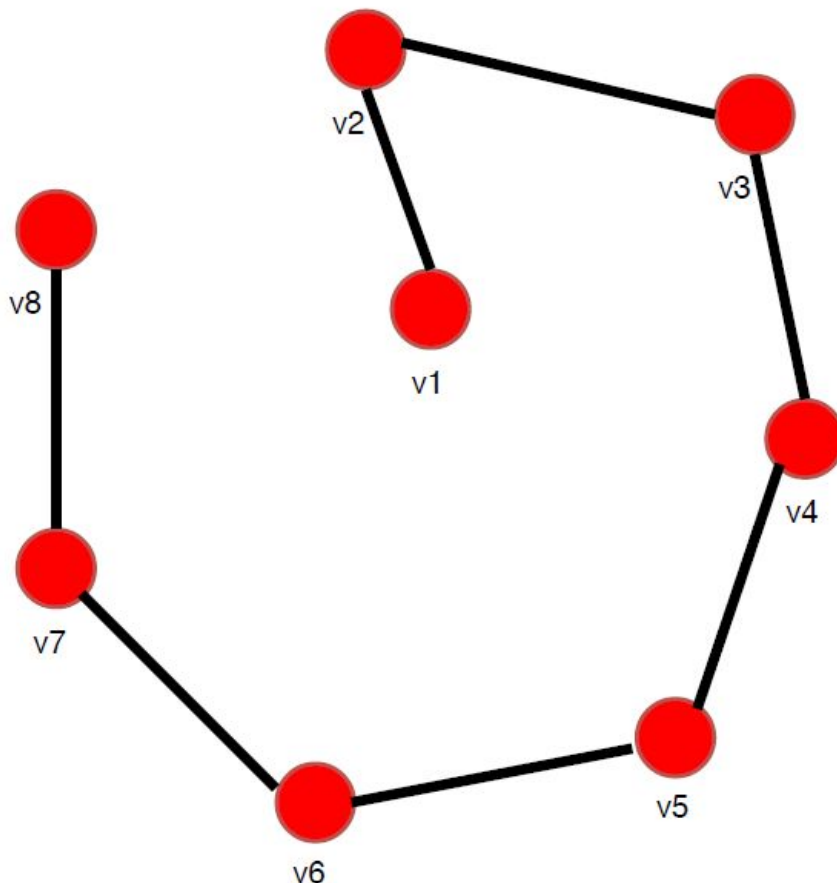
LINES





Potok renderowania geometrii

LINE STRIP



Ray tracing

Potok renderowania geometrii

Test głębokości

Oświetlenie lokalne i globalne

Cieniowanie

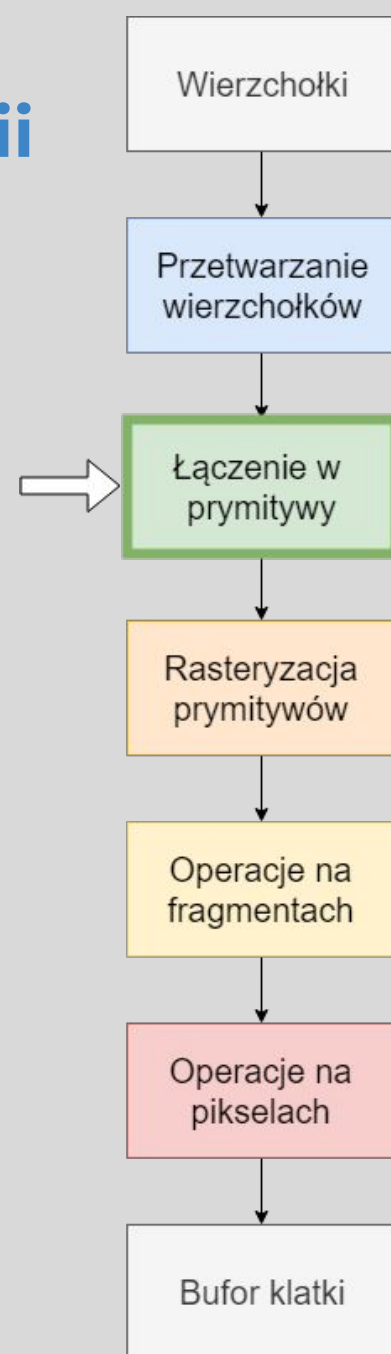
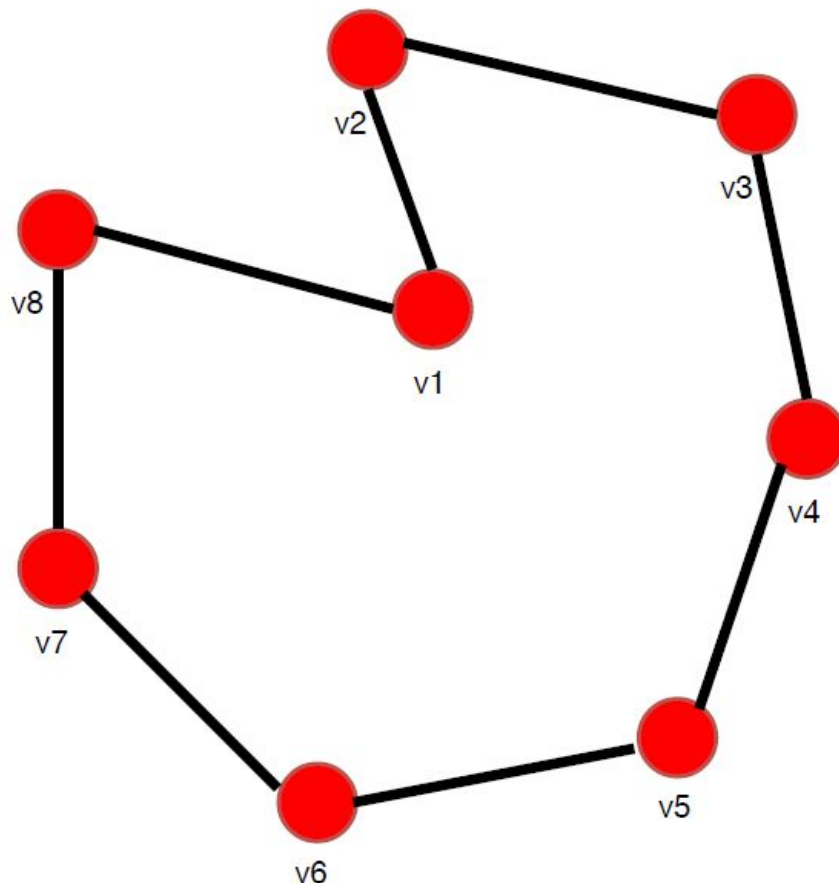
Wektory normalne

Model oświetlenia



Potok renderowania geometrii

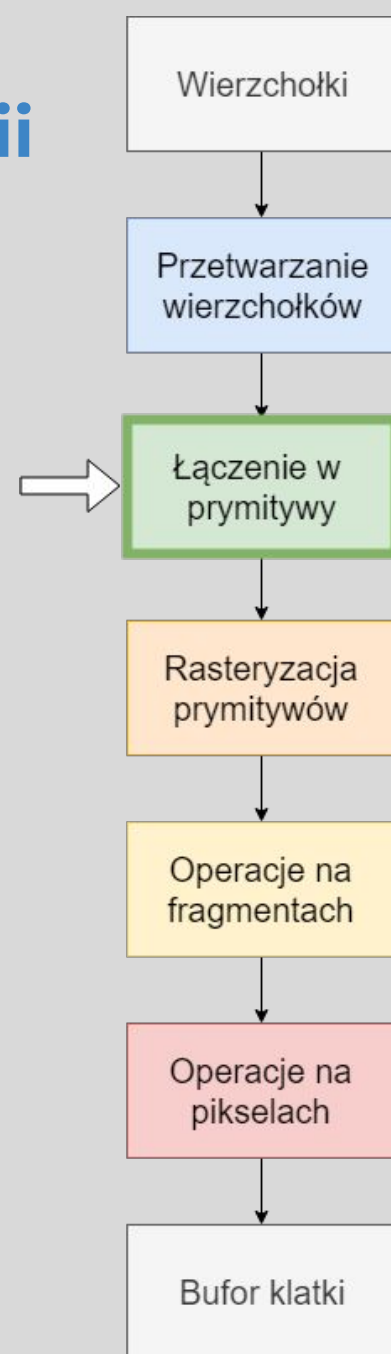
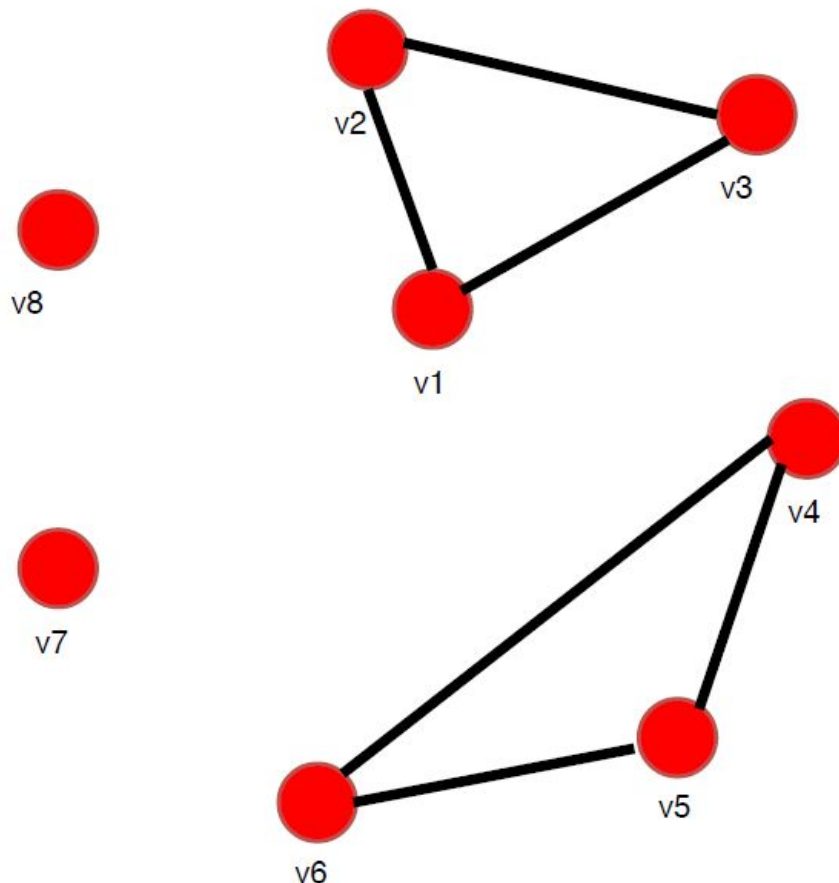
LINE LOOP





