TRƯỜNG ĐẠI HỌC KHOA HỌC VÀ CÔNG NGHỆ HÀ NỘI

UNIVERSITE DES SCIENCES ET DES TECHNOLOGIES DE HANOI

UNIVERSITY OF SCIENCE AND TECHNOLOGY OF HANOI

TEACHING PROFILE

UNDERGRADUATE SCHOOL

MAJOR: INFORMATION AND COMMUNICATION TECHNOLOGY

Academic year: 2017 - 2018

Table of Contents

Contents

UNIVERSITY OF SCIENCE AND TECHNOLOGY OF HANOI
UNDERGRADUATE PROGRAM

Title: Information and Communication Technology
Level: Undergraduate
Diploma: Bachelor

Program Overview

In the last few decades, ICT has always been the key sector at the forefront of science and technology. Due to that fact, ICT is considered as the science of future, its development is important for every country as for Vietnam. ICT is a multidisciplinary science with the Information Technology, Computer Science, Software Engineering and Telecommunication.

The concepts, methods and applications involved in ICT are constantly evolving on an almost daily basis. ICT applications can deliver basic services in a wide range of sectors including: health, agriculture, education, public administration, commerce, among others. Given the right approach, context and implementation processes, investments in ICT applications and tools can result in productivity and quality improvements.

The ICT formation in USTH prepares graduates for careers in various fields of Software Engineering and Computer Science. It focuses on applied research and emerging technologies. The ICT program equips students with the fundamental understanding of computing and system administration, then the professional knowledge of information systems, advance databases and eventually management of projects. Besides the formation aims to train them capable of working effectively in multidisciplinary teams and in an international environment; taking appropriate position in industry upon graduation and growing into leadership position, and also pursuing scientific research.

Program Formation

The ICT curriculum provides a solid foundation in mathematics, informatics, software engineering and communication. The program also builds a core of computer science for higher education. Additionally, students can specialize in particular areas such as mobile and web development, security and system, data

mining, … etc. by selecting the appropriate technical electives and completing the "Group project" and the internship of at least 3 months.

1. Educational Objectives

Information and Communication Technology program at USTH provides students:

- Basic knowledge and skills to work for Information& Communication Technology;
- Specialized skills, social skills and personal characteristics to efficiently work in multi-sector and international enterprises;
- Political qualities, morality, health and sense of serving community;

2. Expected Learning Outcomes

Bachelor Program in Information and Communication Technology, University of Science and Technology of Hanoi (USTH) aims to provide students the basic knowledge and solid professional knowledge, necessary social skills to work effectively in multidisciplinary teams and in the international environment; personal qualities and and sense of life-long learning to succeed in career; political qualities and sense of serving community. These objectives are clearly reflected in the below expected learning outcomes:

1. Solid professional knowledge to adapt to various jobs of researching theories, modeling ideas, designing and develping technology solutions, consulting and managing required systems in field of Information and Communication Technology.
   - 1.1 Ability to apply basic knowledge of mathematics, physics, statistic probability… to describe, caculate and simulate, design and develop Computer systems.
   - 1.2 Ability to apply basic knowledge of programming, coding, database, discrete mathematics, operation systems, information systems, computer structures, computer networks, … to do research and analyse systems, products in Information Technology.
   - 1.3 Ability to apply core knowledge of Information Technology, accompanying with ability to utilize advanced tools to design, implement, control and evaluate ICT systems; ability to analyse practical issues, design and develop Computer applications.
2. Personal and professional skills and attributes:
   - 2.1 Reasoning and problem solving
   - 2.2 Experimentation and knowledge discovery
   - 2.3 System thinking and critical thinking
   - 2.4 Creativity, activeness and seriousness
   - 2.5 Professional Ethics, Integrity, Responsibility and Accountability
   - 2.6 Contemporary issue understandings and lifelong learning
3. Interpersonal skills for teamwork and international communication:
   - 3.1 Leadership and skills to work in Cross-disciplinary teams
   - 3.2 Communication skills: written communication, electronic/multimeadia communication, presentation, persuasive argument, effective negotiation
   - 3.3 English skills, IELTS $\geq$ 6.5.
4. Conceiving, designing, implementing and operating systems/products/technical solutions in the enterprise and social context:
   - 4.1 Awareness of relationship between ICT solutions and socio-economic and global contexts.
   - 4.2 Ability to identify problems, form ideas for technical solutions, participate in ICT projects.
   - 4.3 Ability to design and develop systems, products, technical solutions, plattform, network, database, common application infrastructure, system management, security, application systems, application package.

4.4 Ability to install and operate systems, products, technical solutions, plattform, network, database, common application infrastructure, system management, security, application systems, application package.

4.5 Ability to utilize and maintain systems, products, technical solutions, plattform, network, database, common application infrastructure, system management, security, application systems, application package.

5. Qualification & Certification:

5.1 Qualification on bachelor level of the MOET's general provisions

5.2 Qualification on international bachelor level based on European Credit Transfer and Accumulation System (ECTS).

I. SECOND YEAR PROGRAM

MATH 2.1: PROBABILITIES & STATISTICS

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	25	15	0	**40**

3. **Prerequisites**: N/A
4. **Recommended background knowledge**: Analysis, Integration and ODE
5. **Subject description:**

The course of Probability and Statistics covers topics such as: random, variables, probability distributions, random sampling, regression, analysis of variance, estimation and hypothesis testing.

6. **Objectives & Outcome:**
   1. Ability to manipulate sets and events, use axioms of probability, computer probabilities of events & solve counting problems.
   2. Ability to generate random variables from experiments, use probability distribution and density functions, use common distributions and computer statistical quantities random variables and random vectors.
   3. Ability to determine the distributions and statistical quantities of random variables, to apply limit theorems.
   4. Ability to understand basic statistical concepts, random sampling, statistical inference, to regression and correlation.
   5. Ability to construct point estimation, confidence and prediction intervals.
   6. Ability to test hypotheses
   7. Ability to perform Analysis of Variance.
7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	10	0	0	30	50

8. **Prescribed Textbook(s):**

[1].Spiegel R. M., Schiller J., Srinivasan R. A., *Probability and Statistics*, Third edition, McGraw Hill, 2009.

[2].Montgomery C. D., Runger C. G., *Applied Statistics and Probability for Engineers*, Fifth edition, John Wileys& Sons, 2011.

II. Course content & schedule:

1. **Topic 1: Basic Probability**

Random Experiments, Sample Spaces and Events, Algebra of Events

The Concept and Axioms of Probability

Theorems on Probability, Assignment of Probabilities, Conditional Probability, Independent Events, Bayes' Rule, Permutations, Combinations, Binomial Coefficients, Stirling's Approximation.

2. **Topic 2: Random Variables and Probability Distributions**

Random Variables, Discrete and Continuous Random Variables, Cumulative Distribution, Functions of Random Variables, Graphical Interpretations, Joint Distributions, Independent Random Variables, Change of Variables, Probability Distributions of Functions of Random Variables, Convolutions, Conditional Probability Functions.

3. **Exercises for topic 1 & 2**
4. **Topic 3: Mathematical Expectation**

   Definition of Mathematical Expectation, Expectation of a Random Variable and a Function of Random Variables, The Variance and Standard Deviation, Moments, Moment Generating Functions, Characteristic Functions, Variance for Joint Distributions, Covariance, Correlation Coefficient, Conditional Expectation, Conditional Variance and Moments, Chebyshev's Inequality, Weak Laws of Large Numbers, The Central Limit Theorem, Mode and Median, Skewness and Kurtosis.
5. **Topic 4: Special Probability Distributions**

   The Binomial Distribution and its properties
   The Normal Distribution and its properties
   The Poisson distribution and its properties
   Relations between the Binomial, Poisson and Normal Distributions
   The Multinomial Distribution
   The Hyper geometric Distribution
   The Uniform Distribution
   The Gamma Distribution
   The Cauchy Distribution
   The Gamma Distribution
   The Beta Distribution
   The Chi-Square Distribution
   Student's *t* Distribution
   The *F* Distribution
   Relationships among Chi-Square, *t,* and *F* Distributions
   The Bivariate Normal Distribution
   Geometric and Pascal's Distribution
   Exponential Distribution
   Maxwell Distribution
6. **Exercises for topic 3 and 4.**
7. **Topic 5 Sampling Theory**

   Random Sampling and Statistical Inference, Sampling With and Without Replacement, Population Parameters, Sample Statistics, Approximate Distribution of the Sample Mean, Sampling Distributions (of Means, of Differences and Sums, of Variances)
   Sampling Distribution of Ratios of Variances *(F - Distribution)*
   Other Statistics: Frequency Distributions, Relative Frequency Distributions; Computation of Mean, Variance, and Moments for Grouped Data.
8. **Topic 6 Estimation Theory**

   Unbiased Estimates and Efficient Estimates,
   Point Estimates and Interval Estimates,
   Reliability,
   Point Estimation of Mean, Variance, and Proportion,
   Confidence Intervals for Means, Proportions,
   Differences and Sums, and for the Variance of a Normal Distribution,
   Confidence Intervals for Variance Ratios,
   Maximum Likelihood Estimates,
   Maximum Likelihood Estimator of a Bernoulli Parameter, Poisson Parameter, and normal Parameters.
   Bayes Estimator * (optional)
9. **Exercises for topic 5 and 6**
10. **Topic 7 Tests of Hypotheses and Significance**

