cphbusiness
COPENHAGEN BUSINESS ACADEMY

AP Degree in Computer Science

2007

AP graduate in Computer Science.

Copenhagen Business Academy,
Copenhagen North
Trongårdsvej 44
2800 Kgs. Lyngby
www.cphbusiness.dk
www.knord.dk

Table of Contents:

Introduction

This document describes the Computer Science Programme at Copenhagen Business Academy, Copenhagen North. The present curriculum has been prepared on the basis of the curriculum requirements for the Higher Professional Education Programme for Information Technology and covers the implementation of these requirements. The requirements and the curriculum itself comply with the Danish Executive Order for higher professional educational programmes: AP in Computer Science.

Formal Requirements of the Programme

The AP programme aims at training the students to independently analyse, plan and implement solutions related to new development, further development and integration of IT systems in private and public companies, and organisations.

The graduate must be able to:
1 combine basic company awareness and knowledge about technological concepts with a thorough knowledge of and expert skills in programming and in systems development
2 apply different principles and methods of planning, managing and implementing the development process from conception to implementation, maintenance and operation
3 participate in managerial and cooperative functions together with other staff members having a different educational, linguistic and cultural background.

The Computer Science programme comprises the following subject areas:

Programming 40 ECTS Systems Development 30 ECTS Technology 20 ECTS Analysing the company situation 10 ECTS Specialisation Module 20 ECTS Final Examination Project 15 ECTS

The 4 first mentioned subject areas focus on all essential skills of a graduate. The core subject areas are Systems Development and Programming; whereas the other 2 subject areas support and focus on the context which contains Systems Development and Programming.

The Underlying Educational Principles of the Programme
The principles cover
- Thematic disciplines
- Interdisciplinary approach
- Problem and project orientation

Thematic Disciplines
- because we use it to create a focus, coherence and delineation of the separate modules and
- each module has its own theme

Interdisciplinary Approach
- because the graduate will not only be able to handle one or several subject areas but must also be able to use them in the relevant context
- as we do not have subjects in the separate modules

Problem and Project Orientation
- because the graduate will experience the project as the typical way of working and because solving new problems is a crucial skill
- in almost all modules we work with projects and they play an essential part of the evaluation of the student

Structure of the Programme

Module 1 – The Development of Object-oriented Systems

The working method of this module is a combination of lecturing, exercises and individual (compulsories) assignments. The study activities are planned in such a way that the individual student acquires basic skills primarily in Systems Development and Programming to be used in the project in module 2. The student works with problems connected to developing an application in an object-oriented programming language.

Module 2 – The Development of Multi-user Database Systems in an Organisational Context

The module is project-oriented. The project method is introduced and taught. The module takes the development of a database application for a company as its starting point with a focus on multiple user relational database systems and multiple user operating systems.

Module 3 – The Development of Distributed Systems

In this module the project work takes its starting point in groups of students cooperating in developing a distributed application. The focus is on the development of distributed systems with the help of Web technology.

Module 4 – Systems Development Methods in Perspective

In this module the student continues working on his/her project from module 3.
The focus is on reflection and evaluation of that project (primarily the process). This reflection is based on the introduction to other systems development methods rather than the ones used in the previous modules.

Module 5 – Elective Subjects

This module is designed for the individual part of the Programme and covers 2 elective subjects.

Module 6 – The Final Examination Project

In this module the student writes his/her Final Examination Report, preferably in cooperation with a company. Through the Final Examination Report the student has an opportunity to specialise within a certain subject area.

The Contents of the Following Pages

The individual modules are described in more detail with regard to purpose, objectives and sub-objectives. In addition, the extent of compulsories, project work and individualisation is defined. Each module is expressed in ECTS points: the student's workload. (ECTS: The European Credit Transfer System). In each module there is a reference to topics that are part of the Danish Executive Order. We use a specific terminology for this purpose, for example: P.Algorithms.2. This means that sub topic no 2 of the topic Algorithms is a part of the subject area Programming. If a subject area is specified without any subsequent numbers e.g. P. Algorithms –, it means that all topics of the subject area are included.