Potok renderowania geometrii

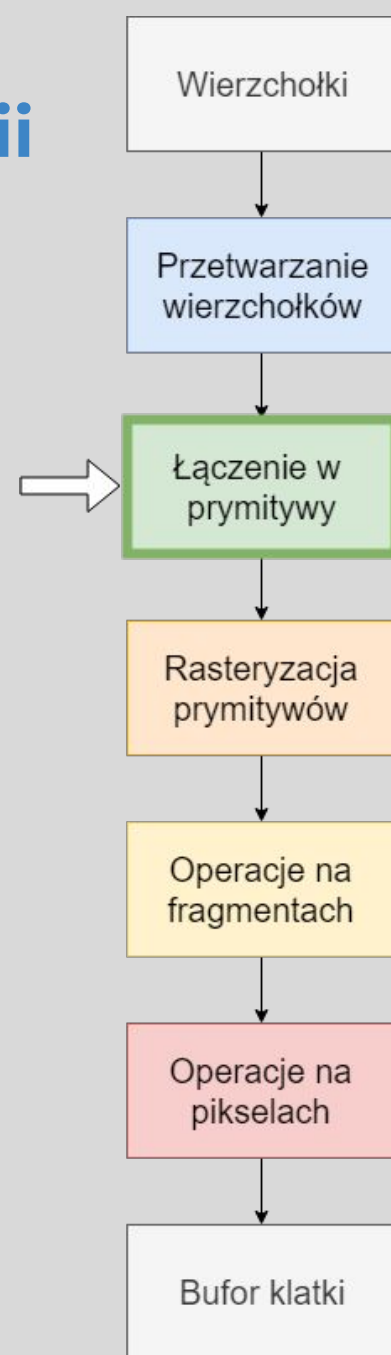
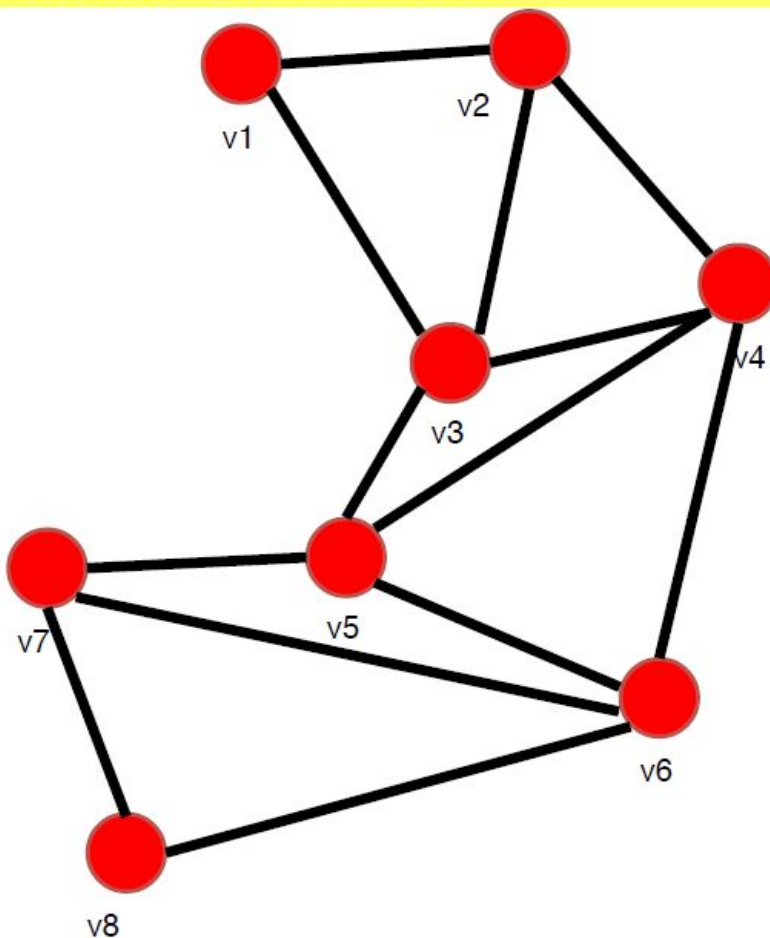
TRIANGLES





Potok renderowania geometrii

TRIANGLE STRIP



Ray tracing

Potok renderowania geometrii

Test głębokości

Oświetlenie lokalne i globalne

Cieniowanie

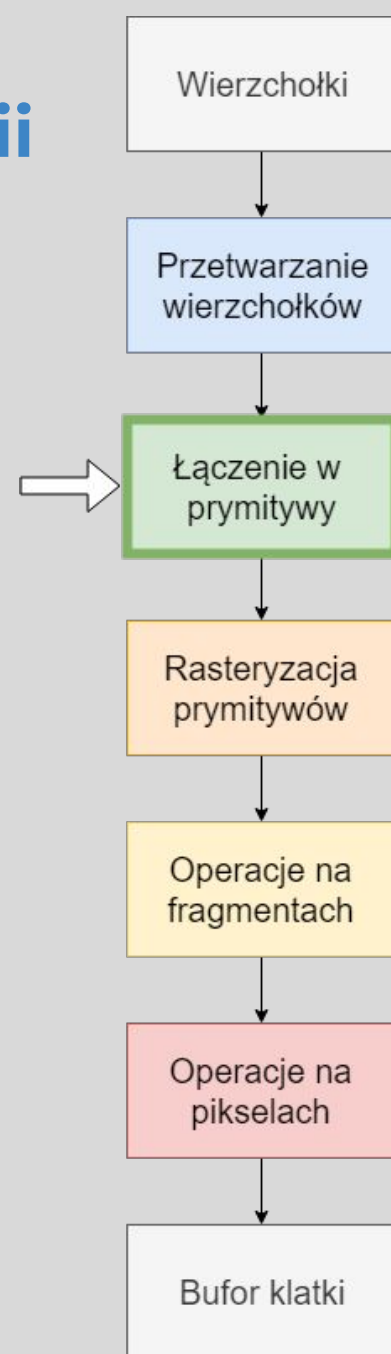
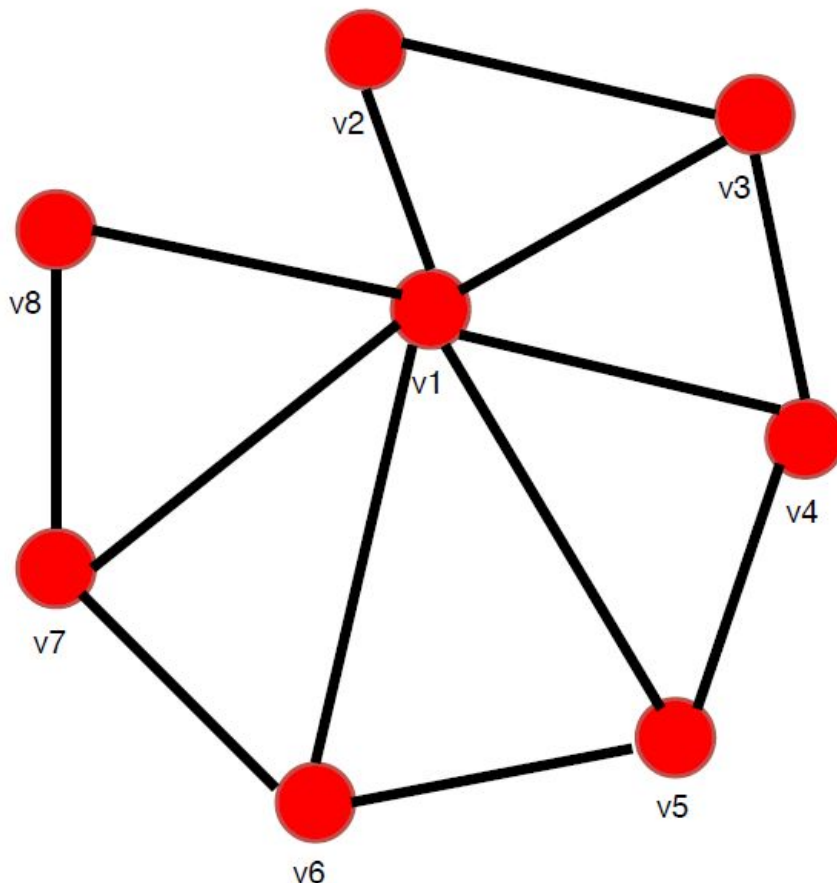
Wektory normalne