Statistical Hypotheses, Null Hypotheses
Type I and Type II Errors, Level of Significance
Tests Involving the Normal Distribution
One-Tailed and Two-Tailed Tests
P Value
Special Tests of Significance (for Large Samples, and for Small Samples)
Relationship between Estimation Theory and Hypothesis Testing
Operating Characteristic Curves
Power of a Test
Fitting Theoretical Distributions to Sample Frequency Distributions
The Chi-Square Test for Goodness of Fit
Contingency Tables

11. Topic 8 Curve Fitting, Regression, and Correlation

Curve Fitting, Regression,
The Method of Least Squares,
Least Squares Estimators of the Regression Parameters,
The Least-Squares Line in and Covariance, Multiple Linear
Terms of Sample, Variances Regression,
Distributions of the Estimators, Standard
Distributions of the Error of Estimate,
The Linear Correlation Coefficient,
Generalized Correlation Coefficient,
Rank Correlation,
Probability Interpretation of Regression and Correlation,
Sampling Theory of Regression and Correlation, Correlation and Dependence

12. Exercises for topic 7 and 8

13. Topic 9 Analysis of Variance

The Purpose of Analysis of Variance,
One-Way Classification or One-Factor Experiments,

- Total Variation, Variation within Treatments, Variation between Treatments,
- Linear Mathematical Model for Analysis of Variance, Expected Values of the Variations,
- Distributions of the Variations,
- The *F* Test for the Null Hypothesis of Equal Means,
- Analysis of Variance for Unequal Numbers of Observations,

Two-Way Classification or Two-Factor Experiments,

- Variations for Two-Factor Experiments,
- Analysis of Variance for Two-Factor Experiments,
- Two-Factor Experiments with Replication

14. Topic 10 Nonparametric Tests

The Sign Test,
The Mann–Whitney *U* Test; The Kruskal–Wallis *H* Test; The *H* Test Corrected for Ties; The Runs Test for Randomness; Spearman's Rank Correlation.

15. Exercises for topic 9 and 10

III. Reference Literature:

[1]. Course notes, in-class exercises and homework problems.
[2]. Spiegel R. M., Schiller J., Srinivasan R. A., *Probability and Statistics*, Third edition, McGraw Hill, 2009.
[3]. Montgomery C. D., Runger C. G., *Applied Statistics and Probability for Engineers*, Fifth edition, John Wileys& Sons, 2011.
[4]. Vidakovic B., *Statistics for Bioengineering Sciences*, Springer 2011.

MATH 2.2: ANALYSIS& ALGEBRAIC STRUCTURES

I. Course description:

1. **Credit points: 4 ECTS**
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	20	16	2	**38**

3. **Prerequisites:**
   Sets, Boolean algebra, mappings, functions, real analysis of one variable, linear algebra/ University; Number theory (divisibility)/ from 6-9 classes; programming/12 class, University.
4. **Recommended background knowledge:**
   Math reasoning, expansions of the function into series
5. **Subject description:**
   This course provides the elementary knowledge on the analysis of several variables: the differentiability concept, integration understanding from the point of view of vector analysis. The algebraic part helps the learners to conquest the mathematical basics of computer sciences in the semi groups, groups, ring, relations, field's terminologies and application in daily using.
6. **Objectives & Outcome:**
   Basic understanding of how the multivariate calculus differs from 1D case: the double limits, continuity, partial derivatives, integration over the domain and curvilinear integrating technics. Students should be able to be familiar with some applications in the optimization problem and graphing visualization. Many new algebraic concepts will be introduced in the relations with ICT branch' preparation: the learners should be known about the built-in structures in the networking basing on partitions, filters ordering of the most simple PC' data processing.
7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	5	5	0	0	20	70

8. **Prescribed Textbook(s)**
   [1] W. F. Trench, Introduction to real analysis (Available online)
   [2] Milner, Group theory, Lecture notes, 2003 (Available online for personal use)

II. Course content & schedule:

1. **Topic 1: Calculus of several variables**
   - **1.1. The limits, continuity of functions**
   - Limits as a new concept in the metric space.
   - Equivalent definitions of the continuity.
   - **1.2. The differentiability**
   - Derivative as the linear mapping.
   - Partial derivatives.
   - Directional derivative, the meaning.
   - Optimization: Lagrange method of multipliers.
   - **1.3. The integration technics**
   A. Multiple integrals
   - Definitions, properties.
   - Changing of variables: Jacobian, polar coordinates, cylindrical coordinates.
   - Volume, center of mass findings.
   B. Curvilinear integrals
   - Parameterizations of the curve.
   - The first kind of curvilinear integrals: in the relation with the length element.
   - The second kind of curvilinear integrals: the orientation, Green formula.

- Method of re-parameterizations.

C. Surface integrals

- Several parameterizations of the surface.
- The area as the integral element for the 1-st kind integral.
- The differential form in 2-nd kind integral.
- Orientation' influence on the sign of integrals.
- The Gauss formula.
- The Stokes theorem.

1.4. Applications:

- The graphing computerizing illustrations: Explanation of graphs, dragging mode while graphing, understanding the role of parameters, exploring the curves and special surfaces.

2. Topic 2: Algebraic structures

2.1. Basic definitions

- Groups, subgroups
- Order
- Cyclic groups
- Cossets
- Normal subgroups

2.2. Semi groups as the flows

- Free semi groups.
- Generators and relations.
- Finitely presented groups.

2.3. Groups acting on the sets

- Permutations.
- The Todd-Coxeter algorithm.

2.4. Computer algebra: algebra of the polynomials, power series, rational functions

- Rings and Fields
- Divisibility and Factorization of the domains.
- Euclidean algorithm.
- Quotient Field and Rational functions.
- The data structures for the polynomials, rational functions and power series.

III. Reference Literature:

[1]. G.Cain, J.Herod: Multivariable calculus, 1997.
[2]. W. F. Trench, Introduction to real analysis (Available online)
[3]. Milner, Group theory, Lecture notes, 2003 (Available online for personal use)
[4]. Gedder et al., Algorithms for computer algebra, Kluwer Academic Publishers, 1992.

ICT 2.1: ALGORITHMS AND DATA STRUCTURES

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	20	15	0	**36**

3. **Prerequisites**: Basic Programming
4. **Recommended background knowledge**: Programming languages (C), Mathematics
5. **Subject description:**

Data Structures are the programmatic way of storing data so that data can be used efficiently. Almost every enterprise application uses various types of data structures in one or other way. This course will provide basic concepts for algorithms, data structures. The goal is to introduce the various abstract data types such as linear and non-linear data structures. Besides, for the efficient use of data, sorting and searching data algorithms used in real world applications. are presented in this course.

6. **Objectives & Outcome:**

On the completion of this course students should be able to:

- Understand the basic programming concepts
- Understand analysis and design of computer algorithm and data structures.
- Analyze real-world problems and design appropriate data structures according to the problems using techniques from the course
- Work individually or in a group

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	30	0	0	60

8. **Prescribed Textbook(s):**

[1]Adam Drozdek, *Data Structures and Algorithms in C++*, 4th Edition, Cengage Learning, 2012

II. Course content & schedule:

1. **Basic Programming Concepts in C++**
2. **Elementary Data Structures**
3. **Abstract (Linear) Data Types I**
4. **Abstract (Linear) Data Types II**
5. **Recursive Algorithms**
6. **Trees**
7. **Sorting Algorithm I**
8. **Sorting Algorithm II**
9. **Searching Algorithm**
10. **Graphs**

III. Reference Literature:

[1] Adam Drozdek, *Data Structures and Algorithms in C++*, 4th Edition, Cengage Learning, 2012

ICT 2.2: OBJECT-ORIENTED PROGRAMMING

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	26	0	14	**40**

3. **Prerequisites**: Basic programming
4. **Recommended background knowledge**:
   Students are recommended to master the following basic programming skills:

- Understand fundamental concepts of programming
- Basic data types, array, string, pointer, etc,
- Basic algorithms such as sorting, searching, etc
- Have good programming style (e.g. comments, indentation, naming, etc)

5. **Subject description:**
   Object-Oriented Programming is a fundamental programming methodology for creating computer applications. Object-Oriented Programming offers a wide range of advantages over procedural programming such as easier debugging, easier code reuse, etc. This course will provide students with essential object-oriented programming concepts, principles, and techniques in order to create object-oriented computer applications. The concepts will be illustrated using the Java programming language. The topics covered include: objects and classes, encapsulation, inheritance, polymorphism, abstraction, interfaces, exception handling, and input/output streams.
6. **Objectives & Outcome:**
   At the end of the course, students will be able to:

- Understand the basic principles and concepts of object-oriented programming
- Practice how to use these concepts and principles with Java programming language
- Apply object-oriented techniques to develop computer programs

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	20	0	0	50

8. **Prescribed Textbook(s):**
   [1] Kathy Sierra, Bert Bates, *Head First Java*, 2nd Edition, O' Reilly, 2005
   [2] Deitel & Deitel, Java How to Program, 9th Edition, Prentice Hall, 2011.

II. Course content & schedule:

1. **Course introduction - Introduce to Object-oriented programming**
2. **Introduction to Java programming language**
3. **Objects and classes - Instance variables and methods – Encapsulation**
4. **Objects and object references**
5. **Class members vs. Instance members - Java packages**
6. **Inheritance - Polymorphism**
7. **Assignment 1 presentation**
8. **Abstraction**
9. **Interfaces**
10. **Exceptions**
11. **Data structures**
12. **I/O Streams**
13. **Principles of object-oriented design**
14. **Assignment 2 presentation**

III. Reference Literature:

[1]. Kathy Sierra, Bert Bates, Head First Java, 2nd Edition, O' Reilly, 2005.

