



COMPUTER SCIENCE



*Programme
Handbook*

2025/26

CONTENTS

Contents.....	1
1. Welcome and introduction	3
A. A BRIEF INTRODUCTION TO ACT	4
2. List of programme director and academic staff.....	5
3. List of support staff	7
4. Name, position and institution of the external examiner(s) involved in the programme.....	8
5. Introduction to the programme.....	9
6. Programme specification	10
Programme Of Study And Intended Learning Outcomes	12
Suggested Program of Studies	30
Curriculum map COMPUTER SCIENCE	31
7. Module specifications	33
8. Student support, guidance and advice	64
Academic Advising and Mentoring	64
The Learning Hub	64
Health Services.....	64
9. Opportunities for personal development planning	65
Career Services and Guidance.....	65
Internships	65
10. Opportunities and support for study abroad.....	65
ACT Students Studying in the US	65
11. Work placement information	66
12. Facilities and Services.....	66
Bissell Library - General Information	66
Computing Services & Networking Facilities	67
Science laboratories.....	67
Food Services	67
Housing	67
13. Determination of results (Link to OU Regulations).....	68
Minimum requirements for pass	68
Determining module outcomes	68
Bachelor Honours Degree Classification	68
14. Assessment and progression regulations (Link to OU Regulations)	69
Submission of assessed work.....	69
Assessment scores	69
Determining progression and qualification outcomes	70

Stage requirements.....	70
Resit provision.....	70
15. Dissertations and projects	71
How the Bissell Library supports students with their theses.....	71
16. Other institutional policies and regulations (Link to OU Regulations)	72
Academic Misconduct.....	72
Academic Offences	72
Extenuating Circumstances.....	74
Mitigating Circumstances.....	74
17. Student participation and evaluation	76
Student Participation	76
Module Evaluation.....	76
College-wide feedback.....	76
18. General reading list (not module specific), including electronic resources.....	77

1. WELCOME AND INTRODUCTION

WELCOME!

Dear Student,

This Handbook is intended to provide a concise reference and guide for all ACT students. Included herein are brief statements of College policies applicable to and of interest to all College constituencies.

This guide is intended to address some of the common academic and student life questions and concerns that are likely to arise during your years at ACT. Its purpose is not merely to lay out the policies of academic life, but also to point out ways of thinking about your education. Planning carefully and looking ahead will enable you to maximize your opportunities at the College. Knowing when and where to find guidance and counsel is important in ensuring that your educational choices are well-considered and make sense in the context of your larger academic goals.

We do assume, however, that you will seek out the help you need. This guide has been designed to help you do just that. Please read it, keep it, and use it as a reference throughout your academic career.

An additional channel of communication is the ACT Viewbook/Catalogue of Study (annual publication consisting of the analytical programs of study, course offerings and course descriptions) and related resources that can be found on the ACT website.

With best regards,

Dr. Stamos Karamouzis
ACT Provost

A. A BRIEF INTRODUCTION TO ACT

The American College of Thessaloniki (ACT, founded in 1981 as a two-year, Associate-degree granting institution) is the tertiary-level division of Anatolia, a private, non-profit educational institution founded in 1886. It is incorporated in, and chartered by the Commonwealth of Massachusetts, and it is fully accredited by the New England Commission of Higher Education (NECHE). In inspiration, mission, governance, and programs, ACT resembles the traditional New England colleges upon which it has been modeled.

Founded by American Protestant missionaries, Anatolia was originally located in Asia Minor and evolved from a seminary in Constantinople, which began in 1840. The school was closed during the Greek-Turkish War of 1919-1922 and ceased to have a viable mission in Asia Minor when Turkey's minority communities were uprooted under the peace treaties concluding the war. In 1924, Anatolia relocated to Thessaloniki, where the greatest part of the refugee influx from Asia Minor had settled. In the mid-1930s, the school moved to its present location on a forty-five acre campus a few miles from the center of the city.

ACT moved to a four-year college in 1989 and currently offers Bachelor's degrees in the areas of Business, Technology, English, Psychology and Politics & International Relations. Since 2002, ACT introduced graduate programs, the first one in Business (MBA) with concentrations in key disciplines such as Marketing, Management, Entrepreneurship, and Banking & Finance. In 2016, a graduate program in Hospitality & Tourism Management was introduced, followed by an Industrial/Organizational Psychology one in 2020. All graduate programs are designed to accommodate graduates from all disciplines and can be completed on full-time or part-time basis within one or two years. In September 2013, ACT entered a new validation agreement for all its undergraduate programs with one of the top British Universities, the British Open University.

In recent years, ACT has received grants from a number of foundations, notably among which are the Anagnos Foundation, the Andrew Mellon Foundation, the Cleveland H. Dodge Foundation, the N. Demos Foundation, the Minneapolis Foundation, the Pappas Foundation, (US), the J. F. Costopoulos Foundation, and the Stavros S. Niarchos Foundation (Greece). These grants, contributions by many individual donors in Greece and in the US, and most particularly the extraordinary contributions of Mr. George Bissell, Chair of the Board of Trustees, have made possible the creation of a world-class campus and of the Bissell Library, a state of the art facility unique in SE Europe.

2. LIST OF PROGRAMME DIRECTOR AND ACADEMIC STAFF (www.act.edu/academics/faculty)

CHAIR

Dr. Vagelis Chatzistavros

Assistant Professor (Computer Science)(Reg)
Dipl,Electrical and Computers Engineering, Democritus University of Thrace;
MSc, Communication Network, Democritus University of Thrace;
PhD, Electrical and Computers Engineering, Democritus University of Thrace

ACADEMIC STAFF

Dr. Astaras, Alexandros

Assistant Professor (Computer Science) (Reg)
B.A. in Physics, Oberlin College, Oberlin, USA;
PhD in Electronics Engineering, The University of Edinburgh, Edinburgh, Scotland, UK

Dr. Baglavas, Grigoris

Assistant Professor (Computer Science)(Reg)
BSc, Mathematics, Aristotle University of Thessaloniki;
MSc, Telematics, University of Sheffield;
PhD, Computer Science, University of Macedonia

Mr. Christodoulou, Christos

Adjunct Instructor (Computer Science)(Adj)
BSc., Physics, Aristotle University of Thessaloniki, Greece;
MSc., Information Technology, University of Aston in Birmingham, Birmingham, U.K.

Dr. Vasilios Daskalogiannis

Assistant Professor (Mathematics) (Reg)
BS, Mathematics, Aristotle University of Thessaloniki, Greece; MSc in Pure Mathematics, Aristotle University of Thessaloniki, Greece; PhD, Mathematics, Department of Mathematics of the Aristotle University of Thessaloniki

Dr. Konstantinos Kanakoglou

Assistant Professor (Mathematics) (Reg)
B.Sc. in Physics, Aristotle University of Thessaloniki, Greece; M.Sc. in Pure Mathematics, Aristotle University of Thessaloniki, Greece; Ph.D. in Mathematical Physics, Aristotle University of Thessaloniki, Greece

Mr. Kaplanoglou, Pantelis

Adjunct Instructor (Computer Science)(Adj)
BSc., Software Engineering, A.T.E.I. of Thessaloniki, Greece;
MSc., Web Intelligence, A.T.E.T. of Thessaloniki, Greece;
Ph.D Candidate, Machine Learning for Computer Vision, Aristotle University of Thessaloniki, Greece.

Mr. Karamichalis, Menelaos

Adjunct Instructor (Computer Science)(Adj)
BA, Physics, Berea college, Berea, KY, USA;
B.Sc. in Electrical Engineering, Washington University, St. Louis, MO, USA;
MSc. in Electrical Engineering, Washington University, st. Louis, MO, USA;
Master of Engineering Management, Washington University, St. Louis, Mo, USA

Mr. Kourakis, Orestis

Adjunct Instructor (Digital Photography)(Adj)

BSc., School of Agriculture, Aristotle University of Thessaloniki, Greece;

MA in Photography, Savanna College of Art and Design, U.S.A.

