Identification	**Subject**	CMS 350 Computer Graphics, 3KU /6ECTS credits
	Department	Computer Science
	Program	Undergraduate
	Term	Fall, 2023
	Instructor	Javad Mehri-Tekmeh (PhD)
	E-mail:	jmehri@khazar.org
	Phone:	(+994 12) 421 1093 (ext. 266)
	Classroom/hours	41 Mehseti str. (Neftchilar campus), Sat. 15:20-18:30 (group A), Sat. 18:40-21:00 (group B)
Prerequisites	CMS 215 Data Structure, MATH 235 Linear Algebra	
Language	English	
Compulsory/Elective	Required	
Required textbooks and course materials	***Core Textbook:*** 1. Ed Angel & Dave Shreiner, Interactive Computer Graphics, 8th Edition, Pearson Education, 2020. 2. https://codelabs.developers.google.com/your-first-webgpu-app#0 3. https://www.khronos.org/assets/uploads/developers/presentations/Intro-to-WebGPU_May21.pdf	
Course outline	This course provides a comprehensive introduction to computer graphics, with a focus on hands-on experience in creating interactive and visually engaging 2D and 3D graphics applications. Students will explore fundamental concepts and practical implementation using WebGL2, transition to the modern WebGPU technology, and build an understanding of key graphics principles. Through a series of lectures, coding exercises, and projects, participants will gain the skills necessary to create immersive graphics and interactive visualizations.	
Course objectives	By the end of this course, students should be able to: • Develop a fundamental understanding of computer graphics, including the graphics pipeline, key rendering concepts, and terminology. • Acquire proficiency in WebGL2, its architecture, and associated programming paradigms to create interactive 2D and 3D graphics applications. • Learn to write and manipulate vertex and fragment shaders to control rendering, lighting, and shading in WebGL2 and WebGPU. • Create 2D and 3D scenes using WebGL2, employing techniques such as drawing geometric shapes, applying transformations, and setting up camera views and projections. • Understand lighting models and implement shading techniques to create realistic lighting effects in computer graphics applications. • Apply texture mapping to enhance the visual appeal of objects and scenes in WebGL2. • Explore the advantages of WebGPU and gain the ability to port and develop computer graphics projects using WebGPU. • Apply the knowledge gained throughout the course to create a significant, interactive computer graphics project showcasing a wide range of techniques and concepts.	
Learning outcomes	By the end of this course, students will be able to: • **Understand Computer Graphics Concepts:** Define and explain fundamental concepts in computer graphics, including the graphics pipeline, rendering techniques, and key terminologies. • **Master WebGL2:** Develop the proficiency to create 2D and 3D graphics applications using WebGL2, understanding its architecture, and programming paradigms.	

<table>
<tr><td></td><td colspan="3">

Shader Programming Proficiency: Write, modify, and apply vertex and fragment shaders to control rendering, lighting, and shading effects in WebGL2 and WebGPU applications.
Create Interactive Scenes: Design and construct interactive 2D and 3D scenes by implementing techniques like drawing geometric shapes, applying transformations, and setting up camera views and projections.
Implement Lighting and Shading: Apply lighting models and shading techniques to achieve realistic lighting and shading effects in computer graphics applications.
Utilize Texture Mapping: Incorporate texture mapping to enhance the visual quality and realism of objects and scenes in WebGL2 projects.
Transition to WebGPU: Understand the advantages of WebGPU and effectively port and develop computer graphics applications using this newer technology.
Final Project Development: Create a complex and interactive computer graphics project that demonstrates proficiency in applying learned principles and techniques throughout the course.

</td></tr>
<tr><td rowspan="3">Teaching methods</td><td colspan="2">Lecture</td><td>X</td></tr>
<tr><td colspan="2">Group discussion</td><td>X</td></tr>
<tr><td colspan="2">Experiential exercise</td><td>X</td></tr>
<tr><td rowspan="5">Evaluation</td><td>Methods</td><td>Date/deadlines</td><td>Percentage (%)</td></tr>
<tr><td>Midterm Exam</td><td></td><td>25</td></tr>
<tr><td>Project</td><td></td><td>35</td></tr>
<tr><td>Final Exam</td><td></td><td>40</td></tr>
<tr><td>Total</td><td></td><td>100</td></tr>
<tr><td>Policy</td><td colspan="3">
Project Title: "Interactive 3D Scene Visualization and Simulation"
Project Description:
For this project, students will have the opportunity to apply their knowledge of computer graphics, WebGL2, and WebGPU to create an interactive 3D scene that goes beyond mere visualization. The goal is to develop a virtual environment or scenario where users can not only explore but also interact with objects and elements within the scene.
Project Components:
1. 3D Scene Setup: (5 points) Create a 3D environment using WebGL2 or WebGPU, including a variety of objects, such as buildings, terrain, vehicles, or characters. Implement a camera system for navigation within the scene.
2. Interactivity: (5 points) Enable user interaction with the 3D objects or elements. This can include picking up and moving objects, triggering animations, or adjusting parameters within the scene.
3. Realistic Lighting and Shading: (5 points) Implement advanced lighting and shading techniques to make the scene as visually realistic as possible, considering factors like shadows, reflections, and material properties.
4. Texture Mapping: (5 points) Apply textures to objects within the scene to enhance realism and detail.
5. User Interface: (5 points) Create a user interface that allows users to control and interact with the scene. This can include buttons, sliders, or input fields to adjust parameters or initiate actions.
6. Documentation and Presentation: (10 points) Provide clear documentation explaining the project, including how to interact with the 3D scene. Prepare a presentation to demonstrate and explain the project to the class.
Project Goals:
▪ Develop a comprehensive and interactive 3D scene using WebGL2 or WebGPU.
▪ Apply advanced graphics techniques, including realistic lighting, shading, and texture mapping.
▪ Incorporate interactivity and user-controlled elements within the scene.
</td></tr>
</table>