[2]. Deitel&Deitel, *Java How to Program*, 9th Edition, Prentice Hall, 2011.

[3]. Java™ Platform, Standard Edition 7 API Specification, http://docs.oracle.com/javase/7/docs/api/index.

ICT 2.3: SIGNALS AND SYSTEMS

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	20	10	6	**36**

3. **Prerequisites**:
4. **Recommended background knowledge**:
5. **Subject description:**

   This course covers the fundamentals of signal and system analysis. It develops mathematical transform techniques and computational tools to analyze the behavior of analog and digital systems. The main focus is on representations of discrete-time and continuous-time signals, as well as discrete-time and continuous-time systems in time domain and frequency domain.

6. **Objectives & Outcome:**

   When completing this course, it is expected that a student will have the ability to:

- Recognize the terminology that is used in the Signal Processing field.
- Apply transform techniques and concepts in formulating and solving engineering problems.
- Think critically, ask

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	20	10 (bonus)	0	0	30	50

8. **Prescribed Textbook(s):**

II. Course content & schedule:

1. **Signals, Systems, Operation/Transformations**
2. **Discrete time LTI system**
3. **Continuous LTI system**
4. **Properties of LTI system, Frequency Response**
5. **Fourier Series of Continuous-Time Periodic Signals**
6. **Fourier Series of Discrete-Time Periodic Signals**
7. **Continuous-time Fourier Transform**
8. **Fourier Transform: signals & systems**
9. **Discrete-time Fourier Transform**
10. **Sampling Theorem, Reconstruction/Interpolation**
11. **Review**
12. **Octave Introduction**

III. Reference Literature

[1] Signals and Systems [2nd Edition] by Alan V. Oppenheim and Alan S. Willsky
[2] Linear Systems and Signals [2nd Edition] by B. P. Lathi

ICT 2.4: COMPUTER ARCHITECTURE

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	26	0	12	**38**

3. **Prerequisites**: Computer Architect, Programming.
4. **Recommended background knowledge**: Digital Electronics
5. **Subject description:**

   Advanced Computer Architecture describes the computer hardware structure such as pipeline, memory architecture, the how to develop software program that make use of computer architecture.
6. **Objectives & Outcome:**

   Students should gain knowledge about:
   - Advanced Computer Hardware Behavior and Design: Pipeline, Memory Architecture, Superscalar, Multiple Issue Processor
   - Estimate and measure the computer performance: Clock Per Instruction, Frequency, Number of Instruction.

   Students should have skill to:
   - Develop program that can use the computer architect to improve the performance.
7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	10	0	10	70

8. **Prescribed Textbook(s):**[1]*Computer Org and Design*, 3rd Ed., Patterson &Hennessy, ©2007

II. Course content & schedule:

1. Chapter 1: Simple Machine Implementation
- MIPS Instruction Set Architecture (Revised):
  - ✓ Definition of Instruction Set Architecture
  - ✓ Concept of Instruction Set Architecture families
  - ✓ MIPS ISA: Instruction Types and formats; Syntax and Sematics of MIPS Instructions
- MIPS Processor Specification:
  - ✓ Input, Output of the CPU
  - ✓ Programmer Visible Registers of the CPU
  - ✓ Initialization of the CPU
  - ✓ Functionalities of the CPU

2. Chapter 1: Simple Machine Implementation
- Simple MIPS Processor Implementation:
  - ✓ Single Cycle Implementation
  - ✓ Performance of Single Cycle Implementation
  - ✓ Multiple Cycle Implementation
- Pipeline Implementation
  - ✓ Basic Concept

3. Chapter 1: Simple Machine Implementation
- Pipeline Implementation
  - ✓ 5 pipeline stages in MIPS Processor
  - ✓ Description of Pipeline Behavior
  - ✓ Pipeline Performance

- ✓ Pipeline Implementation
- Pipeline Structure Hazard and Solutions
  - ✓ Structure Hazard
  - ✓ Pipeline block duplications

4. **Laboratory work 1: MIPS Assembly Language**
   - MIPSIT Studio tool for MIPS program development and emulation
   - Develop MIPS program to implement basic algorithms such as sum, sort, serial multiplication
5. **Chapter 1: Simple Machine Implementation**
   - Pipeline Data Hazard and Solutions
     - ✓ Data hazard and its effect on performance
     - ✓ Data hazard solution: Stall, Forwarding
6. **Chapter 1: Simple Machine Implementation**
   - Pipeline Control Hazard and Solutions
   - Control hazard
   - Control hazard solution: Cancel instructions, Control hazard, Early estimate branch condition
   - Control hazard solution: delayed slot, branch prediction
7. **Chapter 2: Memory Architecture**
   - Introduction
     - ✓ Memory technology: ROM, DRAM, SRAM
     - ✓ Memory performance vs. CPU performance
     - ✓ Memory Architecture Design Problem
8. **Laboratory work 2: MIPS Pipeline behavior**
   - MIPSIT Simulator for MIPS Pipeline simulation
   - Study pipeline behavior with/without hazard solutions.
9. **Chapter 2: Memory Architecture**
   - Simple Memory Hierarchy:
     - ✓ One level cache
     - ✓ Directed mapped cache
     - ✓ Performance of directed mapped cache: Hit/Miss Rate, Miss Penalty, AMAT
10. **Chapter 2: Memory Architecture**
    - Advanced Memory Cache Design:
      - ✓ Associative Cache
      - ✓ Performance of associative cache: Hit/Miss Rate, Miss Penalty, AMAT
      - ✓ Effect of program structure and cache architecture on performance
    - Many Level Memory Architecture
      - ✓ Basic Idea
      - ✓ Performance
      - ✓ Virtual Memory Concept
11. **Chapter 3: Modern Computer Architecture**
    - Advance Pipelining Techniques
    - Static Multiple Issue
    - Dynamic Multiple Issue
12. **Laboratory work 3: Memory Performance**
    Evaluate program execution time with respect to different memory architecture
13. **Laboratory work 4: Memory Performance**
    Develop program that make use of memoryarchitecture

III. Reference Literature:

[1]. *Digital Design and Computer Architecture*, David Money Harris

[2]. Computer Architecture: A Quantitative Approach, 3rd Edition, Hennessy & Patterson

MATH2.3: DISCRETE MATHEMATICS

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	20	10	9	**30**

3. **Prerequisites**: Linear Algebra
4. **Recommended background knowledge**:
5. **Subject description:**

The aim of this course is to understand and use discrete structures that are backbones of computer science. Topics included are logic, proofs, sets, relations, induction, recursion, combination and counting with an emphasis on applications in computer science.

6. **Objectives & Outcome**

In terms of outcomes, students will be able to:

- Comprehend and use propositional and predicate logic.
- Understand naive set theory, set operations.
- Use concepts of relations; perform various operations with relations and functions (congruence, methods of proof, induction, recursion, etc..).
- Solve problems in combinatorics (permutations, combinations, etc..).
- Providing knowledge to learning other topics: Probabilities, Cryptography, Algorithm & Data Structures, Databases,...

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	20	0	0	0	0	80

8. **Prescribed Textbook(s)**

[1]. Kenneth A. Rosen, Charles R. B. Wright: Discrete Mathematics and Its Applications, 6th Edition, Mc Graw Hill (2007).

II. Course content & schedule:

1. **Logic and Proofs, Sets**
2. **Introduction and Recursion**
3. **Counting**
4. **Advanced Counting Techniques**
5. **Relations**
6. **Boolean Algebra**

III. Reference Literature:

[1]. Kenneth A. Rosen, Charles R. B. Wright: Toán rời rạc ứng dụng trong tin học (bản dịch bởi Phạm Văn Thiều và Đặng Hữu Thịnh), NXB Nhà xuất bản Giáo dục, Hà Nội (2007).

[2]. Susanna S. Epp: Discrete Mathematics with Applications, 4th Edition, Brooks/Cole Publishing Company (2010).

[3]. Bender, E. A., Williamson, S. G.: A Short Course in Discrete Mathematics, ISBN 0-486-43946-1, Dover (2005).

MATH 2.4: NUMERICAL METHODS

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	18	8	8	**34**

3. **Prerequisites**: Linear Algebra, Calculus 1, Calculus 2, Basic Programming
4. **Recommended background knowledge**:
5. **Subject description:**

   This course is an introduction to the numerical analysis. The primary objective of the course is to develop the basic understanding of numerical algorithms and skills to implement algorithms to solve mathematical problems on the computer. During the course, students will have a chance to learn basic Matlab which is widely used in many scientific fields.

6. **Objectives & Outcome:**

   As a result of successfully completing this course, students are able to:

- Understand the gap between theory and practice in computer science.
- Know some algorithms in Numerical methods.
- Have abilities to deploy or develop a Numerical method.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	20	0	0	0	0	80

8. **Prescribed Textbook(s):**

II. Course content & schedule:

1. **Sets Introduction, Errors in Computation**
2. **Interpolation and Approximation**
3. **Numerical solution of Non-linear Equations**
4. **Solution of system of linear algebraic equations**
5. **Numerical integration and differentiation**
6. **Solution of ordinary differential equations (ODE)**
7. **Numerical solution of Partial differential equations (PDE)**

III. Reference Literature:

[1]. Atkinson, Kelldan E.: An introduction to Numerical Analysis, 2rd Edition, Wiley, ISBN-10: 0471624896; ISBN-13: 978-0471624899 (1989)

[2]. Hildebrand, F. B.: Introduction to Numerical Analysis, 2nd Edition McGraw-Hill, ISBN 0-070-28761-9, (1974).

