**1.- Propuesta Educativa para Mate 3 (Estudiantes de Ing. en Computación )**

**Contexto:** Se les plantea una situación interesante que combina geometría analítica con visualización en 3D, cálculos matemáticos, y programación, que son habilidades valiosas en muchos campos de la computación, como la gráfica computacional, la simulación física, y la realidad aumentada o virtual.

**a) Verificación Geométrica y Visualización**

Objetivo: Demostrar que la curva r(t) = (t cos(t), t sin(t), t) se encuentra sobre un cono, y visualizar esta curva y el cono usando herramientas de gráficos 3D.

**Actividad:**

1. Demostración Matemática: Comienza probando que la curva se encuentra sobre la superficie de un cono. Para ello, utiliza la ecuación del cono (x^2 + y^2 = z^2). Sustituye las componentes de r(t) en esta ecuación y verifica si se cumple.

2. Implementación de Visualización: Usa una herramienta de software como Python con bibliotecas como Matplotlib o MayaVi (MayaVi es un visualizador de datos científicos escrito en Python, que utiliza VTK y proporciona una GUI a través de Tkinter), para crear una representación visual del cono y la curva. Esto les ayudará a entender mejor las relaciones espaciales y practicar habilidades de programación gráfica.

**b) Intersección con una Esfera**

Objetivo: Determinar el punto de intersección de la curva con la esfera dada por la ecuación (x^2 + y^2 + z^2 = 2).

**Actividad:**

1. Cálculo de Intersección: Desarrolla un script que inserte las componentes de r(t) en la ecuación de la esfera y resuélvala para (t). Esto puede hacerse mediante métodos numéricos o analíticos.

2. Visualización de la Intersección: Mejora tu script de visualización para marcar el punto de intersección en la gráfica. Esto no solo confirma el cálculo, sino que también proporciona una representación visual clara del evento.

**c) Longitud de la Curva**

Objetivo: Calcular la longitud de la curva desde el origen hasta el punto de intersección.

**Actividad:**

1. Cálculo Analítico: Utiliza la fórmula de longitud de arco de una curva paramétrica y resuelva la integral para el intervalo desde el origen hasta el punto de intersección.

2. Implementación Computacional: Implementar el cálculo de la longitud de arco usando métodos numéricos en Python. Este ejercicio fortalece la capacidad de aplicar métodos numéricos y el uso de software matemático.

**Evaluación y Resultados Esperados:**

- Presentación: Cada estudiante o grupo (no más de 2 personas) presentará su código, mostrando tanto la curva resultante como la longitud calculada. Deberán explicar el impacto de la posición de los puntos de control en la forma y longitud de la curva.

- Informe: Entregar un informe que incluya el código fuente, capturas de pantalla de la interfaz gráfica, una explicación de los métodos utilizados para calcular la longitud y un análisis de los resultados.

- Traer una maqueta que describa el fenómeno, con los elementos más simples que pueda obtener.

**Conclusión y Aplicación Profesional**

Este proyecto no solo aborda el problema matemático en cuestión, sino que también permite a los estudiantes de computación aplicar sus habilidades en un contexto práctico y visual. Al finalizar, los estudiantes no solo habrán aprendido a resolver un problema complejo de geometría, sino que también habrán ganado experiencia valiosa en visualización de datos 3D y programación matemática, habilidades altamente pertinentes en áreas como desarrollo de videojuegos, simulaciones, realidad virtual, y más.

**2.- Implementación y Análisis de Curvas de Bézier**

**Objetivos de la Actividad:**

1. Comprender y aplicar los principios matemáticos detrás de las curvas de Bézier.

2. Desarrollar habilidades en programación gráfica y matemática computacional.

3. Implementar algoritmos para el dibujo y análisis de curvas en un ambiente de programación.

**Descripción de la Actividad:**

Los estudiantes desarrollarán una aplicación en Python utilizando bibliotecas como `matplotlib` o `numpy` para crear y visualizar curvas de Bézier. Además, calcularán la longitud de estas curvas usando métodos numéricos.

Parte 1: Dibujar una Curva de Bézier

- Tarea 1: Investigar sobre los polinomios de Bézier y su implementación matemática.

- Tarea 2: Escribir una función en Python que acepte puntos de control como entrada y produzca la visualización de una curva de Bézier cuadrática o cúbica. Utilizar `matplotlib` para graficar la curva y los puntos de control.

- Tarea 3: Permitir que el usuario interactúe con los puntos de control (usando por ejemplo `ipywidgets` para interfaces interactivas en Jupyter Notebook) para ver cómo afectan la forma de la curva.

Parte 2: Calcular la Longitud de la Curva de Bézier

- Tarea 4: Investigar métodos para calcular la longitud de una curva de Bézier, como la integración numérica de la norma del vector derivado.

- Tarea 5: Implementar una función que calcule la longitud de la curva de Bézier basada en la parametrización de la curva y su derivada. Se pueden utilizar métodos de integración numérica disponibles en `numpy` o `scipy`.

- Tarea 6: Comparar la longitud de diferentes curvas de Bézier al modificar los puntos de control y discutir cómo la forma de la curva afecta su longitud.