- Demonstrate knowledge of both WebGL2 and WebGPU, showcasing the differences and advantages of the latter.
- Present and document the project effectively.

Final Presentation:

At the end of the project, each student or team should present their game to the class. They should explain their design choices, demonstrate gameplay, and discuss the challenges they faced during development.

- **Project**
  This course is not about programming. For this reason, the result of the project is the most important part of it, and **the projects that are not executed on the computer are not given a grade.** The number of people in each group in the project can be maximum 4 people. All team members are responsible for answering any questions about the project.
- **Preparation for class**
  The structure of this course emphasizes the importance of independent study and preparation outside of class. The lecture material will concentrate on the key points raised in the text. Reading the assigned chapters and becoming acquainted with them prior to class will aid your understanding of the lecture. Following the lecture, you should review your notes and work on relevant problems and cases from the chapter's end, as well as sample exam questions.
  We will also have many review sessions throughout the semester. These review sessions will take place during the regular class times.
- **Withdrawal (pass/fail)**
  This course strictly adheres to the grading policy of the School of Engineering and Applied Science. As a result, a student is normally expected to pass with a grade of at least 60%. In the event of failure, he or she will be required to repeat the course the following term or year.
- **Cheating/plagiarism**
  Cheating or other plagiarism during the Quizzes, Mid-term and Final Examinations will lead to paper cancellation. In this case, the student will receive a zero (0) without further consideration.
- **Professional behavior guidelines**
  During class, students must act in a way that fosters a positive academic and professional environment. Unauthorized conversations and unethical behavior are forbidden.
- **Ethics**
  Students should not arrive in late to class.
  All cell phones must be turned off and stowed away before entering class.
  Use of any electronic devices is not allowed in the classroom and violators will. be punished accordingly.

Tentative Schedule

Week	Date/Day (tentative)	Topics	Textbook
1	14-Oct.	**Introduction to Computer Graphics** • Definition and importance of computer graphics. • The graphics pipeline. • Overview of the course structure and requirements **Begin exploring WebGL basics**	**Ch. 1**

		• Introduction to WebGL. • Setting up a simple WebGL application.	
2	21-Oct	**Introduction to WebGL** • Models and Architecture • WebGL Background: History of Graphics Software • WebGL Basics • A Complete Program	**Ch. 2**
3	Make-up	**GLSL and Shaders** • Shaders • Colors and Attributes • Reading and Initializing Shaders • Three Dimensions and Hidden Surface Removal	**Ch. 2**
4	28-Oct	**Input and Interaction** • Animation • Callbacks and Event Listeners • Position Input • Picking • Geometry	**Ch. 3**
5	Make-up	**Geometry and Transformations** • Representation • Homogeneous Coordinates • Transformations • Transformations in WebGL • Applying Transformations • Quaternions	**Ch. 4**
6	04-Nov	**Modeling and Viewing** • Modeling • Rotating Cube • The Virtual Trackball • Classical Viewing • Positioning the Camera • Projection Functions	**Ch. 5**
7	Make-up	**Projection Matrices and Shadows** • Orthogonal Projection Matrices • Perspective Projection Matrices • Meshes • Shadows • Lighting and Shading 1	**Ch. 5, 6**
8	11-Nov	**Review for Midterm exam & Exercise.**	
9	18-Nov	**Midterm Exam.**	
10	25-Nov	**Lighting and Shading** • Lighting and Shading 2 • Lighting in WebGL • Polygonal Shading • Per Vertex and Per Fragment Shading • Marching Squares	**Ch. 6**
11	02-Dec	**Buffers and Texture Mapping** • Buffers • BitBlt	**Ch. 7**

		• Texture Mapping • WebGL Texture Mapping	
12	09-Dec	**Discrete Techniques** • Reflection and Environment Maps • Bump Maps • Compositing and Blending • Imaging Applications • Rendering the Mandelbrot Set	**Ch. 8**
13	16-Dec	**Your first WebGPU app**	**[2], [3]**
14	23-Dec	**Fractals**	**Ch. 10**
15	30-Dec	**Project presentation**	
	TBA	**Final Exam**	

This syllabus is a guide for the course and any modifications to it will be announced in advance.