You will find a detailed description of the tests under each module, except for the elective subjects. The type of test that is required in elective subjects is specified under each subject.

The Academy can give permission to deviate from the Curriculum, provided the student has a special reason.

Module 1

(30 ECTS Points)

Theme

The Development of Object-orientated Systems

Aim

The aim of the module is that the student must master the basic principles of object-oriented programming and systems development, understand the interrelation between systems development methods and programming, and be able to develop parts of applications in an object-orientated programming language.

Objectives

The objectives are that the student:

- can implement parts of systems in an object-oriented programming language by applying an object-oriented systems development method
- can document systems development processes
- understands how programmes operate on a digital machine
- understands basic concepts of object-oriented programming and object- oriented programming languages
- understands basic concepts of systems development, including experiments
- can use basic principles of user interface design
- can use computer tools for modelling and implementing systems
- can assure the quality of systems
- can understand companies' organisation and management with a view to purchasing and using IT systems
- knows the present technological developments.

Sub objectives

The student must have an understanding of the main concepts of programming languages, including semantics (the meaning of words) and syntax (the structure of the language) in order to comprehend programming and the programming language of the 1st module.

The student must have an understanding of the main concepts of programming, including data types, abstract data types, data structures, algorithms, and the structure of a programme with the help of sequence, selection and iteration.

The student must understand general object-oriented principles and concepts, including classes, objects, inheritance, polymorphic systems and data protection.

The student must comprehend systems development as an interdisciplinary, complex and innovative process. In addition, the student must be introduced to main systems development concepts such as analysis, design, implementation and testing.
The student must understand the role and application of models in systems development, and the relevance of experiments as an element of systems development.

The student will be introduced to the principles and general characteristics of a relevant object-oriented method. The student must master the elements of the method that are relevant to implementing a system[1] in an object-oriented programming language. The student must be able to design relevant models (including source code) for an operational application. The student must understand the entire process, including the interrelationship between the models.

The student must master the basic concepts of object-oriented programming such as the implementation of static and dynamic data structures including lists, stacks and queues. The student must implement simple non-recursive algorithms on the above-mentioned data structures, including searching and sorting.

The student must be introduced to the pattern concept in connection with modelling and problem solving, and be able to apply basic object-oriented patterns.

The student must be able to create graphic interfaces by means of the libraries of the programming languages. (S)he should also understand the main patterns used in the programme libraries.

The student should become acquainted with the principles of designing user interfaces to be able to evaluate them. In addition, the student should be acquainted with standard components to structure user interfaces.

In support of the application of the object-oriented method and the implementation of graphic user interfaces, the student must be able to use computer-based tools.

The student must be introduced to the concept of quality and must be able to apply techniques for assuring the quality of documentation and programmes (e.g. tests). By the quality of documentation, we primarily mean the quality of the designed models according to the method and their traceability.

The student must have a general understanding of the organisations' structures and how they operate. The student must be able to understand the difference between designing special systems and purchasing standard software from an organisational point of view, among others, knowledge about the different approaches to implementing IT systems.

The student must know about the characteristics of the standard software systems and adaptation possibilities, as well as, the ways such systems can be used for registering operating information. To support programming and programming languages the student must understand how source codes in the programming language in Module 1 are translated by means of a hierarchy of virtual machines. The student must also understand the concepts of interpretation and translation, and how the source codes are translated on a digital machine on the basis of a concrete hierarchy of virtual machines. To support programming and programming languages the student must know about the organisation of internal and external storage facilities and understand their interaction. The student must be able to use the external storage functions to handle persistence by means of the programming language of Module 1.

[1] By "system" we mean a unique application, i.e. a non-distributed application without several users at a given time.