Model oświetlenia



Potok renderowania geometrii

TRIANGLE FAN



Ray tracing

Potok renderowania geometrii

Test głębokości

Oświetlenie lokalne i globalne

Cieniowanie

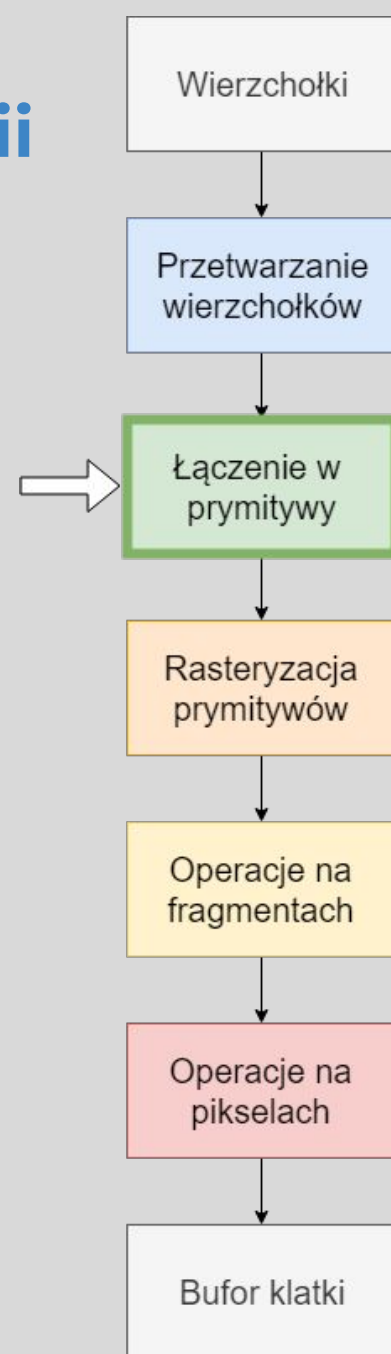
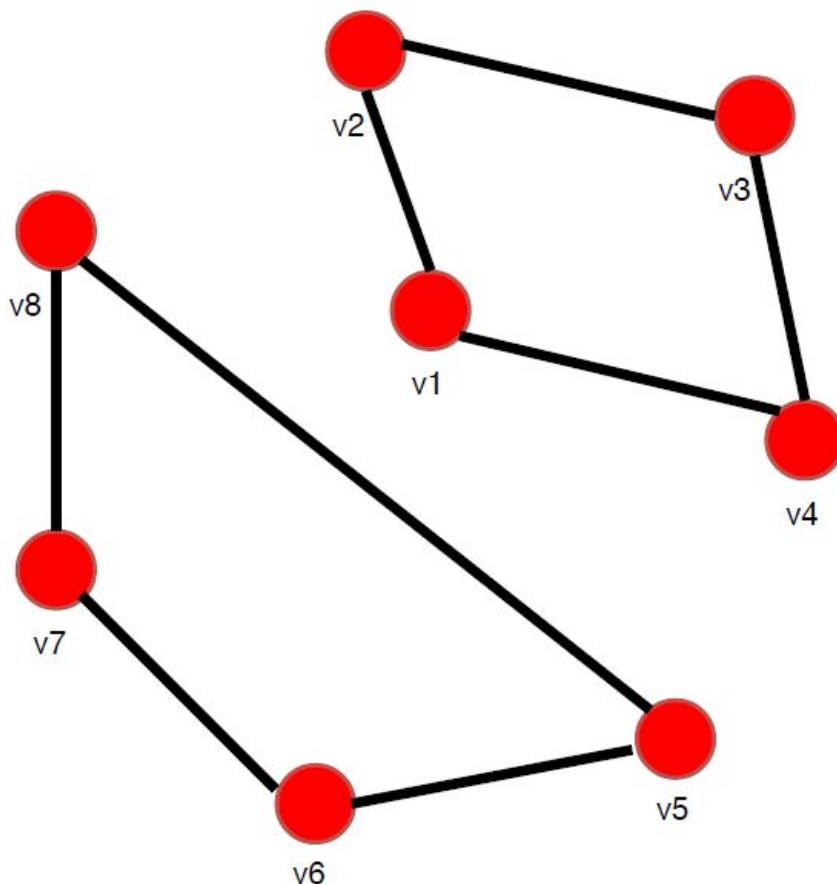
Wektory normalne

Model oświetlenia



Potok renderowania geometrii

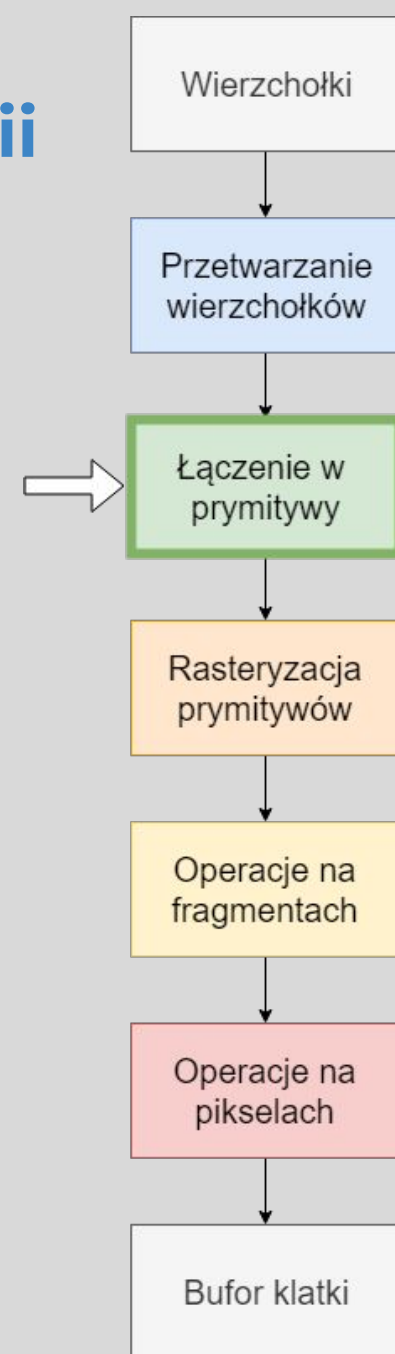
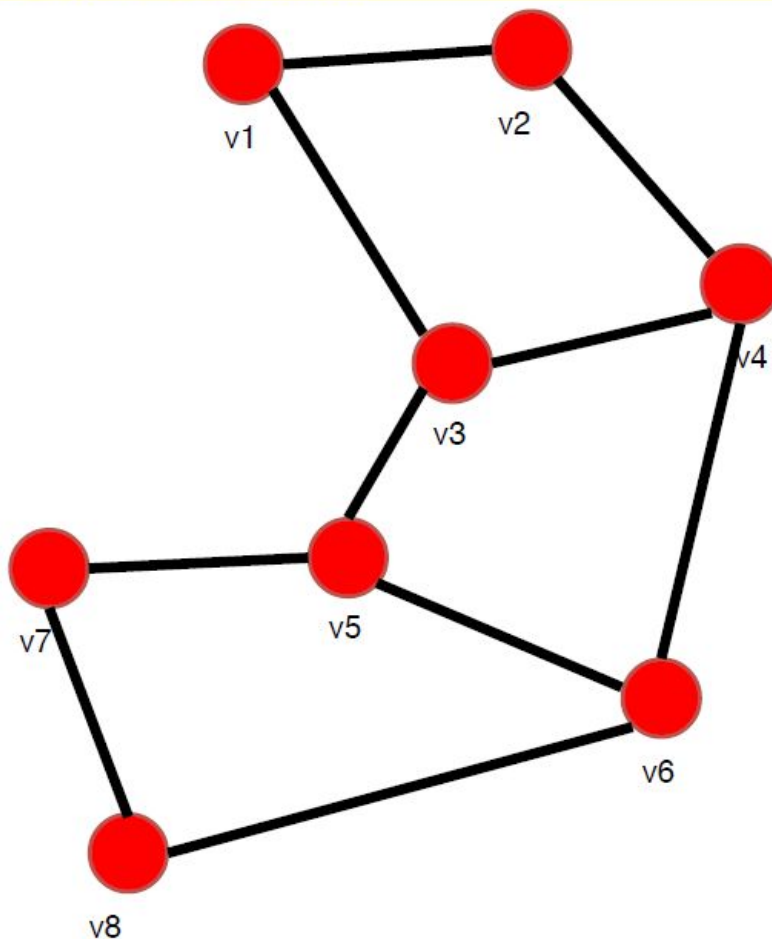
QUADS