Dr. Mengoudi, Kyriaki

Assistant Professor (Computer Science)(Reg)

BSc, Mathematics, Aristotle University of Thessaloniki;

MSc., Applied Statistics and Data Mining, University of St Andrews, Scotland, UK;

Ph.D.,Computer Science, University College

London, UK

Mr. Morris, Brian

Adjunct Instructor (Computer Science)(Adj)

BSc., Computer Processing, Illinois Central College, U.S.A.;

BFA, The School of the Art Institute of Chicago, U.S.A.;

MA, Digital Arts, University of Arts London at Camberwell, London, UK.

Mr. Victoratos, Iosif

Adjunct Professor (Computer Science)(Adg)

BS.,MSC., Computer Engineering and Informatics, University of Patras, Greece;

MSc., Informatics and Management, Aristotle University of Thessaloniki, Greece;

Ph.D.,Semantic Web Technologies, Aristotle University of Thessaloniki, Greece.

3. LIST OF SUPPORT STAFF

ADMINISTRATIVE OFFICES

Office/Division	Name	Location	Phone No. (2310+No.)	e-mail (name@act.edu)
President	Dr. Panayiotis Vlachos	Stephens Hall, 1 st Floor	398204	pvla
Executive Assistant to the President	Ms. Elena Charalambides	Stephens Hall, 1 st Floor	398204	elenacha
Vice-President for Operations & Planning	Mr. Ioannis Tsorbatzoglou	Stephens Hall, 1 st Floor	398326	yatso
Vice President for Institutional Advancement	Mr. Peter Chresanthakes	Stephens Hall, 1 st Floor	398265	peter
Vice President for Finances & HR/CFO	Mr. Pavlos Floros	Stephens Hall, 1 st Floor	398214	pfloros
Assistant Accountant	Ms. Eva Montiadou	Stephens Hall, Ground Floor	398219	emont
Alumni and Public Relations Officer	Ms. Marina Charitopoulou	Stephens Hall, 2 nd Floor	398220	mcharito
Director of Marketing	Mr. Theodore Papanestoros	Stephens Hall, 1 st Floor	398385	theodore
Director of International Programs	Ms. Heather Funk	Constantinidis Hall, 1st Floor	398215	heather
International Programs and Student Services Coordinator	Ms. Maria Maleas	Constantinidis Hall, 1st Floor	398205	mamalea

Director of Admissions	Ms. Roula Lebetli	Bissell Library, 1 st Floor	398239	admissions
Supervisor of the Registrar's Office	Ms. Christina Moma	Constantinidis Hall, 1st Floor	398207	actreg
Senior Officer at the Registrar's Office	Ms. Theodora Zafiriou	Constantinidis Hall, 1st Floor	398224	actreg
Business Liaison & Career Services Officer	Mr. Dimitris Chatzigeorgiou	Bissell Library, Ground Floor	398337	ddiamantis
Assistant Administrative Officer	Mr. Vassilis Loukidis	Constantinidis Hall, 1st Floor	398216	vloukid

4. NAME, POSITION AND INSTITUTION OF THE EXTERNAL EXAMINER(S) INVOLVED IN THE PROGRAMME

- Dr. Tsompanas Antisthenis, University of the West of England
- Dr. Paul Neve, Kingston Univesity

5. INTRODUCTION TO THE PROGRAMME

Contemporary Information Technologies change rapidly in all levels of scope from hardware to conceptual. The ACT programme on Computer Science aims at offering its students a solid foundation that both addresses the fundamentals and provides adaptability in a lifelong career with continuous learning.

The ACT computer science graduate receives a solid and thorough education in fields of computing that interconnect as well as provide a deep and wide background in contemporary computer science. The programme is structured in threads, with some capstone modules unifying among them and a set of Mathematics modules providing necessary background knowledge for the contemporary computer scientist.

The Programme aims to:

- Equip students with knowledge, skills and inspiration for a career at the forefront of innovation or further studies and research in computer science;
- Provide QAA standards level education in computer Science appropriate for either a career in industry or graduate studies. Such education shall cover a wide range of knowledge and understanding in all relevant areas of a rigorous curriculum and foster problem solving skills and information literacy
- Develop cognitive skills needed by the computer scientist: the ability to model systems, the power of abstraction, the ability to communicate technical arguments
- Provide the ability to critically evaluate computer systems, their performance and their specifications and demonstrate a high-level of professional competence across a range of technical, legal and ethical areas.
- Instill professional and entrepreneurial attitudes in students and develop a range of transferable skills that would enable them to advance and exploit the knowledge and technical expertise in pursuing their further career
- Demonstrate the applicability of knowledge and skills in various contexts in which computer systems are developed, either when working alone or effectively participating as members of international teams

ACT currently offers a BCs (Hons) in Business Computing which is both Validated from the OU and Accredited by NEASC (New England Association of Schools and Colleges).

Graduates of this degree are usually gainfully employed in the IT departments of businesses or IT companies, and many follow graduate study courses abroad.

A BSc in Computer Science was offered by ACT accredited NEASC and validated from Wales University, from 2004 till 2010. The Computer Science program was re-introduced and validated from the OU in 2015, and ever since both the Computer Science and Business Computing programs are offered.

In order to receive the BSc degree, the student must have fulfilled all the GER and major requirements and have completed at least 121 US credit hours with an overall G.PA of 2.0 or better. All Computer Science students take a two-semester sequence Senior Thesis I and II course. According to NEASC Standards, students must complete at least one fourth of their undergraduate program, including advanced work in the major or concentration, at the institution awarding the degree. As a consequence, all candidates for an ACT degree must have been in residence at the College during the last two semesters of full-time instruction, assuming availability and equivalency of transferable courses

6. PROGRAMME SPECIFICATION

About the Programme

The mission of the Division of Technology & Science is to offer innovative, leading edge technology programs in Computer Science and academically sound service courses in the areas of Mathematics, Statistics and Science. As Computer Science is a rapidly evolving discipline we continuously adapt our curriculum and facilities to meet the changing demands of the Computer Science profession.

The program targets:

- Students that are interested primarily in Computer Science
- Students or professionals that are interested to specialise in certain areas in computing. In particular the certificate and special programs provide training opportunities for the wider community.

Modules in the Division of Technology & Science are designed to broaden students' perspectives on the role of computer science, mathematics, statistics and science in the modern world, while equipping them with programming, computer literacy and quantitative skills. A broad range of Computer Science courses is offered, the majority having a strong laboratory component with emphasis on application

ACT Degree Competitive Advantage Areas

An ACT graduate with the BSc (Hons) in Computer Science will have obtained a theoretical and practical adequacy in the field of Computer Science application and design, some business domain knowledge and directly marketable skills through the ability to further obtain Professional Certifications (CCNA, ORACLE).

Graduates of the program will be able to pursue careers in the areas of

- Programming
- Databases & Modeling

- Computing Systems and Networks
- Web & Mobile Applications
- Artificial Intelligence and Data Science

Another distinctive feature of this programme is its unique blend of American and British educational standards. ACT offers a four-year degree, which includes many modules additional to the ones comprising the programme and submitted for validation. These modules cover General Education Requirements, an American tradition of Liberal Education, which enriches student learning beyond subject specialization

Special Features

The programs do not concentrate only on the latest technologies, which at some point will become outdated, but provide students with excellent analytical skills and systematic thinking that will allow them to become lifelong learners and succeed in a wide variety of technical and managerial positions. Students are prepared for a successful career in the field of computing and its applications and/or additional study in computing or Business at the graduate level.

