STUDY MODULE DESCRIPTION FORM

Name of the module/subject **Concurrent programming**		Code **1010334451010335200**

Field of study **Information Engineering**	Profile of study (general academic, practical) **(brak)**	Year /Semester **3 / 5**

Elective path/specialty **-**	Subject offered in: **polish**	Course (compulsory, elective) **elective**

Cycle of study: **First-cycle studies**	Form of study (full-time,part-time) **part-time**

No. of hours				No. of credits **4**
Lecture: **16**	Classes: **-**	Laboratory: **12**	Project/seminars: **-**	

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**	ECTS distribution (number and %**)** **4 100%**

Responsible for subject / lecturer:

dr inż. Andrzej Sikorski
email: andrzej.sikorski@put.poznan.pl
tel. +48(61)6653730
Electrical Engineering
ul. Piotrowo 3A 60-965 Poznań

Prerequisites in terms of knowledge, skills and social competencies:

1	**Knowledge**	Basic knowledge of numeric algorithms and combinatorics Basics of calculus, algebra and set theory.
2	**Skills**	Proficiency in any OOP language. Win32 or .NET programming.
3	**Social competencies**	students are expected to be quiet during the lecture

Assumptions and objectives of the course:

Proficiency in concurrent programming. Designing of synchronization schemes. Ability to efficiently and correctly identify possible application of available standard design paterns.

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

Knowledge:

Skills:

Social competencies:

Assessment methods of study outcomes

examination, laboratory reports, projects.

Course description

Sequential optimization. Superscalar optimization. Machine code level optimizations recommended by AMD for FPU operations. Declarative concurrency supported by OpenMP.

Operating System level facilities for concurrent/parallel programming. Constructs and techniques availble on win32, .net,java. Intel TBB dynamic parallelism optimally exploiting the harware and computational state (cache buffering, data availability and internal redundance of processing components)

Basic bibliography:

1. M.Herlihy N.Shavit : The Art of Multiprocessor Programming Publisher: Morgan Kaufmann; 1 edition (March 14, 2008)

2. M. Ben-Ari Principles of Concurrent and Distributed Programming Addison-Wesley; 2 edition (March 6, 2006)

Additional bibliography:

Result of average student's workload	
Activity	**Time (working hours)**

Student's workload		
Source of workload	**hours**	**ECTS**
Total workload	100	4
Contact hours	50	2
Practical activities	50	2