[3]. Leader, Jeffery J.: Numerical Analysis and Scientific Computation, Addison Wesley, ISBN 0-201-73499-0 (2004).

[4]. Robert J. Schilling, Sandra L. Harries: Applied Numerical Methods for Engineers using MATLAB and C, 3rd edition Thomson Brooks/Cole, ISBN-10: 0534370144; ISBN-13: 9780534370145 (2000).

ICT 2.5: BASIC DATABASES

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	24	15	0	**39**

3. **Prerequisites**:
4. **Recommended background knowledge**:
5. **Subject description:**

   A database is an organized collection of related data. In modern organizations databases play an essential part for a wide-range of operations. Some common operations involve personnel management, inventory control, customer tracking, and marketing. Companies also use databases for analyzing consumer demands, service quality, and customer profiles. The aim of this course is to introduce fundamental principles and user-centric methodologies for effective creation of a complete and fully functional database management system. This course focuses on applications of basic database theories and SQL practices in different database systems.

6. **Objectives & Outcome:**

   Upon completion of the course, the students are able to:

   - Understand basic database concepts and theories.
   - Understand basic techniques of relational database programming.
   - Apply relational modeling in database design

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	40	0	0	0	50

8. **Prescribed Textbook(s):**

   [1] Database Systems: The Complete Book (2nd Edition). Hector Garcia-Molina, Jeff Ullman, and Jennifer Widom. Prentice Hall, 2008

   [2] Database Systems: A Practical Approach to Design, Implementation, and Management (6th Edition). Thomas Connolly and Carolyn Begg. Addison-Wesley, 2014.

II. Course content & schedule:

1. **Introduction to databases**
2. **Relational Algebra**
3. **Basic SQL**
4. **More SQL**
5. **Constraints**
6. **Transactions, Views, Indexes**
7. **SQL Programming**
8. **Design Theory for Relational Databases**
9. **Entity-Relationship Model**
10. **10. Other High Level Design Languages**
11. **SQL Authorization**
12. **Group Project**
13. **Group Project Presentation**

III. Reference Literature:

[1]. Database Systems: Design, Implementation, and Management (11st Edition). Carlos Coronel,Steven Morris. Cengage Learning, 2014

[2]. Database System Concepts (6th Edition). Avi Silberschatz, Henry F. Korth, S. Sudarshan.McGraw-Hill, 2010

ICT 2.6: COMPUTER NETWORKS

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	30	0	10	**40**

3. **Prerequisites**: N/A
4. **Recommended background knowledge**: N/A
5. **Subject description:**

In this course, we will go through a number of fundamental concepts in the design and implementation of computer communication networks, their protocols, and applications. The students also will learn about the major software and hardware technologies used on home and enterprise computer networks as well as the global Internet. Although this course is not designed as a certification preparation course, all students will learn from materials/ simulations available through the Cisco Networking Academy program.

The following list includes the main topics covered in the course:

- Introduction to networking, networking fundamentals,
- Physical layers and networking media,
- Protocol principles, protocol verification, HDLC, PPP,
- Ethernet Technologies, 802.11, broadband wireless and switching,
- Routing protocols, congestion control, IP addressing, subnet and VLAN,
- UDP, TCP and network performance,
- Email SMTP, File transfer FTP, Network Management SNMP, Domain Name Management DNS,
- Ipsec, Access list control, Web security.

6. **Objectives & Outcome:**

At the end of the course, students should be able to:

- Understand the basic design, history and the evolution of the Internet and computer networks,
- Describe the key industry standards that define the Internet,
- Explain how digital messages are transported across physical network media through various wired/ wireless technologies and protocols,
- Understand the relationship between the Internet infrastructure, protocols and networked applications,
- Understand the fundamental characteristics of packet-switched data networks and the key protocols that make up the TCP/IP communications suite,
- Understand the key components and design principles associated with wide area networks as used by ISPs to deliver global network services,
- Configure the basic software and hardware required to operate/ support and troubleshoot network services,
- Get prepared for the advanced computer networks and network security courses.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	0	0	40	0	0	60

8. **Prescribed Textbook(s):**

[1] Andrews S. Tannebaum, *Computer Networks, 4th edition*, Prentice Hall, 2002.

[2] CCNA : Cisco Networking Academy Program

II. Course content & schedule:

1. **Introduction to Networking & Internet**
2. **Networking models**

3. **Networking fundamentals**
4. **Physical Layer**
5. **Link Layers and Local Area Networks**
6. **Network Layers and Routing**
7. **Transport Layers**
8. **Application Layers**
9. **Project Presentation**

III. Reference Literature:

[1]. Larry L. Peterson, *Computer Networks: A system approach*, Morgan Kaufmann, 2011.
[2]. J. Kurose and K. Ross, *Computer Networking: A Top-down approach*, Addison-Wesley, 2009.

ICT 2.7: SOFTWARE ENGINEERING

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	28	8	0	**36**

3. **Prerequisites**: Basic programming and Imperative programming
4. **Recommended background knowledge**: The C/C++ programming language, UML
5. **Subject description:**

This course will study a collection of methods which embody an engineering approach for software development. We will discuss the nature of software and software projects, software development models, software process maturity. We will study methods for analysis, design, testing, and implementation of large, complex software systems. We will inquire into the various perspectives on software quality - what it means, how to measure it, how to improve it. Moreover, though group projects, students can obtain hands-on experiences on entire phases and workflow of the software process.

6. **Objectives & Outcome:**

Upon successful completion of this course the student will be able to:

- Describe the history of the term "software engineering" and explain its current meaning and importance
- Explain well-known software development process models
- Select, with justification, a software development process which is most appropriate for the development and maintenance of a diverse range of software products
- Use a common, semi-formal method (for example, UML diagrams) to specify the requirements of a moderately sized software product
- Conduct software design using an accepted program design methodology such as UML
- Distinguish between different types and levels of testing (for instance, unit, integration, systems, and acceptance) for medium-size software products
- Discuss various testing techniques such as white box and black box testing
- Discuss key principles and common methods for software project management such as scheduling, size estimation, cost estimation and risk analysis
- Get familiar with CASE tools and/or environments including UML drawing tools and IDEs

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	30	0	20	40

8. **Prescribed Textbook(s):**

[1]. *Software Engineering*, Ian Sommerville, 8th Edition, Addison Wesley Press, 2007

II. Course content & schedule:

1. **Course introduction – Background information – Introduction to software engineering**
2. **Socio-Technical Systems - Critical Systems**
3. **Software Processes**
4. **Software Requirements and Requirements Engineering Processes**
5. **System Models**
6. **Critical Systems Specification**
7. **Design Principles and Architectural Design**
8. **User Interface Design**
9. **Implementation**
10. **Verification, Validation, and Testing**

11. Project Management

12. Final exam

III. Reference Literature:

[1]. *Software Engineering*, Ian Sommerville, 8th Edition, Addison Wesley Press, 2007

[2]. *Software Engineering: A Practitioner's Approach*, R.S. Pressman, McGraw-Hill, 2010

[3]. G. Booch, J. Rumbaugh, and I. Jacobson, The Unified Modeling Language User Guide, 2nd edition, Addison-Wesley, 2005

[4]. UML Tutorial: http://www.sparxsystems.com.au/uml-tutorial.html

[5]. StarUML – A software modeling tool: http://staruml.sourceforge.net/en/

ICT2.8: OPERATING SYSTEMS

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	25	6	9	**40**

3. **Prerequisites**:
   Computer Architecture
4. **Recommended background knowledge**:
   C programming language
5. **Subject description:**
   The goal of this course is to provide an introduction to the fundamental concept, the internal operation ofoperatingsystems(OS). The course will start with a brief introduction to the important roles, the evolution of operating systems, and then cover the major components of most operating systems.Particular emphasis will be given to three major OS subsystems: process management (processes, threads, CPU scheduling, synchronization, and deadlock), memory management (segmentation, paging, swapping), and file systems
6. **Objectives & Outcome**
   At the end of the course, students should be able to:
   - Know about the roles and the major components of Operating Systems.
   - Understandwell the concurrency, the scheduling algorithm in Operating Systems.
   - Understandmajor problems in Operating Systems such as deadlocks, synchronization,...andsolutions to these issues.
   - Explain and implement virtual memory.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	0	10	30	50

8. **Prescribed Textbook(s)**

[2]. Andrew Tanenbaum, *Modern Operating Systems*, Prentice Hall

[3]. William Stallings, Operating Systems, Prentice Hall

[4]. Harvey M. Deitel, An introduction to operating systems. Addison-Wesley

II. Course content & schedule:

1. **Introduction to Operating Systems**
2. **Operating-System Structures**
3. **Processes**
4. **Threads**
5. **CPU Scheduling**
6. **Process Synchronization**
7. **Deadlocks**
8. **Memory Management**
9. **Virtual Memory**

