

# PREGUNTAS-FRECUENTES-IG-EXAMENES...



**Sr\_Aprobados**



**Informática Gráfica**



**3º Grado en Ingeniería Informática**



**Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación  
Universidad de Granada**



MÁSTER EN

## Inteligencia Artificial & Data Management

MADRID

Formamos  
**talento** para un futuro  
**Sostenible**

saber más



Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandes con una garantía de hasta 100.000 euros por depositante. Consulta más información en [ing.es](https://www.ing.es)



## PREGUNTAS IG EXÁMENES ORDINARIA

Sea la siguiente representación de una malla de triángulos. Describe cómo se puede determinar si la orientación de las normales de las caras es consistente (están orientadas hacia el exterior del objeto)

La orientación de las normales de las caras de una malla de triángulos es consistente si todas las normales apuntan hacia el exterior del objeto. Tenemos varias formas de comprobarlo:

- La primera y más sencilla es ponerle iluminación al objeto, y si no sale correctamente, es que no están bien calculadas las normales.
- Otra opción es calcular un vector que va desde el centro de la figura hacia las caras y compara el sentido de este vector con el vector normal de cada cara. Si el sentido es diferente, se considera que la normal está orientada hacia el interior del objeto.

(También hay veces que nos interesa que las normales apunten hacia el interior del objeto cuando queremos hacer iluminación dentro de este)

La imagen siguiente corresponde a la visualización de un poliedro regular de radio 1, centrado en el origen que aproxima una esfera de color azul medio.



Indica:

- Qué fuentes de luz y que propiedades de material se han usado para generarla.
- Qué cambios se deben hacer en el código para que la visualización simule una esfera, indicando cómo se calcularían las normales.

Tiene dos fuentes de luz, una por encima del eje Y con X negativa y otra con Y negativa y X positiva. Los materiales tienen componentes especulares, ya que se puede observar el reflejo de las fuentes de luz; y difusa, por la apariencia mate de color azul. No tiene componente ambiental ya que se puede observar que hay partes negras, no iluminadas.

Consulta condiciones aquí



do your thing

WUOLAH

Actualmente se ha realizado una iluminación plana del objeto, pero al ser curvo, habría que añadir en el código al final de la construcción del objeto:

- `calcular_normales_vertices()`; Como es una esfera, no necesitamos calcular la normal de las caras sino que directamente podemos calcular, para cada uno de los vértices, la normal dividiendo el vector que une el centro de la esfera con el vértice “i” entre el módulo.
- `colors_diffuse_gouraud(...)`: para realizar el suavizado una vez calculados los vértices.

Describe las formas de interacción de la luz como partícula según la superficie de los objetos y el proceso que se sigue para realizar un suavizado de Gouraud.

- Reflexión especular: ocurre en superficies lisas y pulidas, donde los rayos se reflejan en una sola dirección. Se produce una apariencia brillante. Depende del observador. Ejemplo de objeto: espejo.
- Reflexión difusa: tiene lugar en superficies rugosas, donde los rayos se reflejan en todas direcciones, proporcionando una apariencia mate. No depende del observador. Ejemplo: pared.

El suavizado de Gouraud es útil para objetos curvos. Para obtenerlo, hay que calcular las normales a los vértices, calculando un promedio de las normales de las caras para cada vértice que compartan estas caras. Se calcula el color de cada vértice y lo interpola suavemente por las aristas, pero no representa bien los reflejos especulares.

Explique cómo funciona la Iluminación en OpenGL.

- Componente especular: modela la reflexión en superficies lisas y pulidas, donde los rayos se reflejan en una sola dirección. Se produce una apariencia brillante. Ejemplo de objeto: espejo. Depende del observador.
- Componente difusa: modela la reflexión en superficies rugosas, donde los rayos se reflejan en todas direcciones, dando una apariencia mate. Ejemplo de objeto: una pared.

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en [ing.es](http://ing.es)

Que te den **10 € para gastar**  
es una fantasía.  
ING lo hace realidad.

Abre la **Cuenta NoCuenta** con el código  
WUOLAH10, haz tu primer pago y llévate 10 €.

**Quiero el cash**

[Consulta condiciones aquí](#)



do your thing

# Informática Gráfica



**Comparte estos flyers en tu clase y consigue más dinero y recompensas**



**Banco de apuntes de la**

- 1** Imprime esta hoja
- 2** Recorta por la mitad
- 3** Coloca en un lugar visible para que tus compis puedan escanar y acceder a apuntes

- 4** Llévate dinero por cada descarga de los documentos descargados a través de tu QR



● **Componente ambiental:** es constante. Afecta uniformemente a todos los objetos en la escena, evitando que los objetos se vean negros (sin luz).

Indique cuales son los parámetros para definir la cámara y cómo se usan para obtener la transformación de vista.

