COURSE DESCRIPTION CARD - SYLLABUS

Course name
Computer networks 2 [S1Inf1>SK2]

Course

Field of study
Computing

Year/Semester
3/5

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
Polish

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
30

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

5,00

Coordinators

dr inż. Michał Kalewski
michal.kalewski@put.poznan.pl

dr inż. Michał Sajkowski
michal.sajkowski@put.poznan.pl

Lecturers

Prerequisites

Knowledge: Student starting this module should have basic knowledge regarding computer systems organization, algorithms and data structures, operating systems, and computer networks 1. Skills: He/she should have skills allowing formulation of algorithms and their programming with the use of at least one widely used software tool. He/she should have skills that are necessary to acquire information from given sources of information. Student should understand the need to extend his/her competences and should express cooperativeness in a team. Social competencies: In addition, in respect to the social skills the student should show attitudes as honesty, responsibility, perseverance, curiosity, creativity, manners, and respect for other people.

Course objective

1. Provide students" knowledge regarding computer networks, within the scope of using, configuration, design and programming of local area and wide area networks, and cognition of technical solutions applied in these networks. 2. Develop students" skills in solving simple problems related to the use and configuration of computer networks. 3. Develop students" skills in team work, especially in configuration, design, and programming of technical solutions applied in computer networks.

Course-related learning outcomes

Knowledge:
1. have well-ordered, theoretically based general knowledge on networking technologies - [K1_W4]
2. have knowledge on important directions of computing science, and other related fields of science, especially electronics, telecommunications, and automatics and robotics - [K1_W5]
3. have basic knowlegle about cycle of life of computing science systems, both hardware and software ones, and especially on processes occuring in them - [K1_W6]
4. Knows basic techiques, methods and tools used in a process of solving of computing science tasks, mainly engineering ones, from the field of key issues in computing science - [K1_W7]

Skills:
1. is able to perform the critical analysis of the way of functioning of computing systems and other computing technical solutions an evaluate these solutions, especially: is able to participate in the software inspection and evaluate software architecture from th point of view of non-functional requirements, and is able to systematic performing of functional tests - [K1_U9]
2. is able - according to given specification - to design connection schema, connect and configure selected items of computer network, using appropriate methods, techniques and tools - [K1_U10]
3. is able to secure data against unauthorized access - [K1_U12]
4. is able to organize, cooperate, and work in a team, accepting various roles in it, and is able to define accordingly the priorities used to the implementation of given task from the area of computer networks - [K1_U18]

Social competences:
1. understands that in computing science both knowledge and skills very quickly become out-of-date - [K1_K1]
2. is aware of the meaning of knowledge in solving engineering problems and knows the examples and understands the reasons of malfunctioning computing systems, which led to serious financial and social losses or to the serious loss of health, or even life - [K1_K2]
3. is able to think and act in an enterprising manner, e.g. finding commercial applications for created software, taking into account not only business benefits, but also social benefits of the performed activity [K1_K3]

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:
Formative assessment:
a) lectures:
based on answers to questions on previous lectures,
b) laboratory classes:
evaluation of doing correctly assigned tasks,
Total assessment:
a) verification of assumed learning objectives related to lectures:
i. based on the sum of answers and the activity during lectures.
ii. evaluation of student"s knowledge and skills obtained in lectures based on written exam, covering from 3 to 5 questions, or from 10 to 15 test questions. In order to obtain positive note, the student should obtain 50% of maximum number of points. During the test, student cannot use any lecture notes, books, etc.
b) verification of assumed learning objectives related to laboratory classes:
i. evaluation of student"s skills related to carrying out the lab tasks and configuration task,
ii. monitoring student"s continuing activities during classes,
iii. evaluation of student"s skills based on one or two tests, covering from 10 to 15 questions,

iv. an assessment of the ability to implement a programming task independently during the laboratory activities,

Programme content

BSD Internet sockets interface.
Streaming and datagram network sockets of the IPv4 protocol.
Constructions of concurrent network servers using sub-processes and input-output multiplexing mechanisms.
Encrypted (SSL/TLS) BSD network sockets.
Mechanisms for handling network communication in GUI applications.
Network socket interface in Windows operating systems (winsock).
Hypertext Transfer Protocol (HTTP).
Domain Name System (DNS).
Electronic mail.

Course topics

The lecture should cover the following topics
- Introduction to the BSD network sockets.
- System and auxiliary functions of internet communication of the BSD network sockets and DNS domain name system support.
- Streaming (TCP) network sockets for the IPv4 protocol.
- Construction of network servers with the use of sub-processes and I/O multiplexing.
- Buffing in streaming internet communication.
- Encrypted (SSL/TLS) BSD network sockets.
- Network communication mechanisms in applications with a graphic user interface.
- Datagram (UDP) network sockets for the IPv4 protocol.
- Network sockets in Windows operating systems (Winsock).
- HTTP protocol.
- DNS system.
- E -mail: SMTP and IMAP protocols and MIME format.
- Traffic control in Internet networks.
- Introduction to the Software-Defined Networking and OpenFlow.

The lab-classes should cover the following topics:
1) Implementation of a network client application using the sockets interface,
2) Implementation of an iterative network server using the sockets interface,
3) Implementation of concurrent network servers using the sockets interface,
5) Buffering in connection-oriented TCP communication,
4) Implementation of encrypted connection-oriented communication using the sockets interface and the TLS protocol,
5) Implementation of datagram communication using the sockets interface,
6) Handling typical behavior of socket interface functions during network errors and anomalies,
7) Implementation of network applications with a graphical user interface,
8) DNS,
9) HTTP,
10) Email system and SMTP,
11) Final project.

Teaching methods

Lectures: multimedia presentation, presentation illustrated with examples presented on blackboard.
Labs: solving tasks, practical exercises with use of network devices, discussion, teamwork, multimedia showcase, configuration task verified during laboratory classes, final project implemented at home.

Bibliography

Basic
1. TCP/IP Protocol Suite, 4th edition, B.A. Forouzan, McGraw-Hill Education, New York, 2009
2. Data Communications and Networking, 5th ed., B.A. Forouzan, McGraw-Hill Education, New York 2012

3. Computer Networks, 5th edition, A.S. Tanenbaum, D.J. Wetherall, Pearson, Boston, 2011
4. Computer Networking: A Top-Down Approach, 7th edition, J.F. Kurose, K.W. Ross, Pearson Education, Boston, 2016
5. Computer Networks: A Systems Approach, L.L. Peterson, B.S. Davie, 5th edition, Morgan Kauffmann, San Francisco, 2012
Additional
1. Network Analysis and Troubleshooting, J. Scott Haugdahl, Addison-Wesley, 1999

Breakdown of average student's workload

	Hours	ECTS
Total workload	125	5,00
Classes requiring direct contact with the teacher	62	2,50
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	63	2,50