10. **File-System Interface**

III. Reference Literature:

ICT 2.9: DIGITAL SIGNAL PROCESSING

I. Reference Literature:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	25	10	10	**45**

3. **Prerequisites**: Mathematic, complex numbers, differential equations, classical matrix operations
4. **Recommended background knowledge**: N/A
5. **Subject description:**
   The aim of this course is to provide a prerequisite for many related multimedia courses. Examples include speech processing, image processing, pattern recognition, and so on.
6. **Objectives & Outcome:**

   When completing this course, it is expected that a student will have the ability to:

- Recognize the terminology that is used in the Digital Signal Processing (DSP) field.
- Apply transform techniques and concepts in formulating and solving engineering problems.
- Analyze basic DSP building blocks: digital filters, spectrum analyzers, sample rate converters (up-sampling and down-sampling), and the fast Fourier transform (FFT) algorithm.
- Design and synthesize these building blocks and use them effectively in applications.
- Think critically, ask questions, and apply problem solving techniques.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	10 (bonus)	10		30	50

8. **Prescribed Textbook(s):**

II. Course content & schedule:

1. **Signals and Frequency**
2. **Difference Equation, Discrete-Time Fourier Transform, Sampling and Quantization**
3. **Z-transform**
4. **Inverse z-transform**
5. **Discrete Fourier Transform**
6. **Signal analysis and synthesis using DFT**
7. **Fast Fourier Transform**
8. **Digital Filters**
9. **FIR and IIR filters**
10. **Interpolation and multi-rate systems, Digital Processors**
11. **Projects**

III. Reference Literature

[1] Digital Signal Processing [4th Edition] by John G. Proakis and Dimitris G. Manolakis

[2] Digital Signal Processing Using MATLAB [3rd Edition] by Vinay K. Ingle and John G. Proakis

[3] Discrete-Time Signal Processing [3rd Edition] by Alan V. Oppenheim and Ronald W. Schafer

[4] Signals and Systems [2nd Edition] by Alan V. Oppenheim and Alan S. Willsky

ICT 2.10: MOBILE WIRELESS COMMUNICATION

I. Reference Literature:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	24	12	0	**36**

3. **Prerequisites**: N/A
4. **Recommended background knowledge**: N/A
5. **Subject description:**

   The aim of this course is to provide students an overview of Wireless and Mobile Network system including wireless signal transmission protocol, networking, mobile network and IEEE wireless data network.

6. **Objectives & Outcome:**

   As a result of successfully completing this course, students are able to:

- Know the basic principles of wireless and mobile network.
- Understand basic wireless and mobile network protocol, mobile wireless management.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	40	0	0	0	50

8. **Prescribed Textbook(s):**

[1]. Mischa Schwartz: Mobile Wireless Communication, CAMBRIDGE UNIVERSITY PRESS, 1st Edition (2005)

[2]. Wireless Communications: Principles and Practice (2nd Edition) by Theodore S. Rappaport

II. Course content & schedule:

1. **Introduction**
2. **Cellular concept and channel allocation**
3. **Multiple Access techniques**
4. **Second Generation Mobile Network System**
5. **2.5/3G Mobile Network System**
6. **Wireless LANs and personal-area networks**

III. Reference Literature

ICT 2.11: INTRODUCTION TO EMBEDDED SYSTEMS WITH FPGA

I. Reference Literature:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	21	0	9	**30**

3. **Prerequisites**: Computer Architecture
4. **Recommended background knowledge**: Computer Architecture
5. **Subject description:**

   In this course, the fundamentals of embedded system hardware and firmware design will be explored. Issues such as embedded processor, hardware/firmware partitioning, circuit design, development tools, firmware architecture, will be discussed. The Field Programmable Gate Arrays (FPGA), a very high-speed logic device, will be selected to study. The architecture of FPGA and Verilog HDL language for logical synthesis will be discussed. The course will culminate with a significant small project which will extend the base of students in the course. Depending on the interests of the students, many topics may be developed.
6. **Objectives & Outcome:**

   The course has several goals. First, it will expose students to the field of embedded systems, and will provide a knowledge foundation which will enable students to pursue subsequent courses such as real-time embedded systems software, computer design, etc. Students will become familiar with the associated technical vocabulary and will learn about potential career opportunities in the field of embedded system design. Second, students will have the opportunity to develop an embedded system and construct the hardware and firmware implementation. This will provide students with an opportunity to gain a thorough understanding of the phases of embedded system development and familiarity with hardware and software development. Third, students will be given the opportunity to be familiar with design skills in embedded system.
7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	0	20	30	40

8. **Prescribed Textbook(s):**

[1] Digital Electronics – Principles, Devices and Applications, Anil. K. Maini, Wiley, 2007.
[2] Digital Fundamentals, Thomas L. Floyd, 11th edition, Pearson Education 2015
[3] Advanced Digital Design with the Verilog HDL, Micheal D.Ciletti, Prentice Hall, 2005

II. Course content & schedule:

1. **Introduction**
   - Embedded System Architecture
   - Introduction to Embedded System Design
2. **Digital Electronics**
   - Number System
   - Digital Arithmetics
   - Logic Gates
   - Boolean Algebra and Simplification Techniques
   - Arithmetic Circuits
   - Multiplexer and Demultiplexer
   - Flip-Flop

- Counter and Register

3. **VeriLog HDL**
4. **Practice on FPGA**

III. Reference Literature

[1]. Fundamentals of Digital Logic with Verilog Design, Stephan Brown and Zvonko Vnaresic, 3rd edition, Mc-Graw Hill, 2013

[2]. Introduction to Embedded System Design Using Field Programmable Logic Arrays, Rahul Dubey, Springer, 2009

II. THIRD YEAR PROGRAM

ICT 3.1: OBJECT ORIENTED SYSTEM ANALYSIS & DESIGN

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	24	0	12	**36**

3. **Prerequisites**: Object-oriented Programming
4. **Recommended background knowledge**: N/A
5. **Subject description:**

The Object-Oriented Analysis and Design (OOAD) course presents the concepts and techniques necessary to effectively use system requirements captured in use cases to drive the development of a robust design model. During this course, students learn to apply the Unified Modeling Language (UML), a de facto industry standard, to fundamental object-oriented analysis and design concepts including architecture, objects, classes, components, stereotypes, relationships, and all supporting diagrams. An emphasis is on group projects in which 3-5 students per a group will work on a medium-scale software system.

6. **Objectives & Outcome:**

This course focuses on:

- A basic explanation of Object-Oriented Analysis and Design (OOAD) concepts
- A hands-on practical experience in applying the techniques for a high-quality design

Upon completion of the course, the students are able to:

- Apply an iterative, use case-driven, architecture-centric process to the development of a robust design mode
- Apply Object-Oriented (OO) concepts: abstraction, encapsulation, inheritance, hierarchy, modularity, and polymorphism to the development of a robust design model
- Describe the different views of software architecture, keymechanisms that are defined in support of that architecture, and the effect of the architecture on the produced design
- Define basic design considerations, including the use of patterns

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	25		0	15	50

8. **Prescribed Textbook(s):**

[1].*Mastering Object-Oriented Analysis and Design with UML*. The IBM course DEV475, 2003.

II. Course content & schedule:

1. **Introduction:**

- Introduce the course
- Best Practices of Software Engineering

2. **Concepts of Object Orientation**
3. **Requirement Overview**
4. **Analysis and Design Overview**
5. **Architectural Analysis**

6. **Use Case Analysis**
7. **Identify Design Elements and Mechanisms**
8. **Describe the Run-time Architecture, Distribution**
9. **User Case Design**
10. **Subsystem Design**
11. **Class design**
12. **Database design**

III. Reference Literature:

[1] *The Unified Modeling Language Reference Manual (2nd edition)*. J. Rumbaugh, I. Jacobson, and G. Booch. Addison-Wesley Professional 1999

[2] *The Object Constraint Language: Getting Your Models Ready for MDA*. Jos B. Warmer and Anneke G. Kleppe. Addison-Wesley Professional, 2003.

ICT 3.2: WEB APPLICATION DEVELOPMENT

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	28	0	8	**36**

3. **Prerequisites**: Basic Programming, Introduction to Informatics
4. **Recommended background knowledge**: N/A
5. **Subject description:**

   Web-based applications offer a wide range of business advantages over traditional desktop applications such as accessible anywhere, easily customisable, accessible for a range of devices, etc. However, creating Web applications requires different approaches than traditional desktop applications and involves the integration of numerous technologies. This course will provide students will basic concepts and essential skills in web application development at both client-side and server-side. The topics covered include: HyperText Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript, jQuery, Bootstrap, Hypertext Preprocessor (PHP), PHP and MySQL, CodeIgniter, Representational State Transfer (REST), and Security.

6. **Objectives & Outcome:**

   At the end of the course, students should be able to:
   - Understand background in Web technologies
   - Understand principles in developing client and server sides of Web applications
   - Have skills to develop Web applications with HTML, CSS, Javascript, and PHP
   - Describe emerging trends in Web environments

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	40	0	0	50

8. **Prescribed Textbook(s):**

[1]. P. J. Deitel and H. M. Deitel; *Internet & World Wide Web: How to Program 4th Edition*, Prentice Hall, 2007, ISBN: 0131752421.