Una cámara es un objeto de la escena para crear una imagen de la escena. Para definir una cámara se debe establecer: el punto de mira (PM), que es la posición a la que apunta la cámara, el vector de inclinación de la cámara (VI) y la posición de la cámara, que será el CP (centro de proyección). Para definir la cámara también hay que indicar el tipo de proyección: paralela (glOrtho) o perspectiva (glFrustum) y la orientación: frontal, en picado, contrapicado, cenital o supina.

En la transformación de vista 3D, el modelo se transforma desde el sistema de coordenadas mundiales 3D a coordenadas de vista 3D (observador). Los parámetros mencionados de la cámara afectan directamente a esta transformación ya que dependiendo del PM, VI y CP de la cámara y el tipo de proyección, la transformación se realizará de una forma u de otra.

Describa las transformaciones de vista que se pueden aplicar a una cámara.

En la transformación de vista 3D, el modelo se transforma desde el sistema de coordenadas mundiales 3D a coordenadas de vista 3D (observador).

A la hora de realizar la transformación de vista, se coloca la cámara, indicando el PM, VI y CP. Además, puede tener una proyección paralela, con glOrtho o una proyección en perspectiva, con glFrustum.

Indique la matriz de OpenGL que almacena dicha transformación de vista

GL\_MODELVIEW



Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa



1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en [ing.es](https://www.ing.es)

### Explica el funcionamiento del Z-buffer

El algoritmo del Z-buffer es del tipo espacio-imagen. Cada vez que se va a renderizar un píxel, comprueba que no se haya dibujado antes en esa posición un píxel que esté más cerca respecto a la cámara (posición en eje Z). Este algoritmo funciona bien para cualquier tipo de objetos: cóncavos, convexos, abiertos y cerrados. Cuando dibujamos un objeto, la profundidad de sus píxeles se guardan en este buffer. Si utilizamos dichos píxeles en pantalla para dibujar otro objeto, se producirá una comparación de las profundidades de dichos píxeles. Si la profundidad del último píxel es mayor que la nueva (está más lejos) el píxel nuevo no se pinta, mientras que si está más cerca (la profundidad es menor), se dibuja el píxel y se guarda la profundidad en el z-buffer.

### ¿Qué son las primitivas?

Las primitivas son los elementos básicos para construir un modelo: puntos, líneas y polígonos.

Indica los pasos que hay que realizar en OpenGL y los elementos que intervienen y por tanto han de estar definidos para conseguir que una escena se vea iluminada

Definir los vectores normales de cada cara: Es necesario definir un vector normal (perpendicular a la superficie apuntando hacia fuera de la parte visible) por cada uno de los vértices de nuestra representación.

Situar las luces: Para iluminar una escena será necesario situar las luces. OpenGL maneja varios tipos de iluminación:

- Luz ambiental: ilumina toda la escena por igual, ya que esta no proviene de una dirección predeterminada.
- Luz difusa: viene de una dirección específica, y depende de su ángulo de incidencia para iluminar una superficie en mayor o menor medida.
- Luz especular: representa los reflejos brillantes que se observan en superficies pulidas. Depende de la posición de la

Consulta condiciones aquí



do your thing

WUOLAH

fuente de luz, la posición del observador y las propiedades especulares del material.

Definiendo materiales: OpenGL permite controlar la forma en que la luz se refleja sobre nuestros objetos, que es lo que se conoce como definición de materiales.

Describe el flujo de transformaciones que se realizarán en OpenGL desde que proporcionamos las coordenadas 3D de un modelo hasta que tenemos una imagen en pantalla. Indique el propósito de cada etapa y el resultado obtenido tras cada una de las transformaciones.

1. Transformación del modelo: Situarlo en escena, cambiarlo de tamaño y crear modelos compuestos de otros más simples.
2. Transformación de vista: Poner al observador en la posición deseada.
3. Transformación de perspectiva: Pasar de un mundo 3D a una imagen 2D.
4. Rasterización: Calcular por cada pixel su color, teniendo en cuenta la primitiva que se muestra, su color, material, texturas, luces, etc.
5. Transformación del dispositivo: Adaptar la imagen 2D a la zona de dibujado.

Explique el sistema de colores que se usa en OpenGL.

El sistema de colores empleado en OpenGL es el sistema RGB. RGB significa los colores rojo, verde y azul: los colores primarios aditivos. A cada uno de estos colores se le asigna un valor en OpenGL generalmente un valor entre 0 y 1. El valor 1 significa la mayor cantidad posible de ese color, y 0 significa ninguna cantidad de ese color. Podemos mezclar estos tres colores para obtener una gama completa de colores.

Enumere y explique las propiedades de la transformación de perspectiva

Acortamiento perspectivo: objetos más lejanos producen una proyección más pequeña.



Punto de fuga: cualquier par de líneas paralelas convergen en un punto llamado punto de fuga.