Computing and Teaching Facilities

The Division maintains the following facilities available to be used by its students:

- **Four teaching computer labs**, annually upgraded in terms of Hardware infrastructure and Software packages (updated versions and licences).
 - **Lab 4:** General purpose computer lab used for computing module instruction (25 stations in total: 24 student workstations, 1 instructor station);
 - **Lab 5:** Electronics and Robotics student/faculty research lab with 21 workstations (20 student workstations, 1 instructor station), a classroom set of 11 Lego EV3 robots, a classroom set of 11 Arduino kits, electronics equipment (Function Generators, Oscilloscopes, Power Supplies, multimeters) and hand tools;
 - **Lab 6:** Lab appropriately designed to fulfil the requirements of multimedia and other computing modules (25 stations in total: 24 student workstations, 1 instructor station);
 - **Lab 7:** General purpose computer lab, used for computing modules' instruction and Bibliographic instruction (25 stations in total: 24 student workstations, 1 instructor station).
- **Cloud computing server infrastructure** is in place and ACT.
- A **CISCO lab:** physical lab with programmable routers/switches and PCs for the instruction of networking modules.
- **Programmable devices** are available to students for their own experimentation and practice, such as a quad copter, Microsoft Kinect camera system, programmable android based watches. The Division can apply for small equipment grants (approx. 1000 € each) to purchase such devices as per the students' and faculty requests in the context of IT-related projects. There is also an annual budget of approximately 2,000 € that can and is used for such purchases.

PROGRAMME OF STUDY AND INTENDED LEARNING OUTCOMES

<u>Programme Structure - LEVEL 4</u>					
Compulsory modules	Credit points	Optional modules	Credit points	Is module compensatable?	Semester runs in
CSC 105 - Structured Programming	15	ECON 101 - Introductory Economics	15		
CSC 106 - Object Oriented Programming	15	MRKT 101 - Introduction to Marketing	15		
CSC 205 - Business Data Management	15	ACC 101 - Financial Accounting	15		
CSC 215 - Data Structures	15				
CSC 230 - Systems Programming	15				
MATH 120 - Calculus I	15				
STAT 210 - Introductory Statistics	15				

Intended learning outcomes at Level 4 are listed below:

Learning Outcomes – LEVEL 4

3A. Knowledge and understanding

Learning outcomes:

On completion of this level you will be able to:

A1. **understand** a computer science related scientific method and its applications to problem-solving in a specific area

A2. **identify** and **describe** some essential facts

A3. **describe** and **explain** principles and theories relating to subject areas of computer science

A4. **identify** knowledge and **name** computer applications as appropriate to the course of study

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Learning Outcomes – LEVEL 4

3A. Knowledge and understanding

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3B. Cognitive skills

Learning outcomes:

On completion of this level you will be able to:

- B1. **recognise** critical thinking, including its relevance to everyday life
- B2. **identify** such knowledge and understanding in the modelling and design of computer-based systems
- B3. **outline** the criteria and specifications appropriate to specific problems
- B4. **review** the criteria met by a computer system, as they are defined for its current use and future development
- B5. **name** and **discuss** a set of rational and reasoned arguments, addressing a given problem or opportunity in a target audience.
- B6. **give** simple **examples** of economic, professional, social, environmental, moral and ethical issues in the sustainable exploitation of computer technology

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts

3B. Cognitive skills

- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3C. Practical and professional skills

Learning outcomes:

On completion of this level you will be able to:

- C1. **Recognise** and **describe** the high-level architecture of computer-based systems
- C2. **Identify** the quality attributes and possible trade-offs a system in the context of a given problem
- C3. **Name** any risks or safety aspects during the deployment of a system or solution in the context of a given problem
- C4. **Compare** some of the essential tools per study area, used for the construction and documentation of an application.

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

3C. Practical and professional skills

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3D. Key/transferrable skills

Learning outcomes:

On completion of this level you will be able to:

- D1. be enumerate and literate in **describing** cases which involve both quantitative as well as qualitative dimensions
- D2. Retrieve information from various sources (search engines, catalogues etc.)

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

3D. Key/transerable skills

D3. **operate** general Information Technology facilities
D4. **practice** on the effective goal setting and action planning

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

Programme Structure - LEVEL 5

Compulsory modules	Credit points	Optional modules	Credit points	Is module compensatable?	Semester runs in
CSC 206 - Web Development	15				
CSC 300 - Mobile Application Development	15				
CSC 306 - Advanced Web Development	15				
CSC 310 - Hardware & Computer Architecture	15				
CSC 312 - Database Management Systems	15				
CSC 340 - Artificial Intelligence	15				
CSC 450 - System Analysis & Design	15				
MATH 220 - Discrete Mathematics	15				

Intended learning outcomes at Level 5 are listed below:

Learning Outcomes – LEVEL 5

3A. Knowledge and understanding

Learning outcomes:

On completion of this level you will be able to:

- A1. **practice** on a computer science related scientific method and apply it for problem-solving in a specific area
- A2. **identify** and **distinguish** over a wide range of essential facts and concepts of Computer Science
- A3. **experiment** and **test** principles and theories on intermediate level
- A4. **identify** and **name** computer applications as appropriate to the course of study
- A5. **analyse**, **test** and **experiment** with the appropriate theory, practices and tools for the specification, design, implementation and evaluation of computer-based systems

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Learning Outcomes – LEVEL 5

3A. Knowledge and understanding

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3B. Cognitive skills

Learning outcomes:

On completion of this level you will be able to:

B1. **demonstrate** critical thinking, including its relevance to everyday life

B2. **employ** and **apply** such knowledge and understanding in the modelling and design of computer-based systems

B3. **predict** and **produce** a set of the criteria and specifications appropriate to specific problems

B4. **formulate** and **revise** the criteria met by a computer system, as they are defined for its current use and future development

B5. **explain** and **illustrate** a set of rational and reasoned arguments, addressing a given problem or opportunity in a target audience.

B6. **categorise** a number of economic, professional, social, environmental, moral and ethical issues in the sustainable exploitation of computer technology

B7. **design** and **develop** the appropriate theory, practises and tools for the specification, design, implementation and evaluation of computer-based systems

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts

3B. Cognitive skills

- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3C. Practical and professional skills

Learning outcomes:

On completion of this level you will be able to:

- C1. **Construct** and **illustrate** the architecture of reliable, secure and usable computer-based systems
- C2. **analyse** and **examine** the quality attributes and possible trade-offs a system in the context of a given problem
- C3. **Inspect** and **test** any risks or safety aspects during the deployment of a system or solution in the context of a given problem
- C4. **Employ** some of the essential tools per study area, used for the construction and documentation of an application.

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

3C. Practical and professional skills

- C5. **Operate** under specific project requirements to produce deliverables that take into consideration project/system requirements and budget.
- C6. **Analyse** and **discover** the process involved in the development and deployment of a system for solving real-life problems

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3D. Key/transferrable skills

Learning outcomes:

On completion of this level you will be able to:

- D1. be enumerate and literate in **describing** cases which involve both quantitative as well as qualitative dimensions

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

3D. Key/transerable skills

- D2. retrieve information from various sources (search engines, catalogues etc.)
- D3. **operate** general Information Technology facilities
- D4. **practise** on the effective goal setting and action planning
- D4. **Identify** problems that may arise and **devise** their solutions in the context of a computer science project
- D5. **Outline** and **generate** the best possible outcome while working along with a group of individuals

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

Programme Structure - LEVEL 6

Compulsory modules	Credit points	Optional modules	Credit points	Is module compensatable ?	Semester runs in
CSC 321 - Operating Systems	15	CSC 219 - Video Game Design	15		
CSC 322 - Computer Networks	15	CSC 330 - Mobile Robotics	15		
CSC 325 - Distributed Applications	15	CSC 333 - Computer Networks II	15		
CSC 412 - Object Oriented Design Patterns	15	CSC 422 - Advanced DBMS	15		
CSC 421 - Computer Systems Security	15	PRAC 300 - Practicum	15		
CSC 443 - Thesis I	15				
CSC 443 - Thesis II	15				