Compulsory Tests/Assignments
A total of: 10

Individualisation
None

Subjects from the Danish Executive Order
See Appendix 1

Module 2
(30 ECTS Points)

Theme
The development of multi-user database systems in an organisational context

Aim
The student should be well acquainted with multi-user relational database systems and multi-user operating systems. Furthermore, s/he should be able to develop database applications for organisational purposes.

Objectives
The objectives are that the student
- understands the essence of the project work, participates in it and reflects on the roles of the project work
- can implement a system in a multi-user database system by using an object-oriented systems development method
- can implement advanced data structures and algorithms in an object-oriented programming language
- understands how logical models are represented physically in the database system
- understands the structure of one multi-user operating system
- understands the problems inherent in multi-user operating and multi-database systems
- can carry out the development of a system in an organisational context
- can understand and analyse organisations to describe systems that support company strategies and IT strategies
- can understand IT-based business concepts and company strategy to formulate an IT strategy.

Sub objectives
The student carries out the project work within a predetermined frame. The student documents his/her ability to cooperate, to understand, and to follow a plan. It must be ensured that the student understands the difference between assignments and project work. It must also ensure that the student understands his/her own roles as well as the roles of the others through specific experiences with project work. In addition, the student must be able to evaluate the suitability of the distribution of roles.

The student must be able to write a report which documents and presents the completed systems development process.

Based on a modelling language the student must be able to design models of a database system including carrying out mapping - from class model to relational model to implementation model.

The student must be acquainted with relational algebra as a prerequisite for using programming languages to create and manipulate relational models.

The student must be able to assess the quality of a specific relational model including the integrity concept and normal forms with the help of the theory of relational models.

The student should know the central principles of the physical structure of the database. In addition, the student should know the difference between an implementation model and the physical structure of the database.

The student must understand the problems specifically connected to multi-user systems, including resource sharing and security.

The student must understand the structure of one concrete operating system focusing on its handling of processes and their synchronisation, storage management and virtual storage systems as well as file systems.

The student must master a declaratory programming language to establish and manipulate databases.

The student should be able to implement advanced dynamic data structures, among others, trees in an object-oriented programming language. S/he should also be able to implement advanced recursive algorithms including searching, sorting and traversing. In addition, the student should know the qualitative and quantitative characteristics of the data structures and algorithms.

The student must be able to implement a multi-user database system by integrating the use of one declarative language and one imperative language, and to assess the quality and effectiveness of the developed programme. In addition, the student must be able to apply relevant patterns related to the implementation of multi-user database systems.

The student must understand that parts of a method are technology-independent while other parts are connected with the applied technology and dependent on it.

The student must be introduced to using the method, previously presented (in Module 1) to implement a multi-user application. In connection with a defined project the student must be able to apply the method to implement an operative database application

The student must understand how important it is to involve the users and know the various techniques and incorporate them. Furthermore, the student must know different kinds of user tests to assure the quality.

The student must understand and model the company’s processes and structure to give an account of the consequences of the changes caused by the introduction of newly developed applications.

The student must understand how the IT systems change the user organisation and the systems strategic consequences. The student must have a general knowledge of the technological development.

The student must understand and analyse how an organisation has chosen to handle IT security organizationally, and know about risk assessment and vulnerability assessment as elements of assessing an organisation's IT security.

Compulsory Tests/Assignments

Number: 3

Project work

Subject:	Systems development, database, programming and company modelling
Size:	A large company-related group project (12 ECTS)
Report:	Max. 60 pages (excl. list of references and appendices)

Individualisation

None

Examination

Test 1: Modules 1+2

Subject matter: Subject areas of the module and the project report

Type: A combination of the project in groups and an individual oral test

Remarks: The 1st part of the test: students are examined in groups. The group presents the project and demonstrates the product of the project. After that, the group and the examiner discuss the contents. The examination time is 30 minutes. The time is distributed evenly on the presentation and the discussion.