Inversión de vista: los puntos que están detrás del centro de proyección se proyectan invertidos.

Distorsión topológica: cualquier elemento geométrico que tenga una parte delante y otra detrás del centro de proyección produce dos proyecciones semiinfinitas.

Explique, lo más detallado posible, las distintas formas de realizar un pick.

Hay tres formas de realizarlo:

- Identificación por color: A cada objeto que se quiere identificar se le asigna un identificador, un número natural. Este número es convertido a un color. Se activa la eliminación de partes ocultas. Cuando se dibuja el objeto, se usa el color que tiene asociado. Al mover el cursor y pulsar para realizar la selección, se guardan las coordenadas  $x$  e  $y$  del pixel seleccionado. Se lee el pixel del buffer en la posición del buffer en la posición  $x$  e  $y$ . Se convierte al color seleccionado.
- Lanzado de un rayo: La idea básica es que cuando pulso un botón del ratón, voy a seleccionar el objeto más cercano que está en la posición del cursor. Para ello, obtenemos la posición  $x$  e  $y$  del cursor en coordenadas del dispositivo y las convertimos en coordenadas de vista. Hacemos pasar una línea recta por el centro de la proyección y la nueva posición. Esto es, obtenemos la ecuación de una recta. Calculamos la intersección con los objetos. Si hay intersección, se añade a la lista guardando el ID del objeto y la profundidad. Por último, ordenamos en profundidad y devolvemos el ID de la intersección más cercana.
- Por ventana: Este método consiste en aprovechar la etapa de discretización del cauce visual. Esto es, cuando pasamos de fórmulas a pixeles. La idea consiste en que cuando se marca la posición con el ratón, se obtiene la posición  $x$  e  $y$ . Alrededor de

Esto no son apuntes pero **tiene un 10 asegurado** (y lo vas a disfrutar igual).

Abre la Cuenta NoCuenta con el código **WUOLAH10**, haz tu primer pago y llévate 10 €.

Me interesa

1/6

Este número es indicativo del riesgo del producto, siendo 1/6 indicativo de menor riesgo y 6/6 de mayor riesgo.

ING BANK NV se encuentra adherido al Sistema de Garantía de Depósitos Holandés con una garantía de hasta 100.000 euros por depositante. Consulta más información en [ing.es](https://www.ing.es)



dicha posición se crea una pequeña ventana. Una vez identificados los píxeles que conforman la ventana, lo único que hay que hacer es dibujar cada objeto al cual se le asigna un identificador. Si al convertir el objeto en píxeles, coincide con alguno o varios de la ventana, entonces hay selección. Se guarda el identificador del objeto y la profundidad. Finalmente, podemos hacer una ordenación por profundidad y quedarnos con el identificador más cercano.

### Explique el Ray Tracing

Permite de una sola vez generar imágenes con eliminación de partes ocultas, proyección en perspectiva, sombras y simular modelos de iluminación empíricos globales.

Consiste en simular el recorrido de un rayo de luz en sentido inverso, desde el ojo de un observador hacia una escena, calculando las intersecciones del rayo con los objetos.

El método funciona en el espacio imagen. Para cada píxel, de la imagen se crea un rayo con origen en el observador y que pase por el centro de dicho píxel. Se encuentra las intersecciones con ls

Ray Tracing permite generar imágenes realistas al mismo tiempo que elimina partes ocultas, proyecta en perspectiva, genera sombras y simula modelos de iluminación globales. Consiste en simular el recorrido de un rayo de luz en sentido inverso, desde el ojo del observador hacia la escena, calculando las interacciones del rayo con los objetos (reflexión, refracción y sombras).

### Explica qué hace la síntesis 3D

1. Transformación de vista 3D: El modelo pasa del sistema de coordenadas mundiales 3D al sistema de coordenadas de vista 3D (observador). Esto puede hacerse de dos maneras:
  - o Mediante rotaciones del sistema de coordenadas de vista hasta alinearlo con el sistema de coordenadas mundiales.

Consulta condiciones aquí



do your thing

WUOLAH

- Calculando una matriz de transformación usando la propiedad de ortogonalidad de Euler.
- 2. Recortado 3D frente al volumen de vista: Se eliminan las partes de la escena que el observador no puede ver. Esto se realiza creando un volumen de vista, aunque en algunos casos, como en Ray Tracing, no es necesario realizar este paso.
- 3. Proyección a 2D: La escena 3D se proyecta en un espacio 2D, generando una representación con coordenadas en el plano 2D.
- 4. Transformación de vista 2D: Se transforma el modelo desde las coordenadas 2D mundiales a las coordenadas de dispositivo (CD), introduciendo el concepto de viewport para mapear la escena a la pantalla.
- 5. Rasterización: Los elementos geométricos 2D se convierten en píxeles, produciendo la imagen final que se visualizará en el dispositivo.