Intended learning outcomes at Level 6 are listed below:

Learning Outcomes - LEVEL 6

3A. Knowledge and understanding

Learning outcomes:	Learning and teaching strategy/ assessment methods
<p>A. Knowledge and understanding - On completion of this level you will be able to:</p> <p>A1. reproduce a computer science related scientific method and extend its applications to problem-solving in a specific area</p> <p>A2. employ and practice advanced facts</p>	<p>Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.</p>

Learning Outcomes – LEVEL 6

3A. Knowledge and understanding

A3. **model** and **test** principles and theories relating to subject areas of computer science

A4. **analyse** knowledge and **revise** computer applications as appropriate to the course of study

A5. **assess, interpret and evaluate** the appropriate theory, practises and tools for the specification, design, implementation and and evaluation of computer-based systems

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3B. Cognitive skills

Learning outcomes:

- A. Cognitive skills** - On completion of this level you will be able to:
- B1. **demonstrate** critical thinking, including its relevance to everyday life
- B2. **combine** and **interpret** such knowledge and understanding in the modelling and design of computer-based systems
- B3. **devise** and **judge** a set of the criteria and specifications appropriate to specific problems
- B4. **measure** and **assess** the criteria met by a computer system, as they are defined for its current use and future development
- B5. **compare** and **conclude** to a set of rational and reasoned arguments, addressing a given problem or opportunity in a target audience.
- B6. **rate** a number of economic, professional, social, environmental, moral and ethical issues in the sustainable exploitation of computer technology
- B7. **propose** and **argue** on the appropriate theory, practises and tools for the specification, design, implementation and evaluation of computer-based systems

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations

3B. Cognitive skills

- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3C. Practical and professional skills

Learning outcomes:

On completion of this level you will be able to:

- C1. **Design** and **illustrate** the architecture of reliable, secure and usable computer-based systems
- C2. **Predict** and **justify** the quality attributes and possible trade-offs a system in the context of a given problem
- C3. **Estimate** and **evaluate** any risks or safety aspects during the deployment of a system or solution in the context of a given problem
- C4. **Develop** some of the essential tools per study area, used for the construction and documentation of an application.
- C5. **Operate** under specific project requirements to produce deliverables that take into consideration project/system requirements and budget.
- C6. **Assess** and **revise** the process involved in the development and deployment of a system for solving real-life problems

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences

3C. Practical and professional skills

- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

3D. Key/transferrable skills

Learning outcomes:

On completion of this level you will be able to:

- D1. be enumerate and literate in **describing** cases which involve both quantitative as well as qualitative dimensions
- D2. retrieve information from various sources (search engines, catalogues etc.)
- D3. **operate** general Information Technology facilities
- D4. **practise** on the effective goal setting and action planning
- D4. **Identify** problems that may arise and **devise** their solutions in the context of a computer science project
- D5. **Outline** and **generate** the best possible outcome while working along with a group of individuals

Learning and teaching strategy/ assessment methods

Guided teaching environment (Lectures & labs) is the principal method of delivery for the concepts, principles and skills involved in the outcomes. Students are also directed to reading from textbooks, academic papers and other relevant material.

Understanding is reinforced by means of exercise classes, discussion groups, laboratories, assignments and project work.

Tools to be used to achieve this will include some or all from the following:

- printed and online teaching texts
- directed readings from textbooks and papers
- Specialised software tools.

Support of learning:

Learning is supported outside the classroom with the use of the learning management system Moodle, instructor office hours, sample answers to assessment and extra lectures as seen appropriate by the instructor.

3D. Key/transferable skills

Tools to be used to achieve this will include some or all from the following:

- self-assessment questions and exercises, included in the teaching texts
- programming tasks, computer-based investigations and open-ended project work
- feedback and guidance from an instructor; tutorials, revisions and in-class activities
- e-mail and individual instructor-learner conferences
- Study and project guides.

Assessment of learning:

An assessment of the understanding of underlying concepts and principles forms part of the overall assessment of final exams/projects submitted/taken.

Tools to be used to achieve this will include some or all from the following:

- Instructor-Marked summative formal examinations
- Instructor-Marked summative projects
- Instructor-Marked summative presentations
- Instructor-Marked formative assignments/assessment
- Instructor-Marked formative projects

[BA Ordinary in Computer Science / 300 credits (120 at Level 4, 120 at Level 5, 60 at Level 6)]

OR

[BA (Hons) in Computer Science / 360 credits (120 at Level 4, 120 at Level 5, 120 at Level 6)]

SUGGESTED PROGRAM OF STUDIES

BSc (Hons) in Computer Science

Year 1					
Fall	Grade	Spring I	Grade	Spring II	Grade
CS105 - Structured Programming		CCS106 - Object Oriented Programming		ACT Module (GER 4)	
ACT Module (CS180 - Discrete Structures)		MATH 120 - Calculus I		ACT Module (GER 5)	
ACT Module (GER 1)		CSC107 - Multimedia toolkit			
ACT Module (GER 2)		ACT Module (GER 3)			
Year 2					
Fall	Grade	Spring I	Grade	Spring II	Grade
CSC205 - Business Data Management		CSC312 - Database Management Systems		CSC450 - System Analysis & Design	
CSC215 - Data Structures & Algorithms		CSC230 - Systems Programming		ACT Module (FE2)	
Business Elective (ECON or MRKT or ACC 101)		STAT210 - Introductory Statistics with R			
ACT Module (FE1)		ACT Module (GER 6)			
Year 3					
Fall	Grade	Spring I	Grade	Spring II	Grade
CSC310 - Hardware & Computer Architecture		MATH 220 - Discrete Mathematics		CSC300 - Mobile Application Development	
CSC206 - Web Development		CSC306 Advanced Web Development		CSC325 - Distributed Applications	
CSC340 - Artificial Intelligence		ACT Module (FE3)			
ACT Module (GER 7)		ACT Module (FE4)			
Year 4					
Fall	Grade	Spring I	Grade	Spring II	Grade
CSC321 - Operating Systems		CSC322 - Computer Networks		CSC421 - Computer Systems Security	
CSC412 - Object Oriented Design Patterns		CSC444 - Thesis II		ACT Module (GER9)	
Major Elective		ACT Module (GER8)			
CSC433 - Thesis I		ACT Module (FE5)			

Modules in **bold** are OU Majors. **GER:** General Education Requirement. **FE:** Free Elective

7. MODULE SPECIFICATIONS

1. Factual information			
Module title	COMPUTER SCIENCE 105: Introduction to Programming I	Level	4
Module tutor	Dr. Alexander Astaras	Credit value	15
Module type	Taught: Lecture/guided discussion	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

This is an introduction to computing and computer programming using the Java language. It is one of the two options available for all students to satisfy the mandatory 1st year computing course requirement (the other option is CS101). It is the only option for those students interested in keeping their options open towards pursuing a major in Computer Science. This course is designed for students who have no prior experience in programming, just some basic exposure to computing devices as users.

3. Aims of the module

Students are introduced to the basic elements of computing hardware, information technology and computer programming. Programming is explained, demonstrated and practiced using the Java programming language. Ultimately the course aims to advance beyond basic computing skills towards software engineering, instructing students to develop autonomy as sophisticated computer users and programmers.

4. Pre-requisite modules or specified entry requirements

None. It is expected that students have already had some exposure to a windows-based graphical user interface computing environment (Microsoft Windows, Apple OS, Linux/Unix or equivalent).

8. Indicative content.

Hardware and Software (approx. 2 weeks)

- General computer science topics.
- Analog and binary signals.
- Machine language and high level languages.