In the 2nd part of the test: each student is examined individually (30 minutes.). A question is drawn by lot. The question is based on the written guidelines to the project made by the Academy. The questions must broadly cover the subject areas of the modules. The student answers the examination question on the basis of the project report.

Evaluation: The evaluation is based on an overall impression of the project report, the presentation by the group, the discussion, and the individual examination. The result will be expressed in terms of the Danish 7-point marking scale.
Date: Immediately after the module has ended.

Re-The examination must be completed before the start of the next semester. examination date:

Rules for Preparation time None preparation:

Examination Duration of 1st part of the test: 30 minutes (the entire group) rules: examination: 2nd part of the test: 30 minutes (per student)

Resources: The project report, the examination question and appendices, if any, handed out by the examiner

Required material: The project report produced by the student

Subjects: See Appendix 1 from the Danish Executive Order

See Appendix 1

Module 3

(30 ECTS)

Theme

The development of Distributed Systems

Aim

The aim of this module is that the student must master principles of distributed systems. In addition, the student must develop distributed applications.

Objectives

The objectives are that the student

- can carry out project work in an organisational context co-operating with developmental groups
- can implement a distributed system by using an object-oriented method
- understands the principles of a given distributed technology
- can use experiments to explore advanced technologies
- understands basic concepts within distributed programming

Sub-objectives

The student must define the project and be able to cooperate with other developmental groups to develop one common application, including project management and contracts between developmental groups.

The student must be able to write a report that contains and presents the systems development carried out.

Based on a specifically applied communication mode, the student must understand the general principles of networks, including service and protocol concepts. In addition, the student must be able to estimate the advantages and disadvantages of various types of LAN and WAN and relate them to the needs of a specific user organisation.

The student should understand the central principles behind distributed systems with a central service.

The student must be able to understand different ways of handling the communication between client and server. Furthermore, the student must understand the principles of both client and server programming.

The student must be introduced to the method of implementing a distributed application, previously presented in the 1st and 2nd modules. Based on a specified project, the student must be able to design relevant models (including source codes) in order to make the distributed application operational.

The student must be able to apply patterns that are relevant to distributed systems.

Compulsory Tests/Assignments

Number: 3, conducted as experiments, and the project work of the module.

Project Work

Subject: All subjects of the module The project work takes its basis in subjects chosen by the students

Size: A large group project in co-operation with other groups (10 ECTS)

Individualisation

The students form groups. Each group focuses in depth on one programming environment and one distributed technology.

Examination

Test 4:	**Module 3**
Subject matter:	Subject areas of the Executive Order: Programming and Technology, and the project report
Type:	Oral test
Remarks:	The student draws by lot a main question related to the subject area "Programming" of the Executive Order and sub-question(s) of the Executive Order area "Technology". The questions consist of both theoretical and practical elements. All questions are based on the written project guidelines made by the Academy. The student has 80 minutes' preparation time. The examination itself lasts 40 minutes including evaluation of the student's performance.
Evaluation:	In the evaluation the main question is weighted 80%. The evaluation is expressed in terms of the Danish 7-point marking scale.
Date:	Immediately after the module has ended.
Re-examination date:	The examination must be completed before the beginning of the next semester.
Rules for Preparation:	Preparation time: 80 minutes
Rules for the examination:	Duration of the examination: 40 minutes Resources: All written material, including the project report, the examination questions and appendices, if any, handed out by the examiner. Required material produced by the student: The project report

Subjects from the Danish Executive Order

See Appendix 1

Module 4

(10 ECTS)

Theme

Systems development methods in perspective

Objectives

The objectives are that the student

- is familiar with methods, previously widely used and promising new ones
- knows about classifications of systems development methods
- can compare various systems development methods and paradigms and choose the one fit for a certain situation
- knows about different project models and can evaluate the applicability of the project models
- can evaluate the quality of a product and a process

Sub-objectives

The student must be able to consider the systems development methods taught in the programme as being part of different systems development paradigms[2]. (S)he must be able to evaluate the usability of paradigms and methods on the basis of a specific assignment.