[2]. Sam Ruby, *Agile Web Development with Rails 4*, Pragmatic Bookshelf, 2013, ISBN: 978-1-93778-556-7

II. Course content & schedule:

1. **Introduction to World Wide Web**
2. **Hypertext Markup Language (HTML)**
3. **Cascading Style Sheets (CSS)**
4. **Javascript**
5. **jQuery**
6. **Bootstrap**
7. **Assignment 1 Presentation**
8. **PHP & MySQL**
9. **CodeIgniter**
10. **XML**
11. **REST**
12. **Web security**
13. **Trend in Web technologies**
14. **Assignment 2 Presentation**

III. Reference Literature:

[1]. P. J. Deitel and H. M. Deitel; *Internet & World Wide Web: How to Program 4th Edition*, Prentice Hall, 2007, ISBN: 0131752421.

[2]. *Drupal Documentation*, https://www.drupal.org/documentation
[3]. Sam Ruby, *Agile Web Development with Rails 4, Pragmatic Bookshelf*, 2013, ISBN: 978-1-93778-556-7.
[4]. Michal Zalewski, *The Tangled Web: A Guide to Securing Modern Web Applications, No Starch Press, ISBN:* 9781593273880

ICT 3.3: MACHINE LEARNING & DATA MINING

I. Course description:

1. **Credit points**: ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	22		8	**30**

3. **Prerequisites**: Algorithms and Data Structures, Numerical Methods
4. **Recommended background knowledge**: Numerical Methods, Probability and Statistics
5. **Subject description:**

   The aim of this course is to give student basic concepts of artificial intelligent and machine learning: scopes, techniques, problems in Machine Learning and Data mining

6. **Objectives & Outcome:**

   As a result of successfully completing this course, students are able to:

   - Know the basic principles of ML and DM
   - Understand popular technique used in ML and DM
   - Able to solve some simple learning problems

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	30	0	0	60

8. **Prescribed Textbook(s):**

   [1]. A Modern Approach (3rd Edition) by Stuart Russell
   [2]. Machine Learning by Thomas M. Mitchell

II. Course content & schedule:

1. **Introduction to Machine Learning and Data Mining**
2. **Linear Regression**
3. **Logistic Regression**
4. **Regularization**
5. **Principal Component Analysis**
6. **Artificial Neural Network**
7. **Bayes classification**
8. **Group project/Presentation**

III. Reference Literature:

ICT 3.4: INFORMATION SECURITY

I. Reference Literature:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	24	12	0	**36**

3. **Prerequisites**: Calculus, Computer Networks
4. **Recommended background knowledge**:
5. **Subject description:**

   Information security issues in four major themes:
   - Cryptography
   - Access Control
   - Protocols
   - Software
6. **Objectives & Outcome:**

   As a result of successfully completing this course, students are able to:
   - Classify threats to information
   - Explain how modern cryptography works
   - Describe authentication and authorization in access control
   - Describe popular security protocols
   - Describe software-related security issues and solutions (security flaws, malware, security in software, trusted OS)
7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	0	30	0	0	20	50

8. **Prescribed Textbook(s):**

   [1] Stamp, Mark. Information security: principles and practice. John Wiley & Sons, 2011.

   [2] Whitman, Michael E., and Herbert J. Mattord. Principles of information security. Cengage Learning, 2011.

   [3] M. Bishop, Computer Security: Art and Science, Addison Wesley, 2003.

II. Course content & schedule:

1. **Introduction to Information Security**
2. **Classic Cryptography**
3. **Symmetric key crypto**
4. **Public key crypto**
5. **Hash Functions ++**
6. **Authentication & Authorization**
7. **Authentication, Protocols, Real-World Protocols**
8. **Software Flaws and Malware, Insecurity in Software, OS and Security**

III. Reference Literature

ICT 3.5: GRAPH THEORY

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	31	5	0	**36**

3. **Prerequisites**: Data structures, Basic Programming, Algorithms and Data Structures
4. **Recommended background knowledge**: Not necessary but useful: algorithm design paradigms, object-oriented programming, discrete mathematics
5. **Subject description:**

   The course recaps on the variety of algorithm design paradigms, lays foundation on the theoretical aspects of graphs, discusses many realistic applications of graphs and related graph problems, covers the standard graph algorithms for finding shortest paths and producing minimum spanning trees. It also provides insights (possibly in form of assignments) into more advanced graph problems and algorithms, for instance, bi-connected components, graph isomorphism and auto-morphism, cliques, minimal and minimum vertex cover, Hamiltonian paths, Eulerian paths, etc. Working sessions are also provided during and after the lecture for better understanding and hands-on practice of the taught algorithms.

   .

6. **Objectives & Outcome:**

- Graph theory
  - Recall of relevant discrete mathematics
  - Graph definitions and representations
  - Vocabulary and notions around graphs
  - Basic graph problems and their real-world applications
- Core graph algorithms
  - Recap algorithm design paradigms
  - Graph traversal (BFS,DFS)
  - Shortest paths: Dijkstra, Bellman-Ford, Floyd-Roy-Warshall algorithm; Johnson's algorithm
  - Minimum spanning tree: Prim's algorithms; Kruskal'salgorithms; Boruvka's algorithm
  - Algorithms for Binary Trees; Huffman codes; Tree traversal; Binary Search Trees
  - Maximum Flow :Ford-Fulkerson algorithm; Edmonds and Karp's algorithm
  - Graph Coloring algorithm

- Advanced topics on graphs
  - Cliques
  - Independent sets
  - Vertex covers
  - Eulererian paths
  - Hamiltonian paths

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	5	15	0	10	20	50

8. **Prescribed Textbook(s):**

   [1].Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, *Introduction to Algorithms(2nd edition)*, The MIT Press, 2001.

[2]. David Joyner, Minh Van Nguyen, and Nathann Cohen, Algorithmic Graph Theory, 2010.
[3]. Anany Levitin, *Introduction to The Design & Analysis of Algorithms*, Pearson, 2011
[4]. Douglas B. West, *Introduction to Graph Theory*, Prentice Hall, 1996

II. Course content & schedule:

1. **Course introduction: Recapitulation of algorithm design paradigm**
2. **Graph notions and terminology; Graph traversal algorithms: BFS, DFS Algorithms**
3. **Extensions of the graph traversal algorithms**
4. **Shortest path problem Dijkstra's algorithm; Bellman-Ford's algorithm;**
5. **Shortest path algorithms: Floyd-Roy-Warshall algorithm; Johnson's algorithm**
6. **Minimum spanning trees: Prim's algorithm, Kruskal's algorithm, Boruvka's algorithm**
7. **Binary Trees; Huffman codes; Tree traversal**
8. **Binary Search Trees**
9. **Maximum Flow: Ford-Fulkerson's algorithm; Edmonds and Karp's algorithm**
10. **Graph coloring algorithms**
11. **Brief introduction to NP-complete; Advanced topics in graph theory; Eulerian circuits; Hamiltonian circuits**

III. Reference Literature:

[1]. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, *Introduction to Algorithms(2nd edition)*, The MIT Press, 2001
[2]. David Joyner, Minh Van Nguyen, and Nathann Cohen, *Algorithmic Graph Theory*, 2010.
[3]. Anany Levitin, *Introduction to The Design & Analysis of Algorithms*, Pearson, 2011.
[4]. Douglas B. West, *Introduction to Graph Theory*, Prentice Hall, 1996.

ICT 3.6: ADVANCED DATABASES

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	25	6	9	**40**

3. **Prerequisites**: Basic Database
4. **Recommended background knowledge**: Basic Database; Programming skills
5. **Subject description:**

The goal of the course is to introduce students to modern database and information systems as well as research issues in the field. Topics and systems covered may include object-relational, object-oriented, active, semi-structured and streaming databases. Also discussed will be recent advances in database systems such as advanced query processing, distributed databases, data warehousing, on-line analytical processing, distributed information integration, XML query engines, web and semi-structured data management. Students will have to solve some small written and programming assignments that will help them to understand and digest the covered material.

The following list includes the main topics covered in the course:

- Introduction to database management system (DBMS), Relational model,
- SQL, Advanced SQL,
- Entity-Relationship Model,
- Relational database design.
- Object-based databases,
- XML databases,
- Storage and file structure, Indexing and hashing,
- Query processing,
- Transaction, concurrency control,
- Data analysis and mining,
- Parallel databases
- Distributed databases

6. **Objectives & Outcome:**

At the end of the course, students should be able to:

- Know how to model data using the entity-relationship model
- Be able to model data using a relational model
- Be able to manipulate relational data using relational algebra and calculus
- Understand and be able to use the basic SQL constructs
- Be able to identify functional dependencies in relational databases
- Know several database design algorithms and be able to use them
- Know how to create and manage XML database
- Understand the relationships among disk organization, file structures, and hashing
- Know several indexing structures for files and be able to use them
- Know several algorithms for processing queries and be able to use them
- Understand the basic concepts behind transaction processing
- Know how to manipulate concurrency control
- Understand the concepts behind parallel and distributed databases.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	0	10	30	50

8. **Prescribed Textbook(s):**

[1] Abraham Silberschatz, Henry F. Korth and S. Sudarshan, *Database System Concepts, 6th edition*, McGraw-Hill, 2010
[2] Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw-Hill, 2003

II. Course content & schedule:

2. **Relational database**
3. **Formal Relational Query Languages, SQL**
4. **Database design**
5. **Object-based databases**
6. **XML database**
7. **Data storage and querying**
8. **Transaction management**
9. **Data analysis and mining**
10. **System architecture: parallel and distributed databases**

III. Reference Literature:

[1]. Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 2000.
[2]. YannisManolopoulos, YannisTheodoridis and Vassilis J. Tsotras, Advanced Database Indexing, Kluwer Academic Publishers, 1999

