Course Syllabus

Code - Course Title	**OGP122 - Generative Coding and Algorithmic Design**
Course Type	II. SEMESTER
Laboratory + Practice	1+2
ECTS	**3.0**
Language of Instruction	Türkçe
Course Type	Area Elective Courses
Course Instructor(s)	ÖĞR. GÖR. GÜVENÇ USANMAZ

Mode of Delivery	This course is delivered through face-to-face instruction. Theoretical lectures and laboratory sessions are integrated throughout the semester. The default tool for practical work is p5.js; the instructor may prefer an equivalent environment (e.g., Processing, openFrameworks, TouchDesigner).
Prerequisites	There are no prerequisites or co-requisites for this course.
Courses Recomended	It is recommended to have completed OGP103 Programming I and XXX7968 Fundamentals of Digital Art and Design before taking this course. Taking it concurrently with OGP104 Programming II is also suitable. Basic programming logic and visual design knowledge will facilitate the learning process.
Recommended Resources	p5.js. Official reference: https://p5js.org/reference/Shiffman, D. The Nature of Code: https://natureofcode.com/ (selected sections)
Work Placement	This course has no internship or placement requirement.

Course Syllabus

Weeks	Learning Outcomes	Topics	Teaching Methods
Week - 1		Introduction to generative coding; algorithmic aesthetics, visual art with code; p5.js development environment.	
Week - 2		Draw loop, coordinate system, color and form basics.	
Week - 3		Generative geometry: form construction with points and lines, repetition and symmetry.	
Week - 4		Grid-based designs; tile, pattern and parametric variation.	
Week - 5		Randomness and noise: Perlin noise, fBm (fractal brownian motion).	
Week - 6		Abstract map/texture generation with noise; game background and terrain concept.	
Week - 7		Pixel and image processing: image loading, filters, color manipulation.	
Week - 8		Pixel sorting, glitch art and error aesthetics.	
Week - 9		Complex systems: cellular automata, Conway's Game of Life.	
Week - 10		Introduction to shader programming: fragment shader, GLSL basics.	

Course Syllabus

Weeks	Learning Outcomes	Topics	Teaching Methods
Week - 11		Mathematical visualization: functions, waves and fractal forms.	
Week - 12		Generative architecture: rule-based (L-system, shape grammar) structure generation.	
Week - 13		Daily coding practices (daily sketch); iteration and portfolio selection.	
Week - 14		End-of-term generative portfolio presentation; process documentation and game context.	

Learning Outcomes

#	Learning Outcome
1	Explains algorithmic aesthetics and generative coding concepts in a game/art context.
2	Produces parametric geometry and grid-based patterns with p5.js.
3	Generates procedural texture/map using Perlin noise and fBm.
4	Applies pixel-level image processing and glitch art techniques.
5	Designs cellular automata and Game of Life simulation.
6	Creates mathematical visualization with basic fragment shaders.
7	Applies rule-based generative architecture/structure algorithm.
8	Prepares and presents a generative portfolio through daily sketch practice.

Sustainable Development Goals

No content has been provided.

Course Theme

No content has been provided.

AKTS Workload

Activity	Duration (Hour)	Count	Workload
Teorik Derse Katılım	0	0	0
Sınıf Dışı Çalışma (Ön Hazırlık & Pekiştirme Amaçlı)	0	0	0
Ödevler	0	0	0
Proje - Rapor - Sunum Hazırlama	0	0	0
Ara Sınav Hazırlığı (sınav süresi dahil)	0	0	0
Final Sınavı Hazırlığı (sınav süresi dahil)	0	0	0
Uygulama (Laboratuvar, Müzik/Enstrüman Uygulaması, Atölye/Saha Çalışması vb.)	0	0	0
Staj Uygulaması	0	0	0
		Calculated ECTS	**0**

Assessment Method and Passing Requirements

No content has been provided.