Theory of Programming (approx. 2 weeks)

- Compilers
- Algorithms
- Problem Decomposition – Stepwise Refinement

Java Programming (rest of the semester)

- How to run Java programs.
- Translating Java source code into bytecodes.
- How byte codes are interpreted.
- How to create a Java program.
- Writing a program
- Java variable types
- Storing and changing values in variables
- Selection statements
- Repeating actions with loops
- Using arrays

- User input error trapping

1. Factual information

Module title	Computer Science 106 - Introduction to Programming II – Object Oriented Programming		
Module tutor	Mr. Pantelis Kaplanoglou	Level	4
Module type	Taught: Lecture/guided discussion	Credit value	15
Mode of delivery	100% face-to-face		
Notional learning hours	150		

2. Rationale for the module and its links with other modules

The course aims to refine the participants' fundamental programming skills. It builds upon CS105 (Introduction to Programming I) and serves as preparation for CS215 (Data Structures). It is based on the Java programming language.

3. Aims of the module

The module covers of Object Oriented Modeling and Applications. Topics include Object Models, Object Class Design, Inheritance and Polymorphism, Software Reuse with Classes, Application Modeling, Simulation with Object Classes, and Business Process Modeling with Objects.

Object-oriented programming (OOP) is a revolutionary concept that changed the rules in computer program development. OOP is organized around "objects" rather than "actions", data rather than logic. Historically, a program has been viewed as a logical procedure that takes input data, processes it, and produces output data. The programming challenge was seen as how to set up the logic, not how to define the data. Object-oriented programming adopts the viewpoint that “what we really care about” are the objects we want to manipulate rather than the logic required to manipulate them.

The module expands on the material covered in CS105 with the following aims:

- Further cultivation of algorithmic thinking and refinement of procedural programming skills
- Familiarization with the Object Oriented programming methodology
- Exposure to Java classes for building graphical interfaces and other extensions

4. Pre-requisite modules or specified entry requirements

CS105

8. Indicative content

Functions and modules

- *Methods*
- *Libraries and clients*
- *Recursion*

Object Oriented Programming Methodology

- *Data types*
- *Creating data types*
- *Inheritance*
- *Variable access control*
- *Polymorphism and Interfaces*
- *Testing and debugging*

8. Indicative content

- *Exception handling*

Interface design

- *Event handling*
- *Listeners*
- *Layout classes*
- *Inheritance*

1. Factual information			
Module title	COMPUTER SCIENCE 205 – Business Data Management	Level	4
Module tutor	Christodoulou, Chris	Credit value	15
Module type	Taught	Total learning hours	150

2. Rationale for the module and its links with other modules
The purpose of COMP SCI 205 is to introduce the idea of business data management, data modelling, and processing methodologies with the use of standalone design tools and personal databases. It aims at fostering proper data design through the relational methodology and developing all necessary data processing and presentation skills.

3. Aims of the module
<p>The aims of this module are to:</p> <ul style="list-style-type: none"> Define the role of Systems Analyst and Database designer. Explain System Analysis and interpersonal communication skills that the System Analyst must have Explain Project Management and discuss tools that the system analyst must have Explain the Methodologies that are used for Systems Analysis and Database Design Explain the various tools that certain methodologies use Provide students the opportunity to work on the most popular database (Oracle), in a project in order to implement the taught methodologies.

4. Pre-requisite modules or specified entry requirements
CS 105.

8. Indicative Content
This module deals with numerous forms of Business Data employed in monitoring business operations, including Data Analysis, Data Design, Table Design and learning how to implement them using two popular database programs.

1. Factual information			
Module title	COMPUTER SCIENCE 206 – Web Development	Level	5
Module tutor	Dr. Viktoratos Iosif	Credit value	15
Module type	Taught: Lecture/guided discussion	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

COMP SCI 206 is an introductory module for beginning web designers. We will explore some essential concepts related to the creation of effective web sites. In the last portion of the module we will concentrate on client-side scripting using the programming language JavaScript.

3. Aims of the module

This module aims at introducing students the basic web design guidelines, Fundamentals of Hyper Text Markup Language (HTML), and how to use a Simple HTML Editor as well as Web Authoring Tools. Also, one of the main goals of the module will be to understand what scripting languages are and to be able to develop scripts.

4. Pre-requisite modules or specified entry requirements

8. Indicative content.

Web Design Guidelines

- What is the World Wide Web
- Basic Design Principles
- Interface & Navigation
- Good & Bad Design
- Colour on the WWW

Validation & Assessment

- Validating HTML and CSS
- Accessibility & Usability
- Web site optimization

HTML & CSS basics

- HTML source document, tags
- Lists, Tables, Links, Images, Graphics
- Intro to CSS
- CSS value and Common CSS Properties

HTML5 & CSS3

- New elements
- Multimedia, Canvas element
- Forms and Styles

Web Authoring Tools

- Setting Up a Web site
- Page Layout, CSS, Templates, Navigation
- Working with Texts, Lists and Tables
- Working with Images
- Adding Interactivity (Flash, Forms)
- Working with Online Data
- Building Dynamic Pages with Data
- Working with Code
- Publishing to the Web

JavaScript

- Introduction
- Variables
- Operators
- Functions, Statements
- Form validation

1. Factual information			
Module title	Computer Science 215 – Data Structures and Algorithms	Level	4
Module tutor	Mr. Pantelis Kaplanoglou	Credit value	15
Module type	Taught: Lecture/guided discussion	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements
The purpose of CS215 is to introduce students to the main concepts and implementation principles of data structures, using object oriented programming principles and Java as the programming language. This module builds on the knowledge and skills acquired in CS105 – Introduction to Programming I and CSC106 - Introduction to Programming II: Object Oriented Programming.

3. Aims of the module
The module starts with a fast paced review of object-oriented programming using Java, re-enforcing the fundamental programming concepts learned in CS105 and CS106. Students are subsequently introduced to data structures, including arrays, lists, queues, stacks, trees, heaps, hash tables and graphs. Searching, sorting, inserting, deleting and other simple operations on these structures will also be discussed.

4. Pre-requisite modules or specified entry requirements
CS106

8. Indicative content.
<ul style="list-style-type: none"> ▪ Java object oriented programming (review) ▪ Object-Oriented Design ▪ Stacks, Queues ▪ Vectors and Lists ▪ Big-O notation ▪ Sorting algorithms ▪ Trees ▪ Heaps ▪ Search Trees ▪ Hash Tables ▪ Sorting ▪ Graphs

1. Factual information			
Module title	COMPUTER SCIENCE 219 - Video Game Design with Unity and 3ds Max	Level	6
Module tutor	Mr. Brian C. Morris	Credit value	15
Module type	Taught: Lecture/guided discussion/Project	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

This module introduces the critical study of computer video games and the professional practice of game design. Through readings, discussions, research, and practical “hands-on” projects, students will better understand the current market for games and simulations and develop the fundamental skills necessary to enter the international computer games industry. Although the commercial video game pipeline will be discussed, the actual production framework for the class will mirror an 'Indie" game team "prototype game level" development.

3. Aims of the module

Students will be expected to fill multiple roles in the production process and gain hands-on experience in the collaborative processes of game design, project management, scripting, content creation pipeline, in game animation, and play-testing.

