COURSE DESCRIPTION CARD - SYLLABUS

Course name
Performance evaluation of Computer Systems

Course

Field of study
Computing
Area of study (specialization)
Advanced Internet Technologies
Level of study
Second-cycle studies
Form of study
part-time

Year/Semester
1/1
Profile of study
general academic
Course offered in
Polish
Requirements
compulsory

Number of hours

Lecture
16
Tutorials

Laboratory classes
16
Projects/seminars

Other (e.g. online)

Number of credit points

4

Lecturers

Responsible for the course/lecturer:
dr inż. Rafał Klaus
email: Rafał Klaus@put.poznan.pl
tel: 48 61 6652574
wydział: Faculty of Computing and Telecommunications
adres: ul. Piotrowo 2, 60-965 Poznań

Responsible for the course/lecturer:

Prerequisites

Course objective

Introducing basic knowledge on methodology of computer system performance evaluation in typical applications of electronic commerce, and web applications
Developing ability of discovering, analyzing and solving computer performance problems.

Course-related learning outcomes

Knowledge

have general understanding of algorithms, complexity, computer system architecture, operating systems, network technologies, programming languages and paradigms, databases

knows the life cycle of hardware and software systems: Genesis and aging of benchmarks, approaches to the scalability of benchmarks
know the trends and main new developments in computer science and IT, as well as in related fields of science and technology
have extended understanding of selected issues of computer science, such as performance metrics, infrastructure for performance evaluation tests, methods of performance evaluation used in the past
know basic methods, techniques and tools applicable in solving complex engineering tasks in the field of performance evaluation, detecting and removing performance problems

Skills

design and conduct experiments, including performance measurements, analyze the results, and draw conclusions from the results; exploit simple research and analysis methodologies to formulate and solve engineering and research tasks
integrate knowledge from various branches of computer science, while formulating and solving engineering tasks
designate directions of the further study, conduct a successful study
acquire knowledge from literature, databases and other sources (in Polish and English), integrate and critically interpret it, on basis of that draw conclusions, formulate justified opinions formulate and test hypotheses for engineering and simple research problems related to the performance of computer systems
asses utility of methods and tools applied to solve an engineering task consisting in constructing and evaluation of an information system or its components, as well as recognize limitations of such methods and tools.
propose improvements in the existing technical solutions to increase performance of computer systems
assess practicality and applicability of new advances in computer science and IT to solve computer performance problems

Social competences

realize that in computer science and IT knowledge and skills outdate quickly
know examples of, and understand reasons why, faulty performance of computer systems caused financial and social damages or even severe health or life loss

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment:
a) lectures:
• answers to short questions related to the content of the lecture,
• 2-3 short tests related to the subjects of the lecture,
b) labs:
• on the basis of progresses in doing the current assignments
Total assessment:
a) lectures:
• evaluation of the knowledge and skills by a written exam. The exam has 5 to 8 questions related to theoretical issues and quantitative performance estimation methods presented in the lectures. To pass the exam at least 50% of possible points must be scored.
• discussion of exam results
b) labs:
• assessment of readiness and skills necessary to conduct the lab exercises,
• continuous assessment on all labs by questions and answers, rewarding progresses of skills in applying the taught methods
• assessment of the reports prepared partially during the labs and finished afterwards; the assessment includes contribution to the team work,
Additional elements cover:
• accurate comments to and explanations of the considered issues of computer performance evaluation
• essays on the current state and trends in computer performance evaluation,
• accurate indication of information technologies performance limitations and ways of bypassing such limitations,
• ability to cooperate in a team to solve the assignment from the lab exercises
• indicating possible improvements in the teaching process and materials

Programme content

The lecture covers the following topics: Place and goals of performance evaluation studies. Relationships between the components in the infrastructure of performance evaluation. Three classic techniques of performance evaluation: analytic methods, simulation, experiments, criteria of choosing evaluation techniques. Methodologies of performance evaluation. Classification of performance metrics. Examples of typical performance metrics related to speed, resources, reliability, cost, fairness. Tools and techniques of performance evaluation: types of workloads, selection, specification, generation of workloads. Performance monitors. Past and the current benchmarks to illustrate methods of performance evaluation methodologies, aging of benchmarks, scaling of benchmarks to prevent the aging. Benchmarks of CPU, memory, I/O, application benchmarks for DBMS, autonomic computing and reliability benchmarks, energy efficiency benchmarks. Computer system capacity planning: instrumentation, simple techniques of time series analysis. Experimental designs: simple experimental design, 2^k factorial design, 1- and 2- factor experiments. Problems with the graphical presentation of data. Queuing systems. Performance simulation.
The labs consist of 15 2-hour exercises starting with a 2-hour introductory instruction session. The exercises are conducted in 2-people teams. The lab exercises cover experimental measurement of

computer hardware and software performance, performance determinants, and performance optimization in e-commerce class applications. Students prepare at least two experimental designs to measure speed of communication links, programs written in binary-compiled vs. interpreted computer languages, CPU performance, memory subsystem performance, file system performance, DBMS performance. During the demo sessions the issues of performance measurement, data presentation, testing WWW servers, popular WWW servers log analysis, web-page load time as a metric of performance and optimizing it, optimizing the Apache server, web-page traffic as a measure of performance, electric energy consumption as a measure of performance, are discussed.

Teaching methods

Lecture: multimedia presentation, illustrated with examples given on the board.
Laboratory exercises: a multimedia presentation, a presentation illustrated with examples given on the board and tasks given by the lecturer, practical exercises.

Bibliography

Basic

1. R.Jain, The Art of Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, Wiely, 1991
2. R.Hockney, The Science of Computer Benchmarking, SIAM Press, Philadelphia, 1996
3. G-P.Musumeci,M.Loukides, Optymalizacja systemów komputerowych,Wydawnictwo RM,Warszawa, 2002
4. B.Gregg, Wydajne systemy komputerowe. Przewodnik dla administratorów systemów lokalnych i w chmurze, Helion, Gliwice, 2014
5. K.Kanoun, L.Spainhower, Dependability Benchmarking for Computer Systems, J.Wiley and Sons, IEEE Computer, 2008
6. J.Błażewicz, W.Cellary, R.Słowiński, J.Węglarz, Badania operacyjne dla informatyków, WNT, Warszawa, 1983
7. NIST/SEMATECH e-Handbook of Statistical Methods, 2003. http://www.itl.nist.gov/div898/handbook/

Additional

www.tpc.org, www.spec.org

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,0
Classes requiring direct contact with the teacher	32	2,0
Student's own work (literature studies, preparation for laboratory classes/tutorials, preparation for tests/exam, project preparation) [1]	68	2

[1] delete or add other activities as appropriate