

## PROGRAMA DE CURSO

Código	Nombre			
CC5502	Geometría Computacional			
Nombre en Inglés				
Computational Geometry				
SCT	Unidades Docentes	Horas de Cátedra	Horas Docencia Auxiliar	Horas de Trabajo Personal
6	10	3	0	7
Requisitos			Carácter del Curso	
CC3001,(CC3101/AUTOR)			Electivo	
Resultados de Aprendizaje				
<p>Este curso tiene por objetivo estudiar y analizar algoritmos geométricos bi- y tridimensionales (2D y 3D) para modelar objetos reales en forma discreta, tanto desde el punto de vista teórico como práctico.</p> <p>Estos algoritmos permiten crear objetos a partir de especificaciones simples o a partir de operaciones geométricas aplicadas sobre objetos existentes, ya sea para para visualizarlos, usarlos en simulaciones, juegos, o en otras aplicaciones en ciencia e ingeniería.</p> <p>Al final del curso el/la estudiante debe ser capaz de:</p> <ul style="list-style-type: none"><li>• conocer las áreas de aplicación de la geometría computacional</li><li>• conocer los problemas clásicos en geometría computacional</li><li>• conocer y aplicar las técnicas/estrategias más usadas para resolver problemas geométricos</li><li>• resolver problemas geométricos teniendo presente casos degenerados y de robustez del algoritmo</li><li>• diseñar e implementar algoritmos teniendo en cuenta casos degenerados y problemas de robustez</li><li>• escoger el algoritmo más apropiado para un problema geométrico dado dependiendo de su entrada</li><li>• estimar la complejidad de un algoritmo geométrico</li><li>• conocer algoritmos geométricos paralelos sobre las GPUs</li><li>• conocer y usar las bibliotecas geométricas más importantes</li></ul>				

Metodología Docente	Evaluación General
<p>El curso consiste en clases de cátedra tradicionales y en clases usando la metodología de aprendizaje basado en problemas. Los/as estudiantes deberán desarrollar tareas de programación de algoritmos geométricos, un proyecto computacional y leer artículos /capítulos de libro en inglés.</p> <p>Este semestre se dicta en modalidad online.</p>	<p>El curso posee tres controles de lectura (cuyo promedio es el control C1), tres evaluaciones grupales (2 o tres personas), cuyo promedio es el control C2, tareas de programación (NT) , proyecto computacional (NPC) y la presentación oral del proyecto computacional (NPO). No hay examen. La nota final (NF) se calcula como sigue:</p> $NC = (0.3 * C1 + 0.3 * C2 + 0.3 * NPC + 0.1 * NPO).$ $NF = 60\% NC + 40\% NT$ <p>NC y NT deben <math>\geq 4.0</math> independientemente.</p>

### Unidades Temáticas

Número	Nombre de la Unidad	Duración en Semanas
1	Conceptos Básicos	3
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<ul style="list-style-type: none"> <li>- Geometría computacional</li> <li>- Definición de polígono</li> <li>- Operaciones geométricas básicas</li> <li>- Propiedades de las triangulaciones</li> <li>- Implementación de un algoritmo de triangulación</li> <li>- Algoritmo para el cálculo del área de un polígono</li> </ul>	<ul style="list-style-type: none"> <li>- Conocer los problemas inherentes a resolver un problema geométrico</li> <li>- Aprender a abordar un problema geométrico</li> <li>- Conocer bibliotecas geométricas open-source</li> </ul>	<p>[Mark98] [Rourke1994] [CGAL 2010]</p>

Número	Nombre de la Unidad	Duración en Semanas
2	Algoritmos de búsqueda e intersecciones	2
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<ul style="list-style-type: none"> <li>• Punto en un polígono</li> <li>• Intersección entre segmentos.</li> <li>• Intersección de polígonos convexos.</li> </ul>	<ul style="list-style-type: none"> <li>• Aprender algoritmos eficientes para resolver problemas comunes</li> <li>• Aprender a detectar y a enfrentar problemas de robustez y casos degenerados</li> <li>• Implementar y validar algoritmos</li> </ul>	<p>[Mark98], [Rourke1994]</p>

Número	Nombre de la Unidad	Duración en Semanas	
3	Cerradura convexa en 2D	2,5	
Contenidos		Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<ul style="list-style-type: none"> <li>Definición de cerradura convexa</li> <li>Algoritmos sencillos, problemas de robustez y casos degenerados</li> <li>Algoritmo Gift-Wrapping.</li> <li>Algoritmo Quick-hull.</li> <li>Algoritmo de Graham.</li> <li>Algoritmo incremental.</li> <li>Dificultades y aplicaciones.</li> </ul>		<ul style="list-style-type: none"> <li>Conocer los algoritmos para resolver uno de los problemas clásicos en geometría computacional</li> <li>Aprender a detectar y enfrentar problemas de robustez y casos degenerados</li> <li>Conocer distintas estrategias para enfrentar un mismo problema</li> <li>Implementar y validar algoritmos</li> </ul>	[Mark98], [Rourke1994]

Número	Nombre de la Unidad	Duración en Semanas	
4	Cerradura Convexa en 3D	1,5	
Contenidos		Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<ul style="list-style-type: none"> <li>Definición de poliedro.</li> <li>Poliedros regulares.</li> <li>Algoritmo Gift-wrapping.</li> <li>Algoritmo incremental.</li> <li>Dificultades y aplicaciones.</li> </ul>		<ul style="list-style-type: none"> <li>Conocer los algoritmos para resolver uno de los problemas clásicos en geometría computacional en 3D</li> </ul>	[Mark98], [Rourke1994]