4. Pre-requisite modules or specified entry requirements

Computer Science 107

8. Indicative content.

Video Game Theory and Concepts
 Game Types
 Anatomy of a Game
 The Key Components of Video Games
 Structure of a Video Game

Design Components and Processes
 The Stages of the Design Process
 The Game Design Team Roles
 The Game Design Documents

Gameplay as challenges and actions

Core Mechanics

Game Balancing

General Principles of Level Design

3ds Max
 3d modeling and materials

The Game industry
 Game Engines
 Game delivery environments

Unity 3d Game Engine
 Unity 3d concept

Unity3d User Interface (GUI) The 3d workspace Unity Game Physics Scripting in Unity Player Camera Controllers Terrains Prefabs Audio Animation Particles AI and Enemies
Game Assets in Unity 3d Working with 3d assets in Unity Imported Assets and Packages Material and Textures

1. Factual information			
Module title	COMPUTER SCIENCE 230: Introduction to Systems Programming	Level	4
Module tutor	Dr. Alexander Astaras	Credit value	15
Module type	Taught: Lecture/Laboratory	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements
<p>The module continues along the structured programming training approach introduced in CS105, aiming to familiarize students with the C++ language, a variety of fundamental software engineering challenges, as well methodologies to analyze them and design coding solutions. The module methodically cultivates the development of algorithmic design skills with increased emphasis on systems programming. More elaborate data structures are manipulated, the role of libraries accessing operating system resources (I/O streams, file streams, etc.) is discussed, demonstrated and practiced. Students are finally introduced to the concept of pointers within the historic context of evolution of object-oriented programming languages. In this manner the module serves as a bridge between programming fundamentals and the Computing Systems programming thread.</p>

3. Aims of the module
<p>The module employs a high-level language (C++) and further investigates structured programming, following up on the introductory programming module (CS105). More elaborate structures are introduced and employed in order to provide solutions for a wide range of tasks. The intricacies of the C/C++ languages are investigated and related to computer architecture: pointers, variable addresses, memory allocation.</p> <p>In addition to further development of algorithmic thinking skills, the module also serves as an introduction to the Computing Systems programming thread. Furthermore, the module investigates the relationship of high-level programming languages with underlying computing hardware, while system programming tasks involving I/O are related to a variety of external devices (user interface, storage devices, microcontrollers with reduced memory/CPU resources, etc).</p>

4. Pre-requisite modules or specified entry requirements
CS105 or permission by the instructor.

8. Indicative content.

- C++ primitive data types
- Control flow and structured programming in C++
- Arrays, structures, composite structures
- Disk and system I/O programming
- Exception handling
- Pointers and variable addresses
- Debugging and solution verification

1. Factual information

Module title	COMPUTER SCIENCE 300 – Mobile Applications Programming	Level	5
Module tutor	Dr. Iosif Vitoratos	Credit value	15
Module type	Taught: Lecture/guided discussion	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

This course focuses on the fundamentals of mobile strategy and development, application architecture and design. Students will have the opportunity to learn the benefits and challenges of mobile application planning, design, development and strategy through real world examples and actual project work. Through readings, discussions, research, and practical “hands-on” projects, students will better understand the current market for mobile applications and develop the fundamental skills necessary to enter the mobile application industry.

3. Aims of the module

This module aims to teach how to build cross-platform mobile solutions to solve complex problems using iOS and Android phones and tablets. The course will teach students how to develop software for iOS and Android mobile devices through real world examples and strategies. Students will be guided through a complete mobile development lifecycle during the semester, and be given the opportunity to develop a series of applications.

4. Pre-requisite modules or specified entry requirements

CS 105 or permission by the Department. Students need to have basic knowledge of software development and object oriented programming

8. Indicative content.

During the course, students will create applications using the Xcode IDE and Eclipse IDE software

The course is broken up into seven modules:

1. Introduction mobile application development
 - Overview of the Android Software Development Kit and Java programming language
 - Overview of the iOS Software Development Kit and objective-C programming language
2. Mobile Strategy and Planning
 - Application models for mobile application frameworks
3. Basic concepts when designing for mobile platforms for iOS and Android devices
 - User-interface design for mobile applications

- Managing application data
- 4. Development using both frameworks (for Android and iOS)
 - Application Lifecycle
 - User interface components
 - Sensing
 - Gestures and multi-touch interfaces
 - Using the network and a database
- 5. Integrating with Cloud Services
- 6. Mobile Testing
 - Testing methodologies for mobile applications
- 7. Deployment
 - Publishing
 - Deployment
 - Maintenance
 - Management

1. Factual information			
Module title	COMPUTER SCIENCE 306 – Advanced Web Development		
Module tutor	Dr. Viktoratos Iosif	Level	5
Module type	Taught	Credit value	15
Mode of delivery	100% face-to-face online		

2. Rationale for the module and its links with other modules

COMP SCI 306 builds upon the skills and knowledge about creating and publishing Web pages and sites taught in CS 206. It also introduces students to advanced web development areas, required for students interested in pursuing a career in web site design.

3. Aims of the module

This module aims mainly on client-side scripting using the programming language JavaScript. The objective will be to understand what scripting languages are and to be able to develop scripts. The module will also offer an introduction to jQuery library, Asynchronous JavaScript and XML (AJAX), basically showing the benefits of their use and applying it to certain programming tasks. In the last portion of the module, students will gain a practical knowledge on advanced issues of the mostly used web development language, namely PHP. By combining lectures with seminar discussions and extensive hands-on experiences the course will introduce the students both to the applied aspects of web application development technologies, but also to the theoretical issues involved.

4. Pre-requisite modules or specified entry requirements

CSC 206

8. Indicative content.

JavaScript

- Advanced Techniques
- Detect visitor's Browser & Platform
- JavaScript Validation
- jQuery library
- jQuery UI & plugins

Asynchronous JavaScript and XML

- Basics
- Ajax Frameworks
- XML/CSS into Ajax Applications

Advanced PHP

- Object-Oriented PHP
- Design Patterns for the Web

1. Factual information			
Module title	COMPUTER SCIENCE 310: Computer Architecture	Level	5
Module tutor	Dr. Alexander Astaras	Credit value	15
Module type	Taught: Lecture/Laboratory	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

This course addresses the structure and function of modern digital computing devices, ranging from the compilation process down to the hardware at the level of logic gates and transistors. Despite the pace of change and variability in the fields of informatics, electronics and computer engineering, certain fundamental digital design concepts apply consistently throughout. CS310 students will gain both the relevant theoretical understanding and have a chance to apply it in practice designing, simulating, troubleshooting and optimizing their own combinational and sequential logic circuits. The course concludes with a discussion on system level organization and architecture of modern computing devices, as well as an introduction to programming with Assembly. This module builds on knowledge and skills acquired in CS105 – Introduction to Programming I.

3. Aims of the module

Upon successful completion of the module students be able to:

- Understand and be able to explain the significance and function of fundamental components within a typical modern computing device (processor, memory, I/O, operating system), their interconnections with each other and the outside world.
- Comprehend and follow the data flow through the internal structure of a digital microprocessor.
- Understand the importance and function of logic gates as primary building components in digital design.
- Analyse combinational digital circuits and optimize them using Karnaugh maps.
- Be able to design, simulate, troubleshoot and optimize their own combinational and sequential digital logic circuits.
- Recognize and understand basic Assembly language and Machine Code.
- Understand and follow the compilation and execution of high-level programs all the way down to hardware level, through a series of computation steps occurring at various system levels.

4. Pre-requisite modules or specified entry requirements

CS105

8. Indicative content.

- Logic gates
- Digital building blocks
- Basic Boolean algebra
- Combinational logic
- Combinational optimization: Karnaugh map analysis
- Digital arithmetic circuits
- Sequential logic: flip-flops
- Registers

- Asynchronous and synchronous counters
- Finite state machines
- Assembly language and machine code
- Microarchitecture
- Memory and I/O systems
- Computing device system level architecture

1. Factual information			
Module title	COMPUTER SCIENCE 312 – Database Management Systems		
Module tutor	Mr. Christos Christodoulou	Level	5
Module type	Taught	Credit value	15
Mode of delivery	Taught, 100% face-to-face		
Notional learning hours	150; Notional value based on estimates of what it would take for a good student to achieve all learning outcomes		
Office hours			

2. Rationale for the module and its links with other modules

The purpose of COMP SCI 312 is to offer a systematic coverage of modern Database Computing theory and technology. Topics include Relational Algebra, Data Modelling, Database Design, Client-Server Database Management Systems, Interface Design, trends in Database Systems, combination of Object Oriented Modelling and Relational Databases.

This is a module in continuance of the CS 205 Business Data and it educates students how to create and maintain a fully functional relational database. This knowledge will be applied on CS 325, CS412, CS 422 and CS 444 where students are required to create and/or alter various databases, used along for programming assignments.

