COURSE DESCRIPTION CARD - SYLLABUS

Course name
Smart card programming [N2Inf1-IWPB>PKE]

Course

Field of study
Computing

Year/Semester
1/2

Area of study (specialization)
Information Technology in Business Processes

Profile of study
general academic

Level of study
second-cycle

Course offered in
Polish

Form of study
part-time

Requirements
elective

Number of hours

Lecture
16

Laboratory classes
18

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

4,00

Coordinators

dr hab. inż. Marek Mika
marek.mika@put.poznan.pl

Lecturers

Prerequisites

The student should have basic knowledge of electronics, operating systems and cryptography.She/he should also have the skills to: solve basic problems in the field of application design, programming in high-level languages and obtain information from the printed and/or online sources. Moreover she/he should understand the necessity to expand her/his competences.

Course objective

Provide students with the basic knowledge regarding smart cards (standards, applications, designing and programming systems which use smart cards and/or automated identification systems). In addition, developing students" skills in designing and programming systems which use smart cards.

Course-related learning outcomes

Knowledge:
1. The student has orderly, theory-based, general knowledge in the field of construction, principles of operation, programming and applications of smart cards.
2. The student has knowledge of: construction of terminals and smart cards, transmission protocols used in smart cards, smard cards operating systems, communication of the card with the terminal,

programming and applications of smart cards.
3. The student has knowledge of development trends and the most important new achievements in the field of smart cards.
4.The student knows the areas and examples of practical applications of smart cards.

Skills:
1. The student is able to design the software for the smart card or the selected automated identification system, according to a predefined specification which takes into account non-technical aspects, and carry out this project at least in part using appropriate methods, techniques and tools
2. The student is aware that during designing software for a smart card or automated identification system, sometimes there is a need to reach for the right standard or specification and apply in practice the knowledge presented in them.

Social competences:
1. The student understands that in the field of smart cards knowledge and skills can quickly become obsolete.
2. She/he understands the importance of standards and specifications in the field.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:
Assessment of knowledge acquired during the lecture is based on a written colloquium in the form of a test, which may include 20 to 50 open and closed questions. In case of closed questions it is a multiple-choice test. The score of individual questions is given in the content of the question. The form of the test and the issues to which it applies are discussed during one of the last lectures. For a score of 3.0 the student should get at least 50% of points, 3.5 at least 60% of points, 4.0 for at least 70% of points, etc.
In terms of laboratories, the verification of the assumed educational results is carried out by:
1. the assessment of the tasks carried out within the framework of subsequent laboratory classes, for each correctly performed task a maximum of 1 point can be obtained; on the basis of the number of points scored, a partial assessment is given.
2. the final test covering the issues practiced within the laboratory classes, the test consists of randomly selected questions concerning each of the exercise topics; for each correct answer 1 point can be obtained; on the basis of the number of points scored, a second partial assessment is given
3. the final grade is given on the basis oftwoe partial grades, as a weighted average

Programme content

The lecture program includes the following issues:
The origin of smart cards. Overview of basic SC applications. The role of standardization. Card types (convex, with magnetic stripe, contact and contactless memory cards, contact and contactless processor cards, multi-megabyte, optical). Physical features of smart cards (formats, contacts, materials, security features, chip modules). Electrical features of SC (contacts, voltage and current supply, clock, data transmission, activation and deactivation sequences). Smart card microcontrollers (semiconductor technologies, processor types, memory types, communication modules, clock and other modules). Data structures. Alphanumeric data encoding. SDL notation. SC as a finite state machine. Error detection and correction codes. Data compression. Cryptology (symmetric encryption algorithms: DES, AES, IDEA, COMP128, Milanage; asymmetric encryption algorithms: RSA, DSS, elliptical curve algorithm; multiple encryption; data alignment; message authentication and cryptographic checksum), hash functions, random number generation and testing, card and reader authentication (unilateral symmetric, bilateral symmetric, static asynchronous, dynamic asynchronous), digital signatures, certificates, key management, person authentication. Communication with card (messages: ATR, PPS, APDU). Secure data transmission between card and reader. Logical channels and protocols. Conecting terminals with higher level systems. Data transmission for contact cards (transport layer, memory card protocols, T=0 and T=1 transmission protocols, USB, MMC and SWP protocols). Data transmission for contactless cards (inductive and capacitive feedback, power transfer, data transfer, NFC, short- and long-range contactless cards, proximity cards). SC programming (commands: file, read and write, search, person and device authentication, cryptography, file and application management, completion, hardware testing, database, data transmission). Commands related to card application (for electronic purses, for credit and debit cards). Managed electronic card files (file structure, file life cycle, file types, file names, file