Furthermore, the student continues working with his/her project from Module 3, but focuses on reflecting on the process and product, including method, project model and quality.

The student can evaluate the applicability of different project models and understands the relevance of adapting project models to specific situations.

The student must know the many-sidedness of the quality concept, and be able to evaluate the quality of his/her own work and other people's work.

Compulsory Tests/Assignments

Number: None

Project Work

Subject: Reflections on the module 3 project

Size: Individual or group work (3 ECTS)

Remark: This report is a supplement to the module 3 project report

Individualisation

None

[2] In the technical literature system development paradigms are also called project models, e.g. the waterfall model the parallel model, prototyping or the experimental development model

Examination

Test 3: Module 4

Subject matter:	Subject area of the Executive Order: Systems Development, the module 3 project report and the module 4 supplementary report. The student selects a maximum of 40 pages as his/her subject matter for the test.
Type:	A combination of the project and an oral examination
Remarks:	In the 1st part of the test: the students are examined in groups. The group presents the project and demonstrates the product of the project. After that the group and the examiner discuss the contents. The examination time is 30 minutes. The time is distributed evenly over the presentation and the discussion. In the 2nd part of the test: each student is examined individually (30 minutes). A question is drawn by lots. The question is based on the written guidelines for the written project made by the Academy. The student answers the examination question on the basis of the project report.
Evaluation:	The module 3 project report, the module 4 supplementary report and the 40 pages specified as evaluation material must be submitted at the end of the module. The student is evaluated on the basis of an overall impression of the project report, the presentation by the group, the related discussion and the individual examination. The result will be expressed in terms of the Danish 7-point marking scale.
Date:	Immediately after the module has ended.
Re-examination date:	The examination must be completed before the beginning of the new semester.

Topics: from the Danish Executive Order

See Appendix 1

Module 5

(20 ECTS) 2 Elective Subjects

Theme
Depending on the 2 elective subjects

Purpose
Depending on the 2 elective subjects

Objectives
Depending on the 2 elective subjects

Sub-objectives
Depending on the 2 elective subjects

Examination

Test 4: Elective subjects

The examination is defined from year to year. There is one examination for the 2 elective subjects together. The student's performance is evaluated according to the Danish 7-point marking scale.

Module 6

(15 ECTS)

Final Examination Report

Focal Points of the Semester:

In his/her final report, the student must document his/her ability to systematically and analytically implement a solution to a problem. The student must also demonstrate his/her ability to combine theory and practice within the scope of extensive interconnected technical perspectives. The student must document these demands both in writing and orally.

Generally, the Final Examination Report is written in groups of 2 to 3 students.

The students choose the topic of their final report in consultation with the Academy and preferably with a company as well. The Academy must approve the project outline.

The Final Examination Report completes the programme.

The project report must not exceed 60 pages (excluding source references and appendices).

The project report is evaluated according to the Danish 7-point marking scale. The minimum mark to pass this semester is 02.

Examination

Test 5:	**Final Examination Report**	
Subject matter:	The Final Examination Report and the curriculum of all previous semesters	
Type:	A combination of the project and an oral test	
Remarks:	In the 1st part of the test: the students are examined in groups. The students must present the Final Examination Report. Examination time is 30 minutes. In the 2nd part of the test: each student is examined individually (20 minutes).	
Evaluation:	The student is evaluated on the basis of an overall impression of the project report, the presentation by the group and the individual examination.	
Date:	Immediately after the module has ended.	
Re-examination	At the next examinations date.	
Date:		
Examination rules:	Duration of the examination:	30 minutes per group 20 minutes per group member
	Resources:	Everything
	Required material produced by the student:	The Final Examination Report