3. Aims of the module

The aims of this module are to:

Teach students what Client-Server Database Management System is, and

To use simple and advanced SQL along with PL/SQL programming features such as IF statements, loops, stored functions, procedures, tables, cursors, packages, triggers

To create and maintain an oracle database.

Microsoft Access and/or SQL Navigator for browsing objects and databases.

ERDPlus is used for ERD's.

Apex, SQL Plus and SQL Navigator are used as user interface of the oracle database

Students develop technical, analytical, and business skills that support the pursuit of professional careers and advanced computer science study.

4. Pre-requisite modules or specified entry requirements

CS 205 or permission by the Department.

8. Indicative Content

The module focus is on teaching SQL and PL/SQL programming languages for oracle databases. Advanced query capabilities and procedural constructs are described using Oracle SQL and PL/SQL. The theoretical foundation for using these capabilities is presented. Performance issues are discussed including indexing, key definitions, DE normalized databases and triggers, and data constraints. The role of application development in ease of use, query optimization, and system performance is discussed

1. Factual information			
Module title	COMPUTER SCIENCE 321 – Operating Systems	Level	6
Module tutor	Mr. Menelaos Karamichalis	Credit value	15
Module type	Taught: Lecture/guided discussion	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

The module deepens understanding of how contemporary computing systems are structured and, in particular, supported by an Operating System (OS). It is a culmination module within the Computing Systems programme thread. The module follows CS230-Introduction to Systems Programming and CS310 – Computer Architecture. The module is frequently updated in view of rapid technological developments and their implications. The module is followed by CS421 – Systems Security as well as any future module covering the subject at the advanced level.

3. Aims of the module

Operating Systems are the brain of any computing system. They handle the body/DNA (hardware) as well as behaviour (usage of system by user). Following rapid to revolutionary technological developments the field of Operating Systems also undergoes tremendous changes, which constantly evolve the conception of an OS and of course the technological challenges involved in its implementation. As such, virtualization technologies are presented, with virtual machines used in class, along with advanced virtualization techniques, such as containers. Finally, cloud services are presented and utilized as yet another method to implement an OS.

The module aims at outlining the role of an OS in a diachronic way while comparing and contrasting design choices spanning the evolution of the field. It aims at defining fundamental needs that a von Neumann machine has from the Operating System in order to be functional, optimal and attractive to the user.

The module explains Operating Systems architecture and examines trade-offs involved in different, evolving systems. It further examines diachronic as well as contemporary issues involved in Operating System design by comparing and contrasting relevant design and algorithmic choices.

The module involves lab work: Communication with the OS at a low level via a Linux shell and programming tasks addressing aspects of Operating System design and implementation.

4. Pre-requisite modules or specified entry requirements

CS 215, CS230, CS 310, STAT 205 or permission by the Department

- 8. Indicative content.**
- Computer System Structures
 - Operating System Structures
 - Processes
 - Threads
 - CPU Scheduling
 - Process Synchronization
 - Deadlocks
 - Memory Management
 - Virtual Memory
 - File-System Interface
 - File-System Implementation
 - I/O Systems
 - Mass-Storage Structure

- Virtual Machines & virtualization
- Containers

1. Factual information

Module title	COMPUTER SCIENCE 322 – Network Operating Systems and Administration	Level	6
Module tutor	Dr. Vagelis Chatzistavros	Credit value	15
Module type	Taught: Lecture/guided discussion	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

COMP SCI 322 will address the principles, architectures and protocols that have gone into the development of modern networks from Local Area Networks to the Internet.

3. Aims of the module

This module aims to provide the student with the knowledge of how computer networks are designed, engineered and operated. This includes knowledge of the fundamental algorithms used in the management of both resources and traffic and how these algorithms may interact with application programs.

Instruction includes, but is not limited to network terminology and protocols, network standards, LANs, WANs, OSI models, cabling, cabling tools, routers, router programming, star topology, and IP addressing. The student will study and design networks using Ethernet, TCP/IP Addressing Protocol, and dynamic routing.

Particular emphasis is given to the use of decision-making and problem-solving techniques in applying science, mathematics, communication, and social studies concepts to solve networking problems.

4. Pre-requisite modules or specified entry requirements

CSC 215.

8. Indicative content.

- a) Client-server and peer-to-peer networks, network applications and protocols, hardware and topologies, OSI model, troubleshooting
- b) Networks and cabling, network diagrams
- c) MAC Address and functionalities, TPC/IP configurations and subnetting, DNS and name resolutions, command-line troubleshooting problems
- d) TCP/IP core functions, communications' management, TCP/IP utilities
- e) Data transmission concepts and metrics, cables physical characteristics, various networking media
- f) Wireless networking: theoretical background and implementation
- g) Subnets and VLANs: subnet implementation and vlan operation

1. Factual information			
Module title	COMPUTER SCIENCE 325 – Distributed Applications	Level	6
Module tutor	Mr. Chris Christodoulou	Credit value	15
Module type	Taught	Total learning hours	150;
Office	Bissel 021	Module type	Taught
Office hours		Contact Hours	22

2. Rationale for the module and its links with other modules
The purpose of CS 325 is to examine in detail the software and hardware technologies prevalent in the Internet and provide an introduction to the principles and methods for creating distributed on-line client/server applications that are the basis for electronic commerce as it is conducted over the Internet. Methods and tools such as HTML, the Common Gateway Interface, PHP, database connectivity tools and MySQL are presented. Coverage is also given to emerging standards for information exchange, encryption and validation.

3. Aims of the module
The aims of this module are to teach students the technological background as well as programming languages useful in developing and deploying internet-based applications with dynamic content. Students will learn the difference between server-side and client-side programming, the overall architectural framework of such application systems and the differences between emerging distributing technologies.

4. Pre-requisite modules or specified entry requirements
CS 105

8. Indicative content.	
Distributed Systems:	Introduction to Distributed Systems, Resource sharing and the Web Architectural Models, Fundamental Models
Protocols	Networking and InterNetworking, Network Principles, Internet
	Network Cases (Ethernet, Mobile, ATM)
PHP:	Variables, Data types, Operators, Expressions
	Control flow statements, Functions
	Arrays, Objects
MySQL:	MySQL Data types
	Basic sql statements
	Transactions and Stored Procedures
	Integration with PHP

1. Factual information			
Module title	COMPUTER SCIENCE 330: Mobile Robotics Programming	Level	6
Module tutor	Dr. Alexander Astaras	Credit value	15
Module type	Taught: Lecture/Laboratory	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

Autonomously navigating mobile robots face the challenge of acquiring data from their surroundings, selecting their own navigation waypoints and dynamically altering their course of action to account for obstacles, power supply restrictions and unexpected events. In this course theoretical instruction is combined with experiential learning and challenge driven software development. Students participating in this course are challenged individually and in teams to build the hardware chassis and software control algorithms for mobile robots. An introduction to robotics fundamentals is followed by guided programming for automation (C, RobotC); basic electronics circuit design and troubleshooting; microcontroller programming; sensor data acquisition algorithms; actuator control; robotic navigation and obstacle avoidance; basic sensor data fusion; and concludes with a final robotic design challenge which integrates all aforementioned knowledge and skills. This module builds on knowledge and skills acquired in CS230 – Introduction to Systems Programming.

