STUDY MODULE DESCRIPTION FORM

Name of the module/subject **Usage of Direct3D in Computer Graphics and Visualization**			Code **1010511261010510023**
Field of study **Bioinformatics**	Profile of study (general academic, practical) **(brak)**		Year /Semester **3 / 6**
Elective path/specialty **-**	Subject offered in: **Polish**		Course (compulsory, elective) **elective**
Cycle of study: **First-cycle studies**	Form of study (full-time,part-time) **full-time**		
No. of hours Lecture: **30** Classes: **-** Laboratory: **30** Project/seminars: **-**			No. of credits **4**
Status of the course in the study program (Basic, major, other) **(brak)**	(university-wide, from another field) **(brak)**		
Education areas and fields of science and art **technical sciences** **Technical sciences**			ECTS distribution (number and %) **4 100%** **4 100%**

Responsible for subject / lecturer:

dr inż. Witold Andrzejewski
email: witold.andrzejewski@cs.put.poznan.pl
tel. (0-61) 665-2965
Wydział Informatyki
60-965 Poznań, ul. Piotrowo 2

Prerequisites in terms of knowledge, skills and social competencies:

1	**Knowledge**	Student starting this module should have basic knowledge regarding programming languages, geometry and computer system architectures.
2	**Skills**	The student should have the skills to solve simple algorithmic problems, program in C/C++ and to acquire knowledge from the designated sources of information.
3	**Social competencies**	The student should also understand the need to extend his/her competences and be ready to work in a team. Moreover, the student should show such attitudes as honesty, responsibility, perseverance, curiosity, creativity, manners, and respect for other people.

Assumptions and objectives of the course:

1. Teach students the basic concepts and definitions related to computer graphics.
2. Teach students the mathematical basics of 3D graphics.
3. Teach students the methods of 3D object animation.
4. Teach students the shading models and hidden surface removal methods.
5. Teach students different ways of 3d model representations.
6. Teach students the basics of raster rendering algorithms and halftone approximation algorithms
6. Teach students the basics of polygon clipping on a plane
6. Teach students the basics of data visualization methods.
7. Develop the students computer graphics application programming skills by introducing popular computer graphics libraries

Study outcomes and reference to the educational results for a field of study

Knowledge:

Skills:

Social competencies:

Assessment methods of study outcomes

Formative assessment:

a) lectures:

- based on answers to questions related to subjects covered during previous lectures,

b) laboratory classes:

- evaluation of correctness of implementation of assigned tasks (following provided lab. instructions),

Total assessment:

a) verification of assumed learning objectives related to lectures:

- evaluation of acquired knowledge on the basis of the written exam (a test, 50 questions, total points achievable 50, 25 points needed to pass).

b) verification of assumed learning objectives related to laboratory classes:

- based on the project implemented by a team of students, each students? grade is evaluated based on the quality of his/hers part as well as answering to several project related questions.

Additional elements cover:

- discussing more general and related aspects of the class topic,
- ability to utilize knowledge covered in previous lectures
- showing how to improve the instructions and teaching materials.
- pointing out flaws in teaching materials and helping lecturer to improve them

Course description

The lectures cover the following topics:

1. Computer graphics basics. Differences between computer graphics and data visualization. Image buffering methods, hidden surface removal, stencil buffers, basic texturing algorithms, texure types, some special effects.

2. Basics of linear algebra and geometry. Homogenous coordinate system. Matrix and quaternion based representation of geometrical transofmrations. Gimbal lock problem. Mathematical basics of camera in a 3D scene. Perspective and orthogonal projection. Geometrical transformations of normal vectors.

3. Animation techniques of 3D models including: per vertex animation, skeletal animation and inverse kinematics.

4. Methods of modelling of light transport in the scene:

* Basic radiometric terms
* Matemathic models of light sources
* Bi-directional reflectance distribution function and its properties. Light transport equation.
* Bi-directional Shading Function by Schlick as a simplification of the BRDF function.
* Diffused light models (Lambertian Model and Minnaert Model), reflected light models (Phong, Phong-Blinn).
* Physical Based Shading Models (Cook-Torrance)
* Monte Carlo rendering

5. Basics of raytracing algorithms. Computation of primary ray, shadow ray, reflected ray and refracted ray. Finding intersections of ray with a plane, sphere, axis aligned bounding box (AABB) and a triangle. Space partition methods allowing for faster searching for intersections within the scene. Whitted's algorithm and generalizations. Path-tracing and Photon Mapping algorithms.