**Evaluación y Resultados Esperados:**

- Presentación: Cada estudiante o grupo (no más de 2 personas) presentará su código, mostrando tanto la curva resultante como la longitud calculada. Deberán explicar el impacto de la posición de los puntos de control en la forma y longitud de la curva.

- Informe: Entregar un informe que incluya el código fuente, capturas de pantalla de la interfaz gráfica, una explicación de los métodos utilizados para calcular la longitud y un análisis de los resultados.

- Traer una maqueta que describa el fenómeno, con los elementos más simples que pueda obtener.

**Competencias por Desarrollar:**

- Habilidades de programación: Uso eficaz de Python para resolver problemas complejos.

- Visualización de datos: Capacidad para visualizar y manipular gráficamente datos geométricos.

- Matemática computacional: Aplicación de técnicas numéricas para la integración y análisis matemático.

**Aplicación Profesional:**

Este proyecto no solo es fundamental para aquellos interesados en gráficos por computadora, sino también en campos como el diseño CAD, animación, simulaciones físicas y desarrollo de videojuegos, donde las curvas de Bézier son herramientas esenciales.

**Recursos Adicionales:**

- Documentación de `matplotlib` y `numpy`.

- Artículos y tutoriales sobre curvas de Bézier y métodos de integración numérica.

- Ejemplos de código en plataformas como GitHub o Stack Overflow.

Este proyecto ofrece una excelente oportunidad para que los estudiantes de computación apliquen sus conocimientos teóricos en un contexto práctico y visual, reforzando su comprensión y habilidades en un área crucial de la computación gráfica y matemática aplicada.

**3.- Proyecto: Simulación y Análisis de Trayectorias sobre Superficies Paramétricas**

**Objetivo del Proyecto**

Este proyecto busca desarrollar y fortalecer las habilidades de los estudiantes de ingeniería en computación en las áreas de visualización 3D, matemáticas aplicadas, y programación gráfica, a través del análisis y la simulación de una trayectoria compleja sobre una superficie esférica parametrizada. Los estudiantes explorarán conceptos de geometría diferencial, cálculo multivariable y computación gráfica.

**Descripción del Problema**

Una hormiga camina sobre la superficie S, definida por la forma paramétrica:

x(u; v) = sin u cos v;

y (u; v) =sin u sin v;

z (u; v) = cos u,

donde las variaciones de u y v son , en t = 0. Se supone que se mueve de tal forma que en cualquier tiempo t > 0 describe la trayectoria

Texto

Descripción generada automáticamente con confianza media

Demuestre que la hormiga se mueve sobre la superficie esférica S y determine cuando y en donde la hormiga abandona el sector descrito de la esfera.

**Tareas del Proyecto**

1. Visualización de la Superficie y la Trayectoria

- Desarrollar una aplicación en Python que utilice `matplotlib` o `mayavi` para visualizar la superficie esférica y la trayectoria de la hormiga.

- Implementar interactividad para visualizar diferentes perspectivas y detalles de la superficie y la trayectoria.

2. Análisis de la Trayectoria

- Demostrar matemáticamente y mediante simulación que la trayectoria de la hormiga permanece sobre la superficie.

- Calcular los puntos específicos donde la trayectoria intercepta los bordes del rango permitido para (u) y (v).

3. Determinación del Abandono de la Trayectoria

- Utilizar técnicas de análisis numérico para determinar exactamente cuándo y dónde la hormiga abandona el sector descrito de la esfera, basándose en los límites para (u) y (v).

- Visualizar estos eventos en la simulación.

4. Informe y Presentación

- Redactar un informe detallado que documente el análisis matemático, el diseño del software, los resultados de la simulación y las conclusiones del estudio.

- Preparar una presentación para mostrar el proyecto, enfocando en los aspectos técnicos y visuales, así como en las aplicaciones de las técnicas aprendidas.

- Traer una maqueta que describa el fenómeno, con los elementos más simples que pueda obtener.

**Competencias Por Desarrollar**

- Programación Gráfica: Implementación de visualizaciones avanzadas en 3D.

- Análisis Matemático y Numérico: Aplicación de cálculo diferencial e integral en problemas complejos.

- Resolución de Problemas: Uso de software y técnicas computacionales para resolver problemas de geometría y física.

- Comunicación Técnica: Habilidad para documentar y presentar proyectos técnicos.

**Aplicaciones Profesionales**

Este proyecto es fundamental para estudiantes interesados en campos como la visualización científica, desarrollo de juegos, realidad virtual y simulaciones físicas, donde las técnicas de modelado y análisis de superficies y trayectorias son esenciales.

**Evaluación**

Los estudiantes serán evaluados por la precisión técnica de su análisis, la calidad de la implementación del software, la creatividad en la visualización, y la claridad en la presentación de sus resultados.

Este proyecto ofrece una excelente oportunidad para que los estudiantes de ingeniería en computación apliquen sus conocimientos teóricos en un contexto práctico y visual, reforzando su comprensión y habilidades en áreas cruciales de la computación aplicada y gráfica.