Potok renderowania geometrii

QUAD STRIP



Ray tracing

Potok renderowania geometrii

Test głębokości

Oświetlenie lokalne i globalne

Cieniowanie

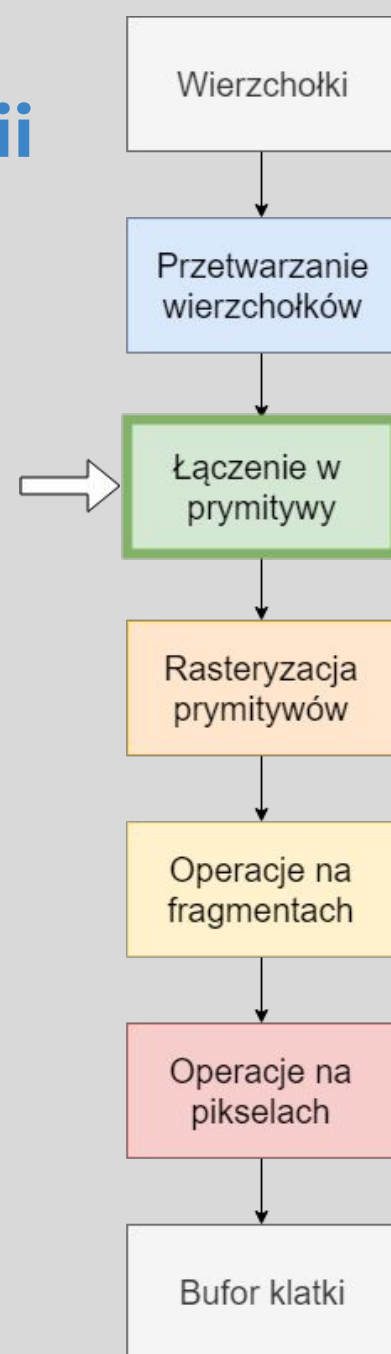
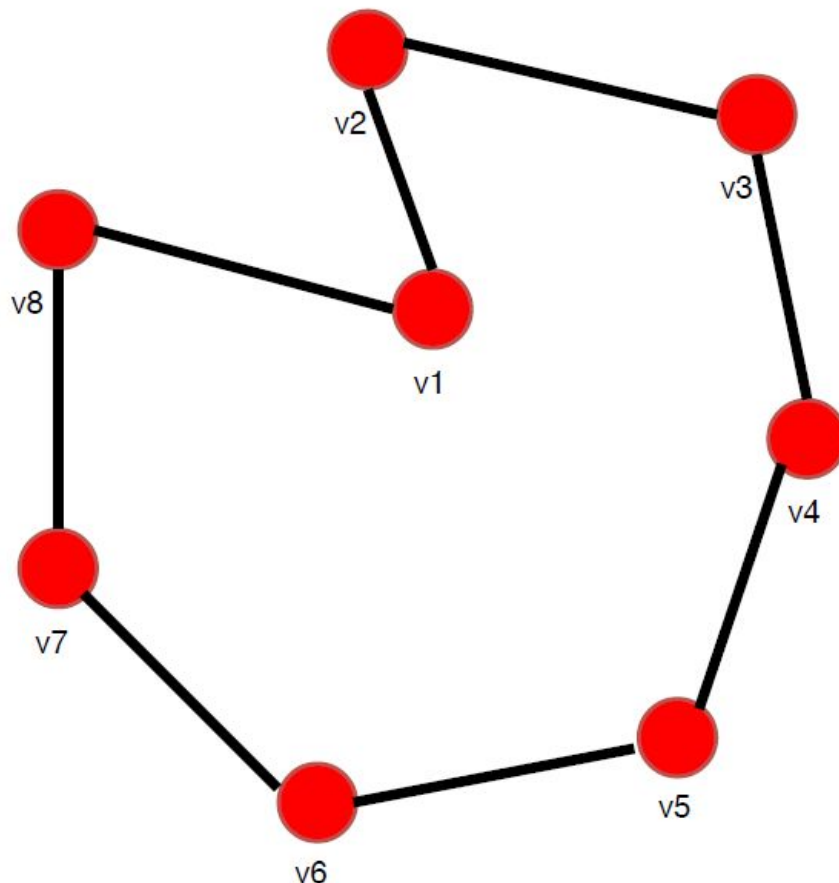
Wektory normalne

Model oświetlenia



Potok renderowania geometrii

POLYGONS



Ray tracing

Potok renderowania geometrii

Test głębokości

Oświetlenie lokalne i globalne

Cieniowanie

Wektory normalne

Model oświetlenia



Potok renderowania geometrii

Ray tracing

Potok renderowania geometrii

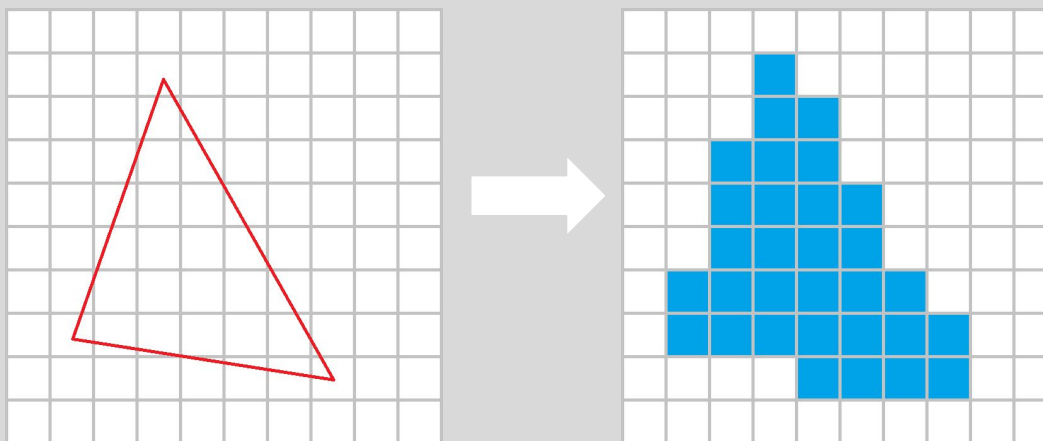
Test głębokości

Oświetlenie lokalne i globalne

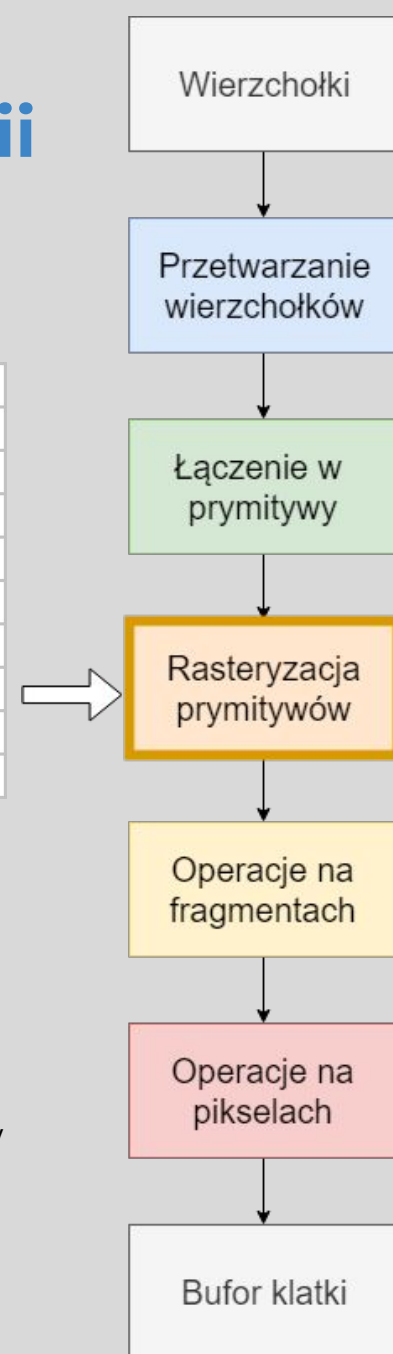
Cieniowanie

Wektory normalne

Model oświetlenia

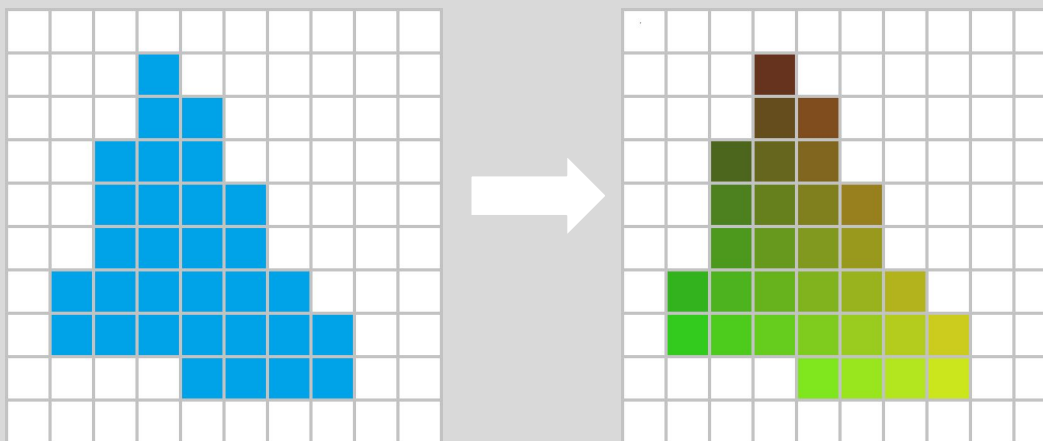


- Generowanie **fragmentów** poprzez **próbkowanie** prymitywów w przestrzeni ekranu.
- **Fragment** to **próbka powierzchni prymitywu**, odpowiadająca **jednemu pikselowi bufora klatki**.
- W procesie rasteryzacji wartości atrybutów wierzchołków są **interpolowane**.

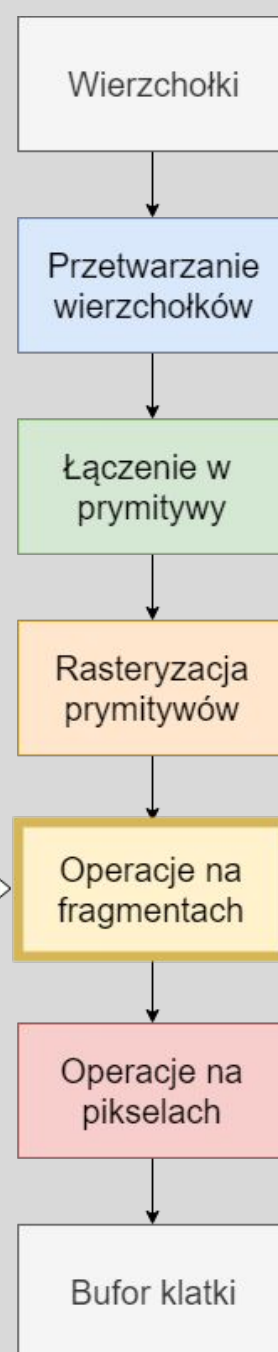




Potok renderowania geometrii



- Wyznaczenie **wartości fragmentu**, która zostanie wzięta pod uwagę w wypełnianiu bufora klatki.
- Oblicza się go na podstawie :
 - **Koloru wierzchołków**
 - **Oświetlenia**
 - **Tekstury**