ICT 3.7: COMMUNICATION SYSTEMS

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	25	10	10	**45**

3. **Prerequisites**: Complex numbers, differential equations, classical matrix operations, Probability & Statistics
4. **Recommended background knowledge**: N/A
5. **Subject description:**
   This course provides an introductory treatment of communications that covers both analogue and digital communication systems. Topics include analogue and digital modulations, effects of noise and channel impairments, design and analysis of simple communication systems, concepts of channel capacity and coding. Simple communication systems are illustrated throughout the subject. Several essential topics in electromagnetic wave propagation including wireless (unguided) propagation and optical signal (guided) propagation topics will be covered.
6. **Objectives & Outcome:**
   At the end of the course, the student should be able to:
   - Understand analogue and digital communication system, effects of noise and channel
   - Design and analysis simple communication systems, as well as channel capacity and coding
   - Differentiate modulation techniques (continuous-wave modulation, pulse modulation)
   - Have the general knowledge of wireless & optical communication network
   - Understand the fundamental mechanisms/phenomenon of propagation
   - Understand the role of propagation on the design of communication systems.
   - Think critically, ask questions, and apply problem-solving techniques.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	10 (bonus)	10	0	30	50

8. **Prescribed Textbook(s):**

II. Course content & schedule:

1. **Introduction to Communication Systems**
2. **Analogue Modulations**
3. **Analogue Modulations (cont.)**
4. **Analogue to digital**
5. **Digital Modulations**
6. **Baseband digital transmission**
7. **Passband digital transmission**
8. **Radio Wave Propagation Fundamentals**
9. **Wideband channel characterization & modeling**
10. **Propagation Channel Models**
11. **Research subject**

III. Reference Literature:

[1] Communication Systems [4th Edition] by Simon Haykin, John Wiley & Sons, Inc., New York, 2001
[2] Wireless Communications: Principles and Practice [2nd Edition] by Theodore S. Rappaport, Prentice Hall, 2001.
[3] Software Receiver Design: Build Your Own Digital Communication System in Five Easy Steps by Johnson Jr, C. Richard, William A. Sethares, and Andrew G. Klein, Cambridge University Press, 2011.

ICT 3.8: IMAGE PROCESSING

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	18	0	12	**30**

3. **Prerequisites**: Signals and Systems, Digital Signal Processing
4. **Recommended background knowledge**:
5. **Subject description:**

   Image processing is becoming a useful assisting tool for many fields of science such as computer science, electrical and electronic engineering, robotics, physics, chemistry, environmental science, biology, and medical science, etc. In this course, fundamental concepts and techniques in digital image processing are introduced to students. The topics covered include Sampling and Quantization, Image Representation, Color Representation, Image Transform, Image Operations, Image Filtering, Edge Detection and Mathematical Morphology. The concepts in these topics will be illustrated using Matlab.
6. **Objectives & Outcome:**

   At the end of the course, students will be able to:
   - Understand the basic principles and concepts of image processing
   - Practice how to use these concepts and principles with Matlab

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	30	0	0	60

8. **Prescribed Textbook(s):**

[1] Solomon, Chris, and Toby Breckon. Fundamentals of Digital Image Processing: A practical approach with examples in Matlab. John Wiley & Sons, 2011.

[2] Petrou, Maria, and Costas Petrou. Image processing: the fundamentals. John Wiley & Sons, 2010.

II. Course content & schedule:

1. **Course Introduction**
2. **Introduction to Matlab for Image Processing**
3. **Color Representation**
4. **Image transform**
5. **Assignment Presentation 1**
6. **Point Processing**
7. **Spatial filtering**
8. **Edge detection**
9. **Mathematical morphology**
10. **Assignment Presentation 2**

III. Reference Literature:

ICT 3.9: COMPUTER GRAPHICS

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	23	7	0	**30**

3. **Prerequisites**: Linear Algebra, Basic Programming in C++
4. **Recommended background knowledge**: Students are expected to be fluent in C/C++ and familiar with modern development tools such as Visual Studio, XCode, or the GNU toolchain. Furthermore, experience in image processing is also useful for this subject.
5. **Subject description:** This course will provide students with a thorough grounding in the fundamentals of computer graphics. During the course, broad issues related to the displaying of objects will be presented, including: color modes, homogeneous coordinates, curves and surfaces, geometrical transformation, projections, visible-surface determination, illumination and shading, ray tracing, rendering. This course further allows students to develop programming skills in computer graphics by practical sessions.
6. **Objectives & Outcome:**

Upon successful completion of the course students will be able to:

- Appreciate mathematical fundamentals of computer graphics techniques.
- Understand computer graphics concepts
- Gain experience in Open GL by completing programming projects
- Work within a group to complete a graphics project

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	50	0	0	40

8. **Prescribed Textbook(s):**

[1] Hughes, J.F. and van Dam, A. and Foley, J.D. and Feiner, S.K. *Computer Graphics: Principles and Practice (3rd Edition)*, Addison-Wesley, 2014, ISBN: 9780321399526.

II. Course content & schedule:

1. **Introduction to Computer Graphics**
2. **Light and color**
3. **Shapes**
4. **Parametric curves and surfaces**
5. **Transformation, camera models**
6. **Viewing, Rendering**
7. **OpenGL, shader**
8. **Texture mapping, Ray tracing**
9. **Global illumination**
10. **Project presentation**

III. Reference Literature:

[1]. Donald D. Hearn and M. Pauline Baker. Computer Graphics with OpenGL (4th Edition). 2010. Prentice Hall Professional Technical Reference.

ICT 3.10: MOBILE APPLICATION DEVELOPMENT

IV. Course description:

9. Credit points: 3 ECTS

10. Time commitment:

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	23	7	0	**30**

11. Prerequisites: Object-oriented programming

12. Recommended background knowledge: N/A

13. Subject description: Mobile application development has been becoming a multi-billion-dollar industry where ICT engineers can build their own career and fortune. This course provides the basic knowledge and necessary skills to develop app for Android, one of the three major mobile platforms. The course will cover core concepts such as UI design, asynchronous programming, to platform specific features including background transfer or notification. After the course, student will be able to develop apps based on their own idea.

14. Objectives & Outcome:

By completing this course, students will be able to:

- Understand key features of mobile devices, mobile application, and mobile platforms.
- Develop app for the Android platform, including core concepts and OS' specific services
- Analyze and improve the performance of app to maximize overall user experience.
- Improve team work and communication skill to work on the course project.

15. Assessment/ Evaluation

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	0	0	30	60

16. Prescribed Textbook(s):

[1]. Dawn Griffiths and David Griffiths, Head First Android Development, Oreilly, 2015.

[2]. Android Programming: The Big Nerd Ranch Guide, 2nd edition, by Bill Phillips and Brian Hardy, 2015

V. Course content & schedule:

11. OS Review & Android Introduction

- Concepts of OS
- Introduction and History of Android

12. Android Fundamental

- Architecture
- Application, Activity, Fragments
- View, Context

13. Resources

- Layout, values, 9-patches images

14. Mid-term Project Presentation

15. Background Tasks & Services

- Threads, A sync Tasks
- Services

16. Networking

- Embedded library: Http Url Connection
- Framework: Volley
- Data representation: JSON/XML

17. Final Project Presentation

VI. Reference Literature:

[1]. Dawn Griffiths and David Griffiths, Head First Android Development, Oreilly, 2015.

[2]. Android Programming: The Big Nerd Ranch Guide, 2nd edition, by Bill Phillips and Brian Hardy, 2015

ICT 3.11: COMPUTER GRAPHICS AND HUMAN COMPUTER INTERACTION

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	23	0	7	**30**

3. **Prerequisites**: N/A
4. **Recommended background knowledge**: N/A
5. **Subject description:**

This subject examines the design, evaluation and implementation of interactive computing systems for human use (HCI) and the major phenomena surrounding them. Also considered are joint performance of tasks by humans and machines, structure of human machine communication, social and organizational interactions with machine design, human capabilities to use machines including their learn ability, engineering concerns that arise in designing interfaces, the process of specification design, implementation and evaluation of interfaces and design tradeoffs.

The following list includes the main topics covered in the course:

- Introduction to HCI
- Frameworks for Cognition and Theories
- Usability
- Graphics and Sound
- Design Methods and Process
- Usability testing
- CSCW
- Mobile and Ubiquitous Interaction

6. **Objectives & Outcome:**

At the end of the course, students should be able to:

- Understand the basic design, history and the evolution of HCI,
- Explain the difference between good and bad design,
- Know how to take into account user's needs in interaction design,
- Describe what is involved in the process of interaction design;
- Apply a number of interaction design techniques;
- Analyze and critique interfaces;
- Know how to evaluate the usability and effectiveness of various products — e.g., web sites, PDAs, game consoles, etc.
- Understand the role of theory and frameworks in HCIcourses.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	40	0	0	0	50

8. **Prescribed Textbook(s):**

[1] Alan Dx, Janet Finley, Gregory D. Abowd, Russell Beale (2004) Human-Computer Interaction, Pearson Education Limited

[2] Ubiquitous Computing: Smart Devices, Environments and Interactions (2009) rof. Stefan Poslad, ISBN 0-470-03560-9.