Número	Nombre de la Unidad	Duración en Semanas
5	Diagrama de Voronoi y particiones de Delaunay en 2D y 3D	2
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<ul style="list-style-type: none"> <li>- Definiciones y propiedades básicas. Algoritmos de construcción:</li> <li>- Algoritmo Incremental</li> <li>- Extensiones de los diagramas de Voronoi</li> <li>- Árbol de cobertura mínima</li> <li>- Aplicaciones.</li> <li>- Otras estructuras espacial</li> </ul>	<ul style="list-style-type: none"> <li>- Conocer un problema de partición del espacio clásico y algoritmos eficientes para resolverlos</li> <li>- Conocer distintas estrategias para enfrentar un mismo problema</li> </ul>	[Okabe2000] [Mark98], [Rourke1994]

Número	Nombre de la Unidad	Duración en Semanas
6	Generación de mallas en 2D y 3D	2
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<ul style="list-style-type: none"> <li>- Especificaciones de la geometría de un objeto.</li> <li>- Algoritmos para triangulaciones de Delaunay y Delaunay restringida</li> <li>- Algoritmos de mejoramiento de una malla y parámetros de calidad.</li> <li>- Algoritmos basados en quadrees</li> <li>- Algoritmos para generación de mallas de tetraedros y mixtas basados en octrees</li> <li>- Aplicaciones.</li> </ul>	<ul style="list-style-type: none"> <li>- Aprender algoritmos clásicos para la generación de mallas en 2D y 3D</li> <li>- Conocer herramientas open- source existentes</li> <li>- Conocer algoritmos paralelos</li> </ul>	[Mark98], [Rourke1994][ACMtog] [Visual] [EngWithComp] [IntConf]

Número	Nombre de la Unidad	Duración en Semanas
7	Partición binaria del espacio	1
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<ul style="list-style-type: none"> <li>- Definición de binary partition trees (BSP)</li> </ul>	<ul style="list-style-type: none"> <li>- Aprender estructuras</li> </ul>	[Mark98]

<ul style="list-style-type: none"> <li>- BPS y algoritmo del pintor</li> <li>- Construyendo el BSP</li> <li>- Aplicaciones</li> </ul>	espaciales útiles para diversas aplicaciones gráficas	
---	---	--

Número	Nombre de la Unidad	Duración en Semanas
6	Planificación de movimientos	1
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<ul style="list-style-type: none"> <li>- Detección de colisiones.</li> <li>- Algoritmo para obtener el camino más corto.</li> <li>- Búsqueda de caminos en regiones con obstáculos.</li> </ul>	<ul style="list-style-type: none"> <li>- Aprender a resolver aplicaciones en geometría computacional</li> </ul>	[Mark98], [Rourke1994]

Bibliografía
<p>[Mark1998] Mark de Berg, Marc van Kreveld, Mark Overmars, Otfried Schwarzkopf , Computational Geometry: Algorithms and applications . Springer (segunda edición) . 1998</p> <p>[Rourke1994] Joseph O'Rourke, Computational Geometry in C. Cambridge University Press, 1994 (second edition 1998).</p> <p>[Okabe2000] Atsuyuki Okabe, Barry Boots, Kokichi Sugihara, Sung Nok Chiu, Spatial tessellations. Concepts and applications of oronoi diagrams. Wiley Series in Probability and Statistics. 2000.</p> <p>[ACMtog] ACM Transactions on Graphics</p> <p>[Visual] Visual computer Journal</p> <p>[EngWithComp] Engineering with Computers</p> <p>[IntConf] International conferences related to the area</p>

Vigencia desde:	2021
Elaborado por:	Nancy Hitschfeld Kahler

**Actividades del curso y**  
**Calendario de Evaluaciones**  
**Primavera 2021**

## **1. Proyectos:**

Temas van desde implementar un algoritmo para resolver un problema geometrico hasta investigar sobre un tema o uso de una libreria geometrica que les interese aprender. Los algoritmos o modelos geoemtricos pueden parte de un juego, para analizar datos o visualizar informacion geometrica, entre otros. Las tecnologias a usar pueden ser:

- CGAL (Computational geometry library) (<http://www.cgal.org/>)
  - Convex-hulls and polygons
  - Cell Complexes and polyhedra
  - Triangulations and Delaunay Triangulations
- Otra librería geométrica
- Cuda u opengl
- OpenGL y Shaders (GLSL)
- WebGL 2 o Three.js
- Unity, Unreal u otro motor de juegos

**Presentacion del tema:** Descripción de media página (6ta semana de clases)

- Tema de presentación
- Tema de Proyecto a desarrollar/investigar (puede cambiar)
- Puede ser individual o en grupos de dos.

### **3. Fechas Controles de Lectura**

- Control de Lectura 1: miércoles 1 septiembre
- Control de Lectura 2: miércoles 6 octubre
- Control de Lectura 3: miércoles 10 noviembre

### **4. Fechas evaluaciones Ejercicios: (en grupo 2 o tres)**

- Evaluación 1: miércoles 29 septiembre
- Evaluacion 2: miércoles 13 octubre
- Evaluacion 3: miércoles 17 noviembre

### **6. Tareas de programación**

- Algoritmos geométricos clásicos
- Programacion orientada a objetos en c++