Potok renderowania geometrii

Ray tracing

Potok renderowania geometrii

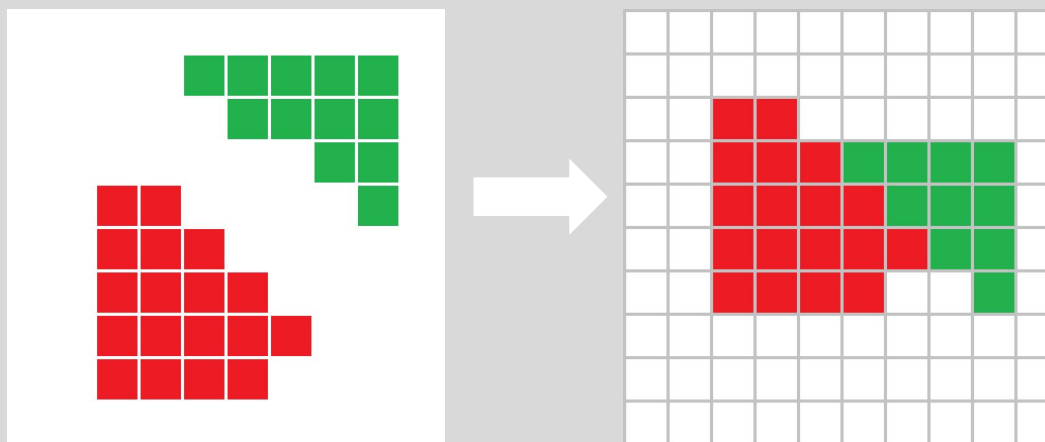
Test głębokości

Oświetlenie lokalne i globalne

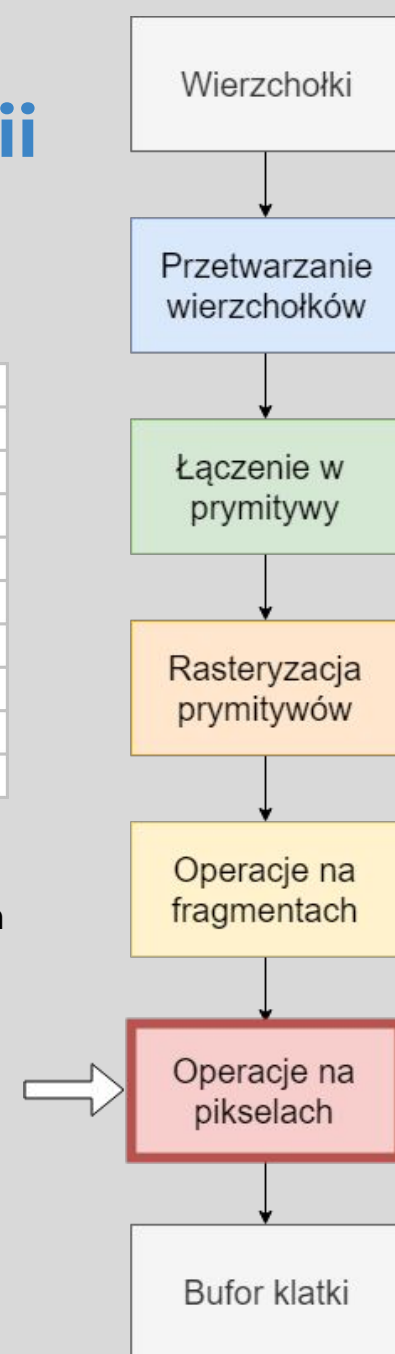
Cieniowanie

Wektory normalne

Model oświetlenia



- **Wypełnienie bufora klatki** na podstawie wyliczonych fragmentów dla wszystkich prymitywów.
- Stosowane operacje :
 - **Test głębokości** (algorytm bufora Z)
 - **Alpha blending** (przezroczystość)
 - **Anti-aliasing**

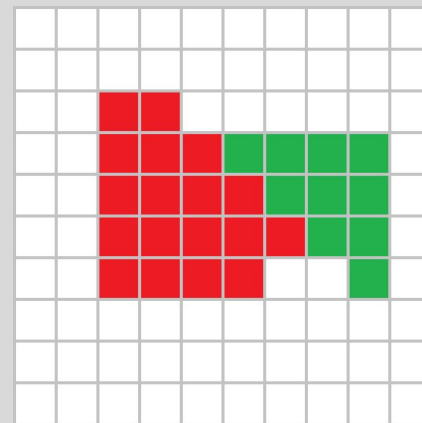
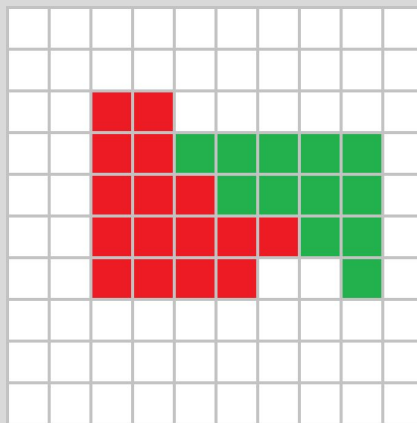
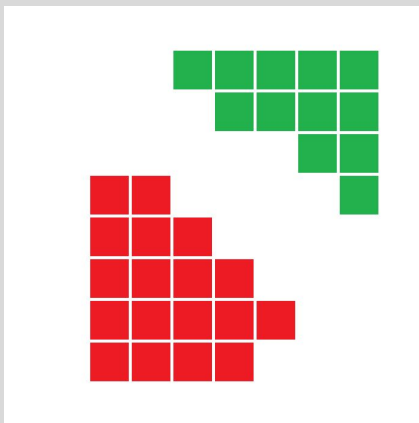
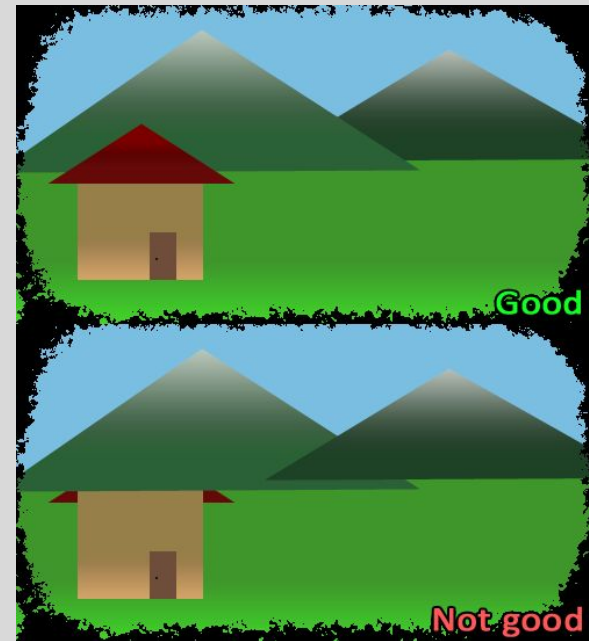




Test głębokości

- Algorytm malarza

- (ang. *painter's algorithm*)
- Nie jest dokonywany test głębokości.
- Obiekty przykrywają się **zgodnie z kolejnością rysowania**
- To co zostanie narysowane później, **przykryje to co zostało narysowane wcześniej.**





Test głębokości

- We fragmentach, oprócz wartości RGB zawierający jego kolor, tworzony jest **bufor głębokości** (bufor Z).
- W buforze Z przechowywana jest informacja o **odległości od kamery**.
- Gdy w procesie renderingu obiektu sceny okaże się, że w danym pikselu jest już zawarty kolor, wykonywany jest test głębokości:
 - Jeśli **odległość** od kamery nowego fragmentu jest **mniejsza niż wartość w buforze głębokości** to **nadpisujemy** ten piksel nową wartością koloru i odległości od kamery.
 - **Nowy obiekt jest bliżej kamery** niż zawarty w buforze koloru.
 - Jeśli **odległość** od kamery nowego fragmentu jest **większa niż wartość w buforze głębokości** to **ignorujemy** tą wartość.
 - **Nowy obiekt jest dalej kamery** i będzie przysłonięty.

Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

Wektory normalne

Model oświetlenia



Oświetlenie

Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

Wektory normalne

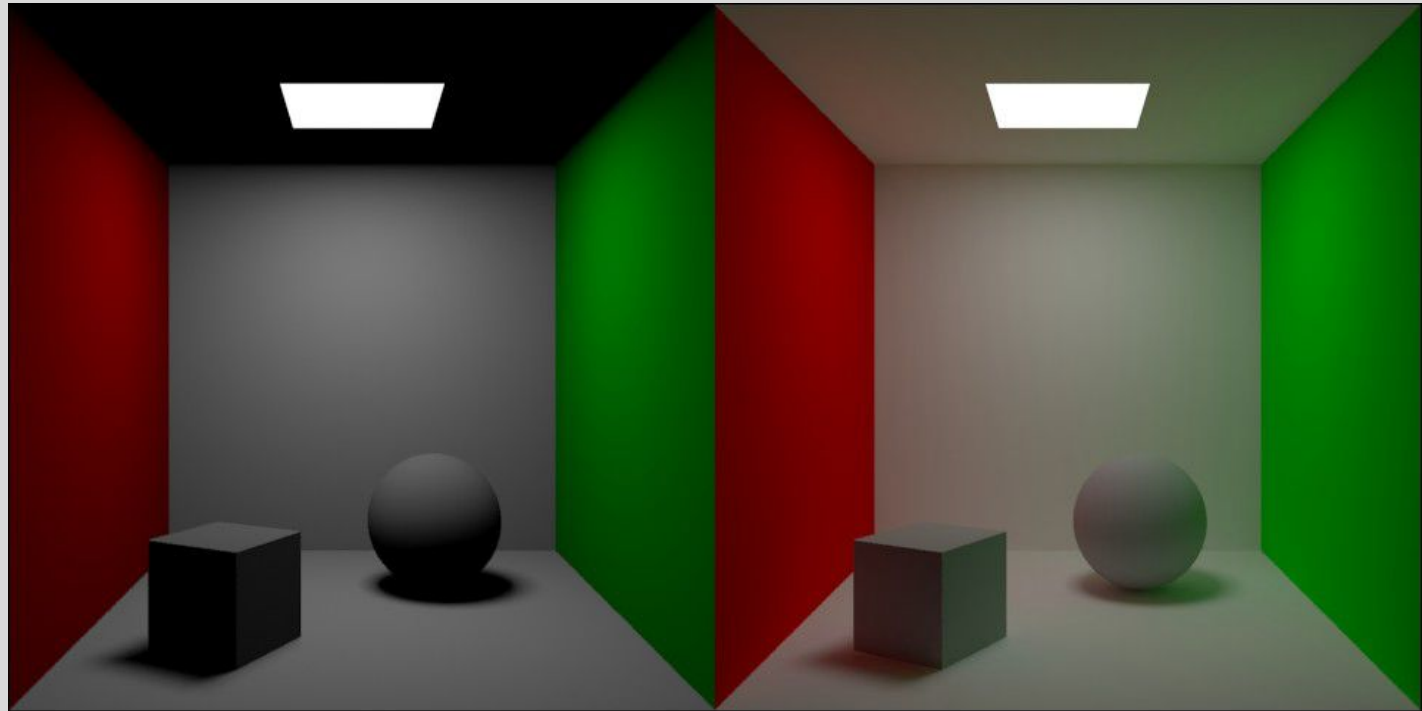
Model oświetlenia



- Zastosowanie **technik oświetlenia** ma na celu **zwiększenie realizmu** renderowanych scen.
- Bez stosowania żadnej techniki oświetlenia, obiekty sceny po wyrenderowaniu wydają się płaskie.