II. Course content & schedule:

1. **Introduction to HCI; Human and Computer; Interaction Styles**
2. **Cognitive Models and Theories:**

- Information Processing including memory, perception, attention
- Cognitive modeling
- External Cognition
- Distributed Cognition

- Mental Models
- Conceptual model
- Metaphors

3. **Usability**: Goals and Principles
4. **Usability Methods**:

- Task analysis
- Cognitive walkthrough
- User Interface Models
- Prototyping
- Card Sorting
- Focus group

5. **Graphics and Sound**
6. **Design Methods and Process 1**

- Ethnographic studies
- Requirement through Persona & Scenario

7. **Design Methods and Process II**

- User Centered Design
- Participatory Design
- Predictive Evaluation
- Heuristic Evaluation

8. **Usability Testing**
9. **CSCW**
10. **Mobile and Ubiquitous Computing**

III. Reference Literature:

[1]. Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S. & Carey, T. (1994) Human-Computer Interaction: Concepts And Design, Addison Wesley, ISBN 0-201-62769-8

[2]. Preece, J., Sharp, H. & Rogers, Y. (2002) Interaction design, Wiley, ISBN 0-471-49278-7

[3]. Alan Dx, Janet Finley, Gregory D. Abowd, Russell Beale (2004) Human-Computer Interaction, Pearson Education Limited

[4]. Ubiquitous Computing: Smart Devices, Environments and Interactions (2009) rof. Stefan Poslad, ISBN 0-470-03560-9

ICT 3.12: MACHINE LEARNING AND DATA MINING II

I. Course description:

1. **Credit points**: 4 ECTS
2. **Time commitment:**

Items	Lecture	Project	Practical	**Total**
No. of hours	20	8	12	**40**

3. **Prerequisites**: Artificial Intelligence & Machine Learning, Algorithms and Data Structures, Basic Programming, Numerical Methods
4. **Recommended background knowledge**:

- Artificial Intelligence & Machine Learning
- Numerical Methods
- Familiar with Matlab or R

5. **Subject description:**

The course introduces the essential basics and concepts in Data Mining. Topics concern mainly the learning problems in clustering and classification and the fundamental methods.

6. **Objectives & Outcome:**

- have general knowledge in data mining
- understand real-world problems and able to propose solutions to these problems
- solve data mining problems using known techniques
- learn to work in team and self-study/research

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	0	30	60	0	0

8. **Prescribed Textbook(s):**

[1] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2009). "The Elements of Statistical Learning: Data Mining, Inference, and Prediction".

[2] C.M. Bishop, "Pattern Recognition and Machine Learning (2006) ", Springer

II. Course content & schedule:

1. **Introduction to Data Mining**
2. **Data Pre-processing& Statistic Tools**
3. **Unsupervised learning – Cluster analysis**
4. **Supervised learning - Classification**
5. **Ensemble Learning**
6. **Project**

III. Reference Literature:

[1] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2009). "The Elements of Statistical Learning: Data Mining, Inference, and Prediction".

[2]. C.M. Bishop, "Pattern Recognition and Machine Learning (2006) ", Springer

ICT 3.13: APPLIED SCIENCE WITH PYTHON

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Project	Practical	**Total**
No. of hours	20	0	10	**30**

3. **Prerequisites**: Basic Programming, Numerical Methods, Algorithms and Data Structures, Machine Learning and Data Mining
4. **Recommended background knowledge**:
5. **Subject description:**

   This course introduces the field of data science through Python language. It shows how to conduct data science by learning how to analyze data. That includes knowing how to import data, explore it, analyze it, learn from it, and visualize it. The course also introduces two powerful areas of data analysis: machine learning and natural language processing. To conduct data analysis, a collection of powerful, open-source, tools is used, including: python, jupyter notebooks, pandas, numpy, matplotlib, scikit learn, nltk, and many other tools.
6. **Objectives & Outcome:**

   As a result of successfully completing this course, students can know:
   - Basic process of data science
   - Python and Jupyter notebooks
   - An applied understanding of how to manipulate and analyze datasets
   - Basic statistical analysis and machine learning methods
   - How to effectively visualize results

   By the end of the course, students should be able to find a dataset, formulate a research question, use the tools and techniques of this course to explore the answer to that question, and share their findings.
7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	10	20	20	0	0	50

8. **Prescribed Textbook(s):**

II. Course content & schedule:

1. **Introduction**
2. **Background of Python**
3. **Jupyter and Numpy**
4. **Pandas**
5. **Visualization**
6. **Machine Learning**
7. **Working with Text and Databases**
8. **Mini Projects**

III. Reference Literature:

[1]. Ilkay Altintas, Leo Porter. *Python for Data Science* (DSE200x), EdX / UC San Diego MicroMasters, https://courses.edx.org/courses/course-v1:UCSanDiegoX+DSE200x+2T2017/info

[2]. Christopher Brooks. *Introduction to Data Science in Python*, University of Michigan, https://www.coursera.org/learn/python-data-analysis

[3]. *Applied Data Science with Python Specialization*, University of Michigan, https://www.coursera.org/specializations/data-science-python

[4]. *Introduction to Python for Data Science* (DAT208x), EdX / Microsoft, https://courses.edx.org/courses/course-_v1:Microsoft+DAT208x+3T2017/info

ICT 3.14: SIMULATION OF COMMUNICATION SYSTEMS

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Project	Practical	**Total**
No. of hours	20	8	12	**40**

3. **Prerequisites**: Communication Systems, Numerical Methods
4. **Recommended background knowledge**:

- Knowledge background of Communication Systems
- Access to Matlab software

5. **Subject description:**
   - Introduction to Matlab
   - Simulation of basic components of a digital communication system
   - Evaluate performance of a digital communication systems (BER) in provided conditions
6. **Objectives & Outcome:**

After attending in this course, students are able to:

- Be familiar with Matlab
- Describe basic components of a digital communication system and their functions
- Use Matlab to illustrate input and output of signals of each component
- Evaluate performance of a system under provided circumstance (used technique, channel)

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Reports	Midterm	Final
Percentage %	0	0	30	0	20	50

8. **Prescribed Textbook(s):**

[1] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2009). "The Elements of Statistical Learning: Data Mining, Inference, and Prediction".

[2] C.M. Bishop, "Pattern Recognition and Machine Learning (2006) ", Springer

II. Course content & schedule:

1. **Introduction**
2. **Channel: Probability, Random Process, Fading**
3. **Inter Symbol Interference (ISI) and filtering**
4. **Digital Modulations**
   - Modulation / Demodulation techniques: BPSK, Q-PSK, M-PSK
   - BER vs E0/N over : AWGN, Rayleigh channel
5. **Orthogonal Frequency Division Multiplexing (OFDM)**
6. **Channel coding**
   - Channel coding techniques
   - Simulation of channel codes
   - Spread spectrum techniques
7. **Project discussions**
8. **Project presentation**

III. Reference Literature:

[1]. Viswanathan, Mathuranathan. "Simulation of digital communication systems using Matlab." Mathuranathan Viswanathan at Smashwords (2013).

[2]. Johm G.Proakis and Masoud Salehi, "Contemporary Communication System using Matlab", Northeastern University.

[3]. John G. Proakis and Masoud Salehi. 1994. Communication Systems Engineering. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
[4]. Tran Xuan Nam: “Simulation of radio communication system using Matlab”, Le Quy Don University.

ICT 3.15: DISTRIBUTED SYSTEMS

I. Course description:

1. **Credit points**: 3 ECTS
2. **Time commitment:**

Items	Lecture	Tutorial	Practical	**Total**
No. of hours	20	10	0	**30**

3. **Prerequisites**: Operating Systems, Computer Networks
4. **Recommended background knowledge**:
5. **Subject description:**

   Distributed systems take many forms, and many of the best known and most widely used computer systems are as prevalent as they are because of their distributed nature. This course unit introduces some of the essential concepts underlying distributed systems. Students are guided into an exploration of exemplar distributed applications (such as the web, email, file sharing and multi-user gaming, etc) and, through these exemplars, are introduced to the basic concepts that underpin modern distributed computing. This course also explains how distributed systems offer transparencies of various kinds and how they must contend with a range of complex issues to achieve that. It answers to other kinds of questions such as how massive distribution enables high-performance computing, how service abstractions in the web enable business-to-business integration and how the web of hyperlinked documents is changing into a richer web of data.
6. **Objectives & Outcome:**

   By completing this course, students will be able to:

- Understand the contrasting features between the distributed view of computing with the centralized one.
- Understand in detail how a few exemplar distributed applications work and what requirements they aim to satisfy.
- Understand in detail how a few exemplar distributed applications work and what issues and challenges they must contend with.
- Acquire practical skills in analyzing distributed applications.
- Acquire practical skills in discovering, describing and classifying some of the fundamental concepts in distributed systems.

7. **Assessment/ Evaluation**

Component	Attendance	Exercises	Assignments	Project	Midterm	Final
Percentage %	10	0	0	3	0	60

8. **Prescribed Textbook(s):**

[1]. Tanenbaum et al., Distributed systems: principles and paradigms, 2nd Edition, Pearson.

II. Course content & schedule:

1. **Introduction to Distributed Systems**
2. **Case Study: Internet, Web, and Email**
3. **Centralized to Threaded to Parallel to Distributed Systems**
4. **Architectural Paradigms for Distributed Systems**
5. **Inter-process Communication**
6. **Socket-level Client/Server Systems**
7. **Case Study: File Sharing**
8. **Case Study: Multi-User Gaming**
9. **High Performance Computing Systems**

III. Reference Literature: