



MODULE DESCRIPTION

Module code	
Module name	Architektury procesorów graficznych
Module name in English	Architectures of Graphical Processing Units
Valid from academic year	2012/13

MODULE PLACEMENT IN THE SYLLABUS

Subject	Computer Science
Level of education	1st degree <i>(1st degree / 2nd degree)</i>
Studies profile	General <i>(general / practical)</i>
Form and method of conducting classes	Full-time <i>(full-time / part-time)</i>
Specialisation	Computer Graphics
Unit conducting the module	The Department of Computer Science
Module co-ordinator	Roman Stanisław Deniziak, PhD hab., Eng., Professor of the University
Approved by:	

MODULE OVERVIEW

Type of subject/group of subjects	Major <i>(basic / major / specialist subject / conjoint / other HES)</i>
Module status	Compulsory <i>(compulsory / non-compulsory)</i>
Language of conducting classes	Polish
Module placement in the syllabus - semester	6th semester
Subject realisation in the academic year	Summer semester <i>(winter / summer)</i>
Initial requirements	The Fundamentals of Computer Graphics <i>(module codes / module names)</i>
Examination	Yes <i>(yes / no)</i>
Number of ECTS credit points	5

Method of conducting classes	Lecture	Classes	Laboratory	Project	Other
Per semester	30			15	



TEACHING RESULTS AND THE METHODS OF ASSESSING TEACHING RESULTS

Module target	The aim of the module is to familiarise students with the structure of graphics processing units (GPUs), co-operation methods of a processor with a graphical processor, and the principles of generating a picture in 3D graphics (in a GPU). Another aim is to acquire programming skills as regards graphical processors in terms of implementing shaders in the Cg language.
----------------------	--

Effect symbol	Teaching results	Teaching methods (l/c/l/p/other)	Reference to subject effects	Reference to effects of a field of study
W_01	Knowledge of the structure and the principle of operation of GPUs.	l	K_W08	T1A_W03 T1A_W04
W_02	Knowledge of techniques of obtaining realistic effects in 3D graphics using shaders.	l	K_W12	T1A_W04 T1A_W07
W_03	Basic knowledge of stream and multi-threaded processing in GPUs.	l	K_W08 K_W11	T1A_W04 T1A_W07
W_04	Knowledge of the directions of development concerning the architectures of GPUs.	l	K_W18	T1A_W05
U_01	The ability to design and implement shaders in the Cg language.	p	K_U02 K_U18	T1A_U02 T1A_U07 T1A_U16

Teaching contents:

Teaching contents as regards lectures

Lecture number	Teaching contents	Reference to teaching results for a module
1	The structure and principles of operation concerning a graphical controller.	W_01
2	Hardware acceleration methods of graphical operations.	W_01
3	Data processing in graphical processors.	W_01
4	Programming graphical processors, shaders.	
5	Cg language: functions, expressions, and data types.	W_02
6	Cg language: transformations, lighting modelling, and controlling constructions.	W_02
7	Cg language: animation and environment mapping.	W_02
8	Cg language: bump mapping and special effects.	W_02
9	Cg language: optimisation methods of shaders.	W_02
10	Dedicated structures of GPUs.	W_01
11	Unified architectures of GPUs.	W_01
12	Stream processing. CUDA.	W_03
13	CrossFire and SLI technologies.	W_01
14	AGP and PCI Express buses of graphical processors.	W_01
15	Development directions of GPU architectures.	W_04

The characteristics of project assignment



The subject matter covers designing and implementing shaders in the Cg language realising the assigned graphical effects (lighting modelling, animation, modelling mirror reflections, transparency, generating inhomogeneous surfaces, etc.), integrating shaders with programs in the OpenGL or Direct3D language, starting and testing programs in computers equipped with GPUs.

The methods of assessing teaching results

Effect symbol	Methods of assessing teaching results (assessment method, including skills – reference to a particular project, laboratory assignments, etc.)
W_01	An examination
W_02	An examination
W_03	An examination
W_04	An examination
U_01	Obtaining a credit for the project based on a report and a program source code.

STUDENT'S INPUT

ECTS credit points		
	Type of student's activity	Student's workload
1	Participation in lectures	30
2	Participation in classes	
3	Participation in laboratories	
4	Participation in tutorials (2-3 times per semester)	3
5	Participation in project classes	15
6	Project tutorials	15
7	Participation in an examination	2
8		
9	Number of hours requiring a lecturer's assistance	65 (sum)
10	Number of ECTS credit points which are allocated for assisted work (1 ECTS credit point=25-30 hours)	3
11	Unassisted study of lecture subjects	15
12	Unassisted preparation for classes	
13	Unassisted preparation for tests	
14	Unassisted preparation for laboratories	
15	Preparing reports	
16	Preparing for a final laboratory test	
17	Preparing a project or documentation	15
18	Preparing for an examination	30
19	Preparing questionnaires	
20	Number of hours of a student's unassisted work	60 (sum)
21	Number of ECTS credit points which a student receives for unassisted work (1 ECTS credit point=25-30 hours)	2
22	Total number of hours of a student's work	125
23	ECTS credit points per module	5



Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego

	<i>1 ECTS credit point=25-30 hours</i>	
24	Work input connected with practical classes <i>Total number of hours connected with practical classes</i>	45
25	Number of ECTS credit points which a student receives for practical classes <i>(1 ECTS credit point=25-30 hours)</i>	2