Oświetlenie

- Techniki oświetlenia można podzielić na dwa rodzaje :
 - Oświetlenie **lokalne** (ang. *direct illumination*)
 - Oświetlenie **globalne** (ang. *indirect illumination*)



Oświetlenie lokalne

Oświetlenie globalne

Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

Wektory normalne

Model oświetlenia



Oświetlenie lokalne

- W **oświetleniu lokalnym** na kolor poszczególnych punktów wpływa **wyłącznie bezpośrednie** (ang. *direct*) oświetlenie ze źródeł światła.
- Najczęściej stosowana jest w **grafice czasu rzeczywistego**.

Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

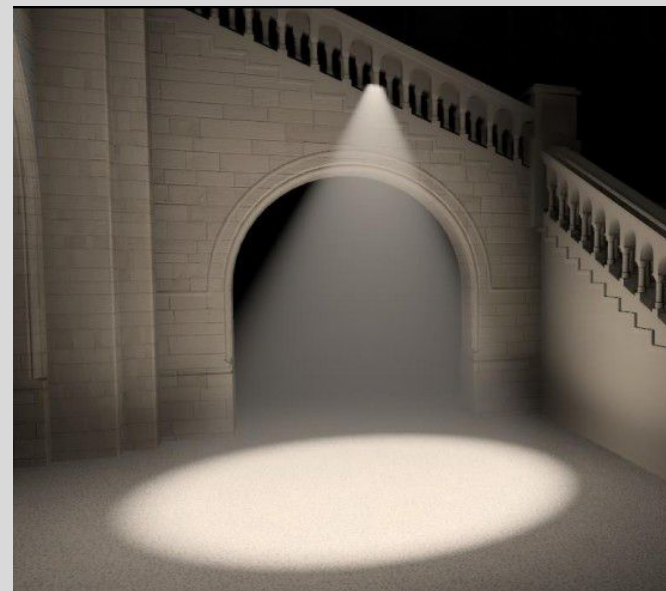
Cieniowanie

Wektory normalne

Model oświetlenia



Oświetlenie lokalne

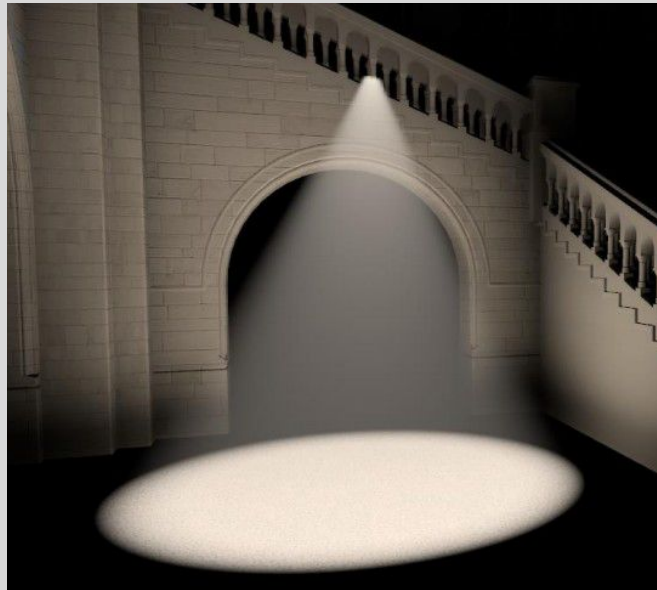


Scena

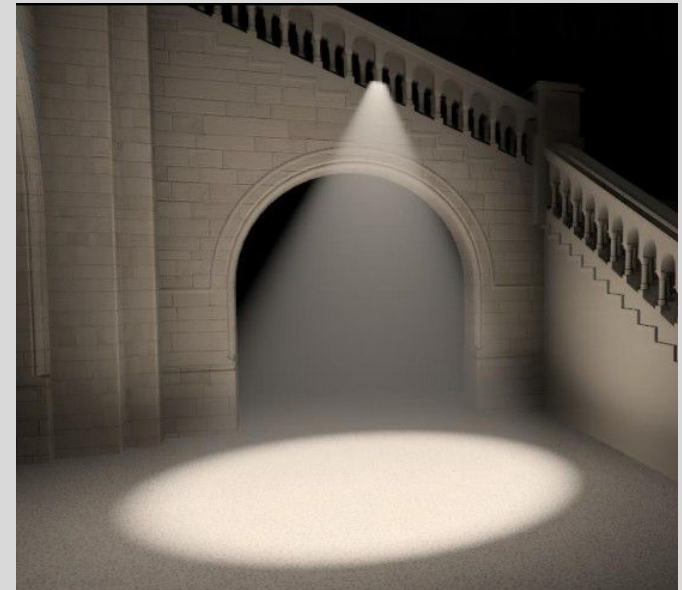


Oświetlenie globalne

- W **oświetleniu globalnym** na kolor poszczególnych punktów wpływa bezpośrednie oświetlenie ze źródeł światła **oraz światło odbite przez inne obiekty** (pośrednie - ang. *indirect*).
- Najczęściej stosowana jest w **grafice realistycznej**.



2 odbicia



3 odbicia

Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

Wektory normalne

Model oświetlenia

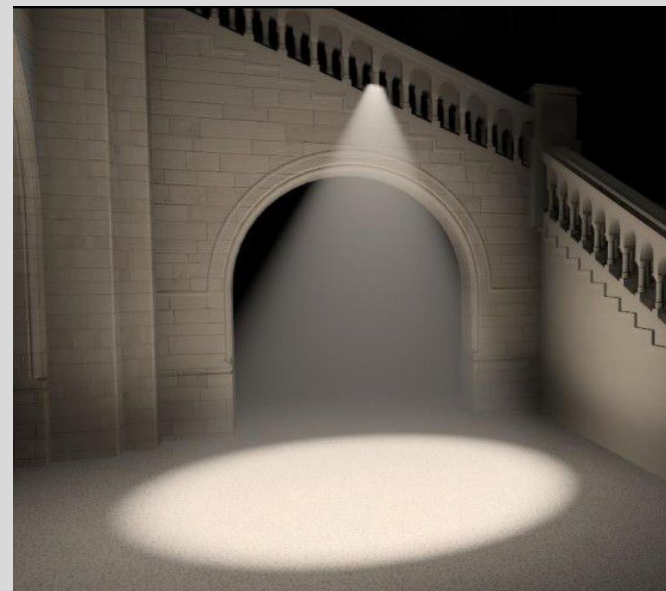


Oświetlenie lokalne a globalne

- Oświetlenie globalne **nie jest używany w grafice czasu rzeczywistego** (bądź jest używane w uproszczonej formie) ponieważ wymagałoby to **śledzenia promienia odbitego**.
- Odbicia są nie są możliwe do odtworzenia ponieważ obliczenia modelu oświetlenia bazujemy na odcinku łączącym fragment obiektu ze źródłem światła.



Oświetlenie lokalne



Oświetlenie globalne

Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

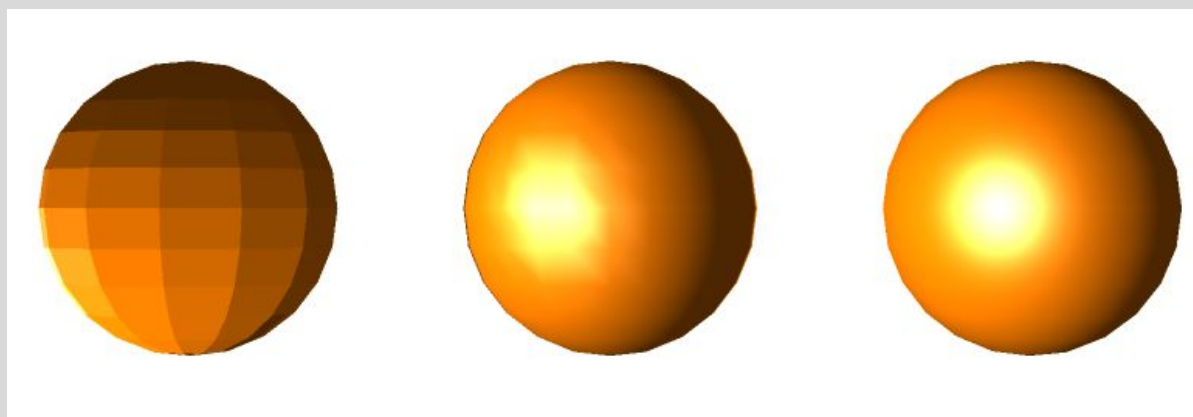
Wektory normalne

Model oświetlenia



Cieniowanie

- Celem symulacji oświetlenia jest modyfikacja kolorów obiektów, aby sprawiały wrażenie oświetlonych fizycznymi źródłami światła.
- Termin **oświetlenie** oznacza model matematycznych pozwalający na wyliczenie koloru bryły uwzględniając jej cechy materiału, źródła światła, a także cechy środowiska.
- Oprócz modelu oświetlenia stosujemy **cieniowanie** (ang. *shading* - nie mylić z *shadowing*!), czyli sposób dystrybucji koloru pomiędzy miejscami, dla których jest on wyliczony.