6. Aliasing problem and Anti-aliasing methods.

7. Polygon clipping algorithms: Cohen-Sutherland, Cyrus-Beck, Sutherland-Hodgeman and Greiner-Hodgeman. Polygon clipping in homogenous space.

8. 3D model representation methods. Multiple different polygon mesh representations. Voxel visualization methodsL Raycasting, Texture-Bawed volume rendering, marching squares, marching cubes and marching tetrahedra. Curve and surface modelling including: quadrics, Hermite curves and Bezier surfaces. Particle systems.

9. Raster-based rendering algorithms:

* polygon rendering. Bresenham's line and circle algorithms, polygon filling methods.
* halftone approximation: treshold method, Floyd-Steinberg method, cell-based method.
* scanline based triangle rendering with data interpolation, with/out perspective correction
* image filtering algorithms

10 Data visualization. Problem definition and applications. Visualization process. Data sources and their structures used in visualization. Classification of data types. Effectiveness of data visualization methods. Data to graphical primitive mapping methods. Chart types. Data dimensionality reduction.

The laboratory lessons cover the following topics:

1. Introduction to Direct3D API. Description of the basic program structure. Introduction of basic subjects related to drawing and animation of 3D models.

2. Exercises related to positioning and animating of 3D objects. Students gain the skills necessary to properly construct geometrical transformation matrices.

3. Introduction to HLSL language. Writing of simple shaders. Per-vertex shading based on Lambertian and Phong shading models.

4. Advanced shading programs in HLSL. Per-pixel shading using Phong model. Cel-shading. Edge detection in eye space.

5. Texturing of 3D objects in Direct3D. Different texture sampling methods including bi- and tri-linear filtering as well as MIP-mapping

6. Multitexturing. Simple Environment mapping.

7. Instancing and fur effect in GLSL. Drawing grass.

8. Normal mapping and Parallax mapping.

9. Introduction to geometry shaders.

10. Shader-based per vertex animation.

Learning methods:

1. Lectures: multimedia presentation, presentation illustrated with examples presented on black board.
2. Laboratory classes: : solving tasks, practical exercises and experiments, discussion.

Basic bibliography:

1. G. Banaszak, W. Gajda: ?Elementy algebry liniowej? część I i II, WNT, Warszawa, 2002
2. B. Kaczmarek: ?Elementy algebry i analizy macierzy?, Wydawnictwo PP, 1689, Poznań, 1992
3. J.D. Foley, A. van Dam, S.K. Feiner, J.F. Hughes, R.L. Phillips, Wprowadzenie do grafiki komputerowej, WNT
4. M. Jankowski, Elementy grafiki komputerowej, WNT

Additional bibliography:

1. A.N. DcGorban, , B. Kégl, D.C. Wunsch, A. Zinovyev, (Eds.) Principal Manifolds for Data Visualization and Dimen-sion Reduction
2. F.H. Post, G.M. Nielson, G.-P. Bonneau, Data Visualization: The State of the Art, Proceedings of the 4th Dagstuhl Seminar on Scientific Visualization
3. C.D. Hansen, C.R. Johnson (eds.): The Visualization Handbook, Elsevier, 2005
4. F. D. Luna: ?Introduction to 3D Game Programming with DirectX 11?

Result of average student's workload

Activity	Time (working hours)

1. participating in lectures 15x2 hours 2. preparing to tests 3. participating in laboratory classes / tutorials: 15 x 2 hours, 4. preparing to laboratory classes 5. implementing a program / programs, running and verification (beyond the time of the laboratories) 6. consulting issues related to the subject of the course; especially related to laboratory classes and projects 7. reading teaching materials and literature 8. preparing to final test and participation in final test (15 hours+ 2 hours) 9. final test results discussion	30 5 30 8 20 2 3 17 1

Student’s workload

Source of workload	hours	ECTS
Total workload	116	4
Contact hours	65	2
Practical activities	58	2