selection, EF file structure, access conditions, attributes). SC operating systems (basic assumptions and functions, command processing, design and implementation rules, card completion, memory organization and management, file management, resource access, atomic operations, multitasking, performance, application management, national codes). Types of SC operating systems: JavaCard, Multos, BasicCard, Linux, Small-OS. Production and quality assurance of electronic cards. Security of smart cards (types of attacks, attack history, attacks and defense during design, production and use). Smart card readers (physical and electrical features, user interface, application interface, security). SC applications in: payment systems, telecommunication systems, healthcare systems, transport systems, identification, passports, IT security. Application design.
Laboratory classes are conducted in the form of 2-hour exercises, taking place in the laboratory. The exercises are divided into two parts. In the first part, students perform subsequent practical exercises concerning various technologies. This part ends with a test that checks the acquired knowledge. The second part is related to the practical or theoretical project. The laboratory program includes the following issues:
Operating the following types of smart cards: JavaCard, SIM, BasicCard, .NET and student ID card. Encryption. Handling and storage of encryption keys and digital signature on the card. Languages and techniques of electronic card programming. Applications of smart cards. Handling of barcodes: coding, printing, reading. RFID technology reading and writing of RFID tags. Exercises in NFC technology.

Course topics

The lecture program includes the following issues:
The origin of smart cards. Overview of basic SC applications. The role of standardization. Card types (convex, with magnetic stripe, contact and contactless memory cards, contact and contactless processor cards, multi-megabyte, optical). Physical features of smart cards (formats, contacts, materials, security features, chip modules). Electrical features of SC (contacts, voltage and current supply, clock, data transmission, activation and deactivation sequences). Smart card microcontrollers (semiconductor technologies, processor types, memory types, communication modules, clock and other modules). Data structures. Alphanumeric data encoding. SDL notation. SC as a finite state machine. Error detection and correction codes. Data compression. Cryptology (symmetric encryption algorithms: DES, AES, IDEA, COMP128, Milanage; asymmetric encryption algorithms: RSA, DSS, elliptical curve algorithm; multiple encryption; data alignment; message authentication and cryptographic checksum), hash functions, random number generation and testing, card and reader authentication (unilateral symmetric, bilateral symmetric, static asynchronous, dynamic asynchronous), digital signatures, certificates, key management, person authentication. Communication with card (messages: ATR, PPS, APDU). Secure data transmission between card and reader. Logical channels and protocols. Conecting terminals with higher level systems. Data transmission for contact cards (transport layer, memory card protocols, T=0 and T=1 transmission protocols, USB, MMC and SWP protocols). Data transmission for contactless cards (inductive and capacitive feedback, power transfer, data transfer, NFC, short- and long-range contactless cards, proximity cards). SC programming (commands: file, read and write, search, person and device authentication, cryptography, file and application management, completion, hardware testing, database, data transmission). Commands related to card application (for electronic purses, for credit and debit cards). Managed electronic card files (file structure, file life cycle, file types, file names, file selection, EF file structure, access conditions, attributes). SC operating systems (basic assumptions and functions, command processing, design and implementation rules, card completion, memory organization and management, file management, resource access, atomic operations, multitasking, performance, application management, national codes). Types of SC operating systems: JavaCard, Multos, BasicCard, Linux, Small-OS. Production and quality assurance of electronic cards. Security of smart cards (types of attacks, attack history, attacks and defense during design, production and use). Smart card readers (physical and electrical features, user interface, application interface, security). SC applications in: payment systems, telecommunication systems, healthcare systems, transport systems, identification, passports, IT security. Application design.
Laboratory classes are conducted in the form of 2-hour exercises, taking place in the laboratory. The exercises are divided into two parts. In the first part, students perform subsequent practical exercises concerning various technologies. This part ends with a test that checks the acquired knowledge. The second part is related to the practical or theoretical project. The laboratory program includes the following issues:
Operating the following types of smart cards: JavaCard, SIM, BasicCard, .NET and student ID card. Encryption. Handling and storage of encryption keys and digital signature on the card. Languages and techniques of electronic card programming. Applications of smart cards. Handling of barcodes: coding,

printing, reading. RFID technology reading and writing of RFID tags. Exercises in NFC technology.

Teaching methods

1.Lecture: multimedia presentation.
2. Laboratory exercises: task solving, practical exercises.

Bibliography

Basic
1. K. Mayes, K. Markantonakis (ed.), Smart cards, tokens, security and applications (2-nd edition), Springer, 2017 (https://link.springer.com/content/pdf/10.1007%2F978-3-319-50500-8.pdf)
2. M. Kubas, M. Molski: Karta elektroniczna : bezpieczny nośnik informacji, Mikom, 2002
3. W. Rankl, W. Effing: Smart card handbook (4-th edition), Wiley, 2010
4. U. Hansmann, M. S. Nicklous, T. Schäck, A. Schneider, F. Seliger: Smart Card Application Development Using Java (2-nd edition), Springer 2012. (https://link.springer.com/book/10.1007%2F978-3-642-55969-3)
5. www.smartcardbasics.com
Additional
1. S. Mangard, E. Oswald, T. Popp: Power analysis attacks: Revealing the secrets of smart cards, Springer, 2007 (https://link.springer.com/content/pdf/10.1007%2F978-0-387-38162-6.pdf)
2.U. Chirico: Smart Card Programming, (2-nd edition), Lulu.com, 2015.

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,00
Classes requiring direct contact with the teacher	36	1,50
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	64	2,50