Płaskie

Gourauda

Phonga

Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

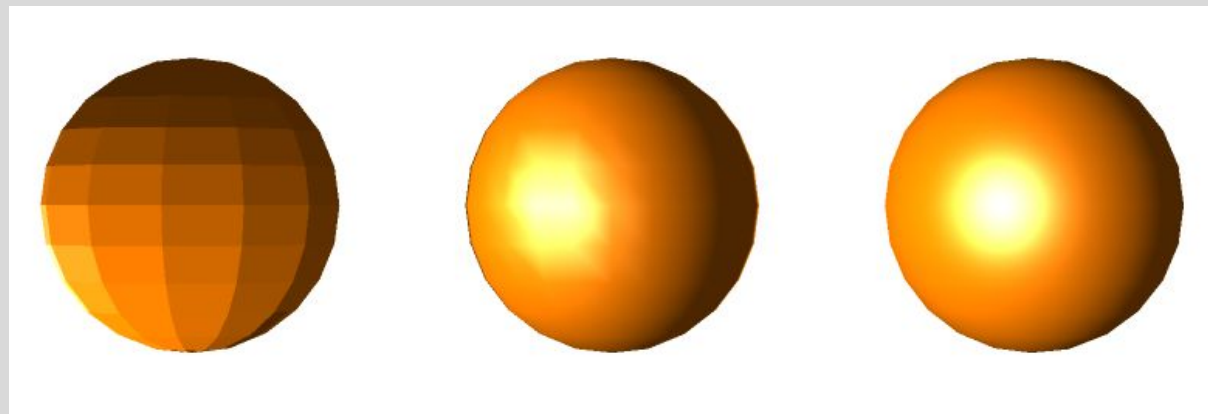
Wektory normalne

Model oświetlenia



Cieniowanie

- Cieniowanie występuje najczęściej w trzech rodzajach :
 - **Cieniowanie płaskie**
 - Liczony jest kolor dla **wierzchołków prymitywu**.
 - Kolor fragmentu jest **średnią** kolorów wierzchołków.
 - **Cieniowanie Gourauda**
 - Liczony jest kolor dla **wierzchołków prymitywu**.
 - Kolor fragmentu jest **interpolowany** z kolorów wierzchołków.
 - **Cieniowanie Phong**
 - Kolor liczony jest dla **każdego fragmentu**.



Płaskie

Gourauda

Phonga

Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

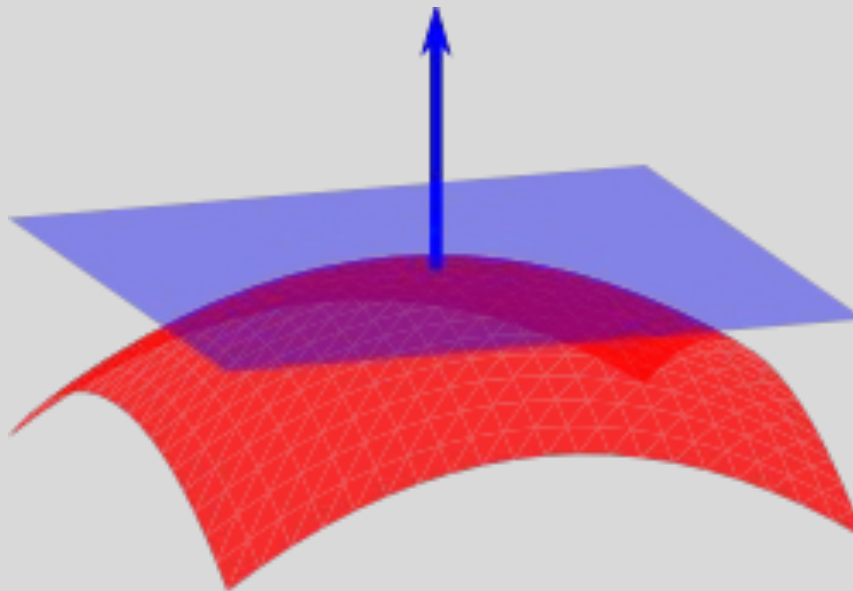
Wektory normalne

Model oświetlenia



Wektory normalne

- Do obliczenia koloru w modelu oświetlenia wykorzystujemy **wektory normalne**.
 - Są to wektory, które są **prostopadłe do powierzchni**.
 - Dla nierównej powierzchni są one **prostopadłe do stycznej**.
 - Są to **wektory jednostkowe**, czyli o **długości równej 1**.
 - **Określa się je dla wierzchołków**.



Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

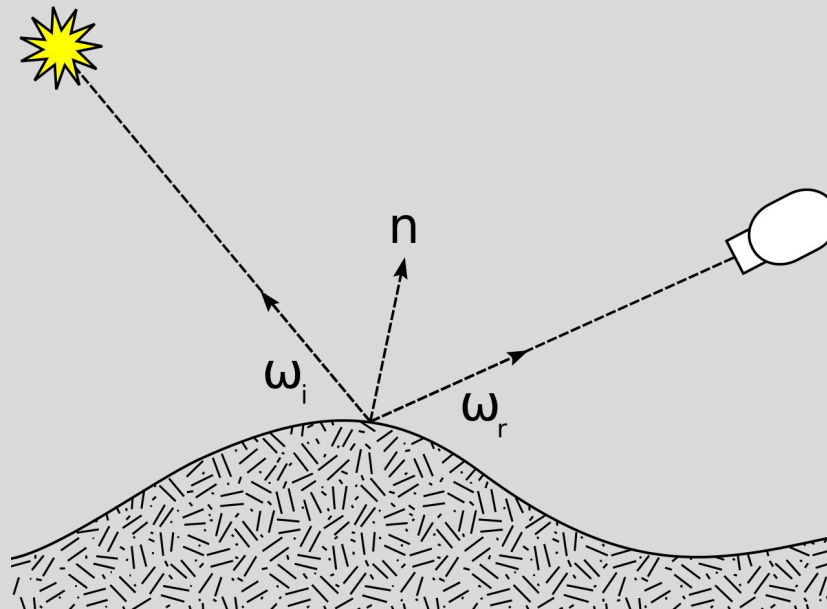
[Wektory normalne](#)

Model oświetlenia



Model oświetlenia

- **Model oświetlenia** ma na celu próbę symulacji zjawisk optycznych.
 - Po obliczeniu powinniśmy otrzymać **radiance** wyjściową po odbiciu światła od powierzchni w danym punkcie na podstawie nadchodzącej fali światła i cech materiału, którym pokryta jest powierzchnia..
 - W grafice realistycznej stosuje się funkcje **BRDF (Bidirectional Reflectance Distribution Function)**.



Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

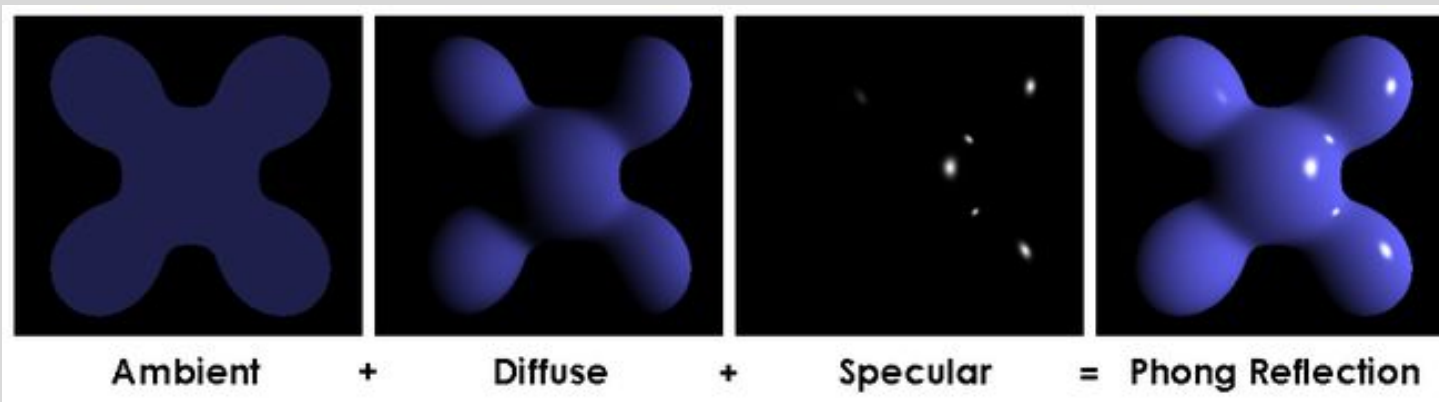
Cieniowanie

Wektory normalne

Model oświetlenia

Model oświetlenia Blinna Phong

- Najczęściej używanym modelem oświetlenia w grafice czasu rzeczywistego jest **model Blinna Phong**.
 - **Nie mylić z cieniowaniem Phong** - to dwie różne rzeczy!
- W modelu Phong rozbijamy światło na **trzy komponenty** :
 - **Ambient** - symulacja światła globalnego
 - **Diffuse** - oświetlenie bezpośrednie
 - **Specular** - odbicie światła





Model oświetlenia Blinna Phong

- Komponenty **ambient**, **diffuse** i **specular** określamy dla **każdego ze źródeł światła**, oraz dla **materiału** dla którego liczymy **kolor powierzchni**.

Ray tracing

Potok renderowania geometrii

Test głębokości

Oświetlenie lokalne i globalne

Cieniowanie

Wektory normalne

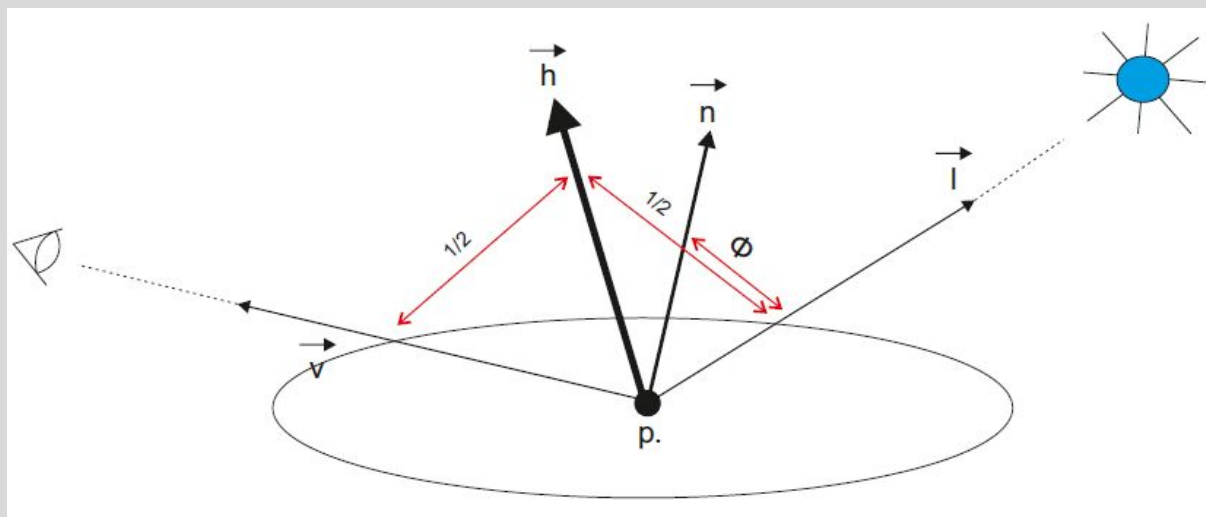
Model oświetlenia

$$L_a = \begin{bmatrix} l_{r_a} \\ l_{g_a} \\ l_{b_a} \end{bmatrix} \quad L_d = \begin{bmatrix} l_{r_d} \\ l_{g_d} \\ l_{b_d} \end{bmatrix} \quad L_s = \begin{bmatrix} l_{r_s} \\ l_{g_s} \\ l_{b_s} \end{bmatrix}$$
$$M_a = \begin{bmatrix} m_{r_a} \\ m_{g_a} \\ m_{b_a} \end{bmatrix} \quad M_d = \begin{bmatrix} m_{r_d} \\ m_{g_d} \\ m_{b_d} \end{bmatrix} \quad M_s = \begin{bmatrix} m_{r_s} \\ m_{g_s} \\ m_{b_s} \end{bmatrix}$$

Model oświetlenia Blinna Phong

- Kolor powierzchni jest to **suma komponentów ambient diffuse i specular dla każdego źródła światła**.

$$I = \sum_{i=1}^N I_a + I_d + I_s$$



Ray tracing

Potok renderowania
geometrii

Test głębokości

Oświetlenie lokalne i
globalne

Cieniowanie

Wektory normalne

Model oświetlenia

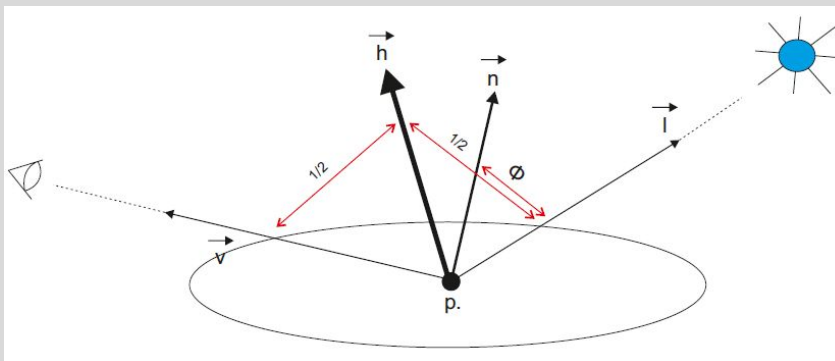


Model oświetlenia Blinna Phong

$$I = \sum_{i=1}^N I_a + I_d + I_s$$

- Składowa **ambient** służy za **BARDZO duże uproszczenie symulacji światła niebezpośredniego**.
- Jest on określany niezależnie od położenia i orientacji światła i obiektu.
- Składową **ambient** liczy się za pomocą zwykłego iloczynu wartości RGB światła i wartości RGB materiału.

$$I_a = M_a \cdot L_a$$

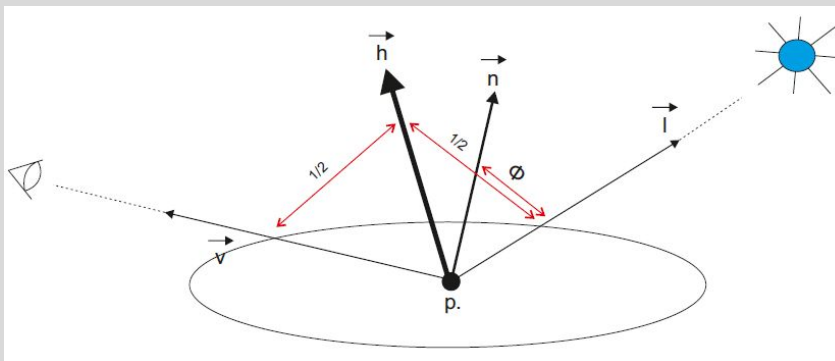


Model oświetlenia Blinna Phong

$$I = \sum_{i=1}^N I_a + I_d + I_s$$

- **Składowa diffuse** dodatkowo uwzględnia kąt padania światła na powierzchnię obiektu.
- Ten kąt oblicza się za pomocą **iloczynu skalarnego wektora normalnego powierzchni oraz kierunku padania światła**.

$$I_d = M_d \cdot L_d \cdot (|\vec{n}| \cdot |\vec{l}|)$$



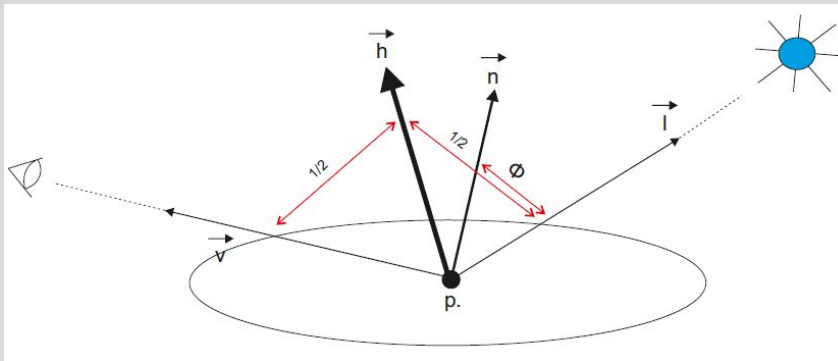


Model oświetlenia Blinna Phong

$$I = \sum_{i=1}^N I_a + I_d + I_s$$

- **Składowa specular** symulująca odbicie światła zależy od iloczynu skalarnego **wektora normalnego i wektora połówkowego wektora obserwacji i padania światła**.
- Dodatkowo dołączony jest parametr **shininess** wskazujący właściwość powierzchni do odbijania światła.

$$I_s = M_s \cdot L_s \cdot (|\vec{n}| \cdot |\vec{h}|)^{M_{shi}}$$



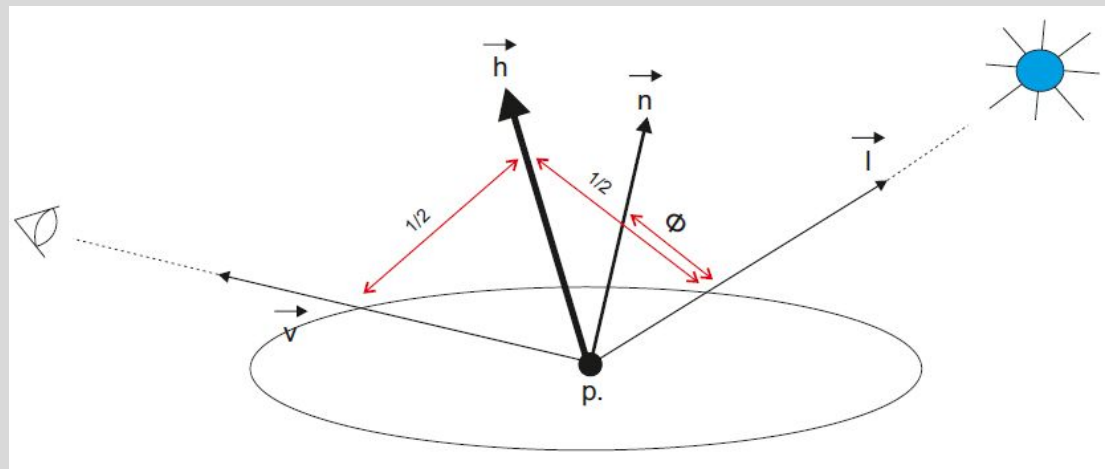
Model oświetlenia Blinna Phong

$$I = \sum_{i=1}^N I_a + I_d + I_s$$

$$I_a = M_a \cdot L_a$$

$$I_d = M_d \cdot L_d \cdot (|\vec{n}| \cdot |\vec{l}|)$$

$$I_s = M_s \cdot L_s \cdot (|\vec{n}| \cdot |\vec{h}|)^{M_{shi}}$$





Grafika Komputerowa

Wykład 5

Dziękuję za uwagę :)



mgr inż. Michał Chwesiuk