3. Aims of the module

Students are guided through a series of lectures, experiments, design challenges and a final course project, learning to holistically design software, electronic circuits and mechanical constructions. Upon completion of the course, they are expected to be able to:

- build, program and troubleshoot mobile robots with a variety of roaming behaviours
- build and debug software to intelligently control mobile robotic systems using the Robot-C language
- design for unforeseen real-world circumstances, producing robust code and mechatronic designs which anticipate unknown changes in the robot's environment
- program a microcontroller board to automatically sample and control a variety of sensors and actuators
- design experiments which prove, characterize, extract and optimize performance parameters from each of their robot prototypes

4. Pre-requisite modules or specified entry requirements

CS230

8. Indicative content.

- Microcontroller programming using C
- Sensors and actuators
- Analogue circuit troubleshooting
- Robotic shell design using EV3
- Higher level programming for robotic control
- Mobile robotic algorithm design using Robot-C
- Mobile robotic control based on sensor data fusion

- Debugging and troubleshooting techniques
- Experimental characterization of robotic designs

1. Factual information

Module title	COMPUTER SCIENCE 333 – Computer Networks II		
Module tutor	Dr. Chatzistavros Vagelis	Level	6
Module type	Taught: Lecture/discussion	Credit Value	15
Mode of delivery	100% face-to-face		
Notional learning hours 150			

2. Rationale of the course within the degree scheme/Prerequisites/other entry requirements

This course is offered as a Level 6 Elective to students that have passed course CS322 (Computer Networks I). The course builds on the material learned in CS322 and expands the students' knowledge and understanding to more advanced and complex network concepts and designs. The focus is mainly routing and switching in LANs and WANs, but other major areas of networks like wireless, security and Broadband technologies. Completing CS322 and CS 333 a student is eligible to take the exam for the Cisco CCNA certification.

3. Aims of the course

The aims of the course are to expand students' knowledge in modern day networks. There are three primary goals; expand the strong foundations for Local Area Networks built in CS322, introduce Wide Areas Network Design and Technologies, and provide the fundamentals of trending networking areas including, but not limited to, wireless and security.

4. Pre-requisite courses or specified entry requirements

CSC 322

6. Indicative content

- Switching Fundamentals
- Advanced Routing Protocols
- Wide Area Network and Broadband Technologies
- Securing Connectivity
- Monitoring and Troubleshooting Networks
- Network security

1. Factual information			
Module title	COMPUTER SCIENCE 340: Introduction to Artificial Intelligence and Machine Learning using Python	Level	5
Module tutor	Dr. Alexander Astaras	Credit value	15
Module type	Taught: Lecture/Laboratory	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

The discipline of artificial intelligence (AI) is concerned with building intelligent algorithmic agents, computing algorithms which perceive their circumstances and take actions which maximize their chance of successfully achieving their goals. This includes machine learning algorithms, intelligent agents which modify their function based on past encounters with input data, under either supervised or unsupervised training circumstances.

This course is designed for intermediate programmers who can already code using structured programming and object-oriented methodologies (in any computing language), as well as have some intermediate level understanding of data structures, search and sorting algorithms.

3. Aims of the module

This course is an introduction to the field of AI and more specifically Machine Learning, including an intensive initial introduction to the Python programming language. Indicative AI topics covered include knowledge representation, problem solving via search, logical and probabilistic reasoning and machine learning algorithms such as decision trees, neural networks, reinforcement learning and genetic algorithms.

4. Pre-requisite modules or specified entry requirements

CS215

8. Indicative content.

Introduction to the Python programming language (approx. 3 weeks)

- syntax, strings basic I/O commands
- variables, conditionals and basic control flow
- loops
- collections: lists, dictionaries, tuples, sets
- functions, libraries
- file I/O

Selected topics in Artificial Intelligence and Machine Learning (approx. 3 weeks)

- informed and uninformed search
- constraint satisfaction
- reinforcement learning
- probability
- Bayes networks
- artificial neural networks
- genetic algorithms

1. Factual information			
Module title	COMPUTER SCIENCE 412 – Object Oriented Design Patterns	Level	6
Module tutor	Mr. Kaplanoglou Pantelis	Credit value	15
Module type	Taught: Lecture/guided discussion	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements	
<p>The module revisits Object Oriented application development methodology at the Senior level, examining its effectiveness in the life cycle of professional applications and software reuse through the adoption of Object Oriented Design Patterns. It presupposes the knowledge earned through the introductory line of the Programming Fundamentals programme thread and follows level 5 modules relating to Data Modelling (CS 312) and Systems Design (CS450) while specialising them within the context of Web Development. Currently CSC 325 (Distributed Systems) is a necessary prerequisite concerning web deployment technologies.</p>	
<p>The module mostly emphasizes the employment of OO concepts to Web Development yet it is of general enough nature for a level 6 module as the design patterns examined are applicable to a wide range of technologies and application domains.</p>	

3. Aims of the module	
<p>Major aims of the module are to teach object oriented modelling and reusable or ad hoc class design. The major features of OOP will be reviewed along with examples and their utility in building reliable applications will be discussed. UML will be used for modelling and connection of the formal models with language-specific class features will be elaborated. Design patterns will be examined in conjunction with OO language features and their utility in reliable application development will be thoroughly investigated.</p> <p>The module focuses around development of Web applications and the tools chosen will be suitable for this class of applications. The MVC design pattern will be presented and a development environment supporting the framework will be employed throughout the second half of the semester in order to examine object orientation in each aspect of the framework. MVC is a Design Pattern in itself.</p>	

4. Pre-requisite modules or specified entry requirements	
CS 215, CS 312, CS325 or permission by the Department.	

8. Indicative content.	

1. Review of Object-Oriented Concepts
2. Advanced Object-Oriented Concepts and introduction to Design Patterns
3. Design Patterns and Class Design guidelines
4. Mastering Inheritance and Composition
5. Frameworks and Reuse: Designing with Interfaces and Abstract Classes
6. Designing with Interfaces and Abstract Classes
7. Creating Object Models
8. Objects and Portable Data: XML and JSON
9. Persistent Objects: Serialization, Marshalling, and Relational Databases
10. Objects in Web Services, Mobile Apps, and Hybrids
11. Model, View and Controller issues of class design
12. SQL injection and similar cyber security threats

1. Factual information			
Module title	COMPUTER SCIENCE 421 – Computer Systems Security	Level	6
Module tutor	Mr. Menelaos Karamichalis	Credit value	15
Module type	Taught: Lecture/guided discussion	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

The module examines security issues in networked and non-networked environments. It is a senior-level module (level 6) for the Computing Systems programme thread. It follows CS321 – Operating Systems and CS322 – Network Operating Systems and Administration. The module provides the theoretical background and practical tools so that using the provided framework existing and future security threats can be analyzed and guarded against.

3. Aims of the module

The module aims at engaging a student’s critical thinking in analyzing security threat scenarios and providing mitigation strategies. The theoretical and practical background concerning issues of security in modern, networked and non-networked systems is provided first. Then, the state of the art approaches are covered, focusing on current best practices. The module relies heavily on case studies and articles on recent news, so it is both current and relevant.

4. Pre-requisite modules or specified entry requirements

CSC 321, CSC 322 or permission by the Department.

- 8. Indicative content**
- Securing data at rest and data in transit:
 - Essential Encryption Algorithms
 - Key management and credentials
 - Steganography and watermarking
 - Human psychology
 - Physical layouts
 - Network security (network segmentation, VLAN, VPNs, jump servers, firewalls)
 - Network Monitoring (HIDS, NIDS, CMDB, application-level firewalls, honeypots)
 - Compliance (regional laws affecting audits & network design)
 - Virtualization and security
 - System Security Policies and Best Practices
 - Threat models
 - Software development
 - Agile Development
 - Coding practices
 - Software release practices
 - DevOps
 - Cloud Computing security considerations

1. Factual information			
Module title	COMPUTER SCIENCE 422: Advanced DBMS		
Module tutor	Mr. Christos Christodoulou	Level	6
Module type	Taught: Lecture/discussion	Credit Value	7.5
Mode of delivery	100% face-to-face		
Notional learning hours 150			

2. Rationale for the module and its links with other modules	
<p>The module expands and deepens understanding of DBMS extending beyond a first year of exposure to the fundamentals. It extends CS312, which has provided the foundations on SQL and PL/SQL. Theoretical issues are examined with the intention of performance optimisation. The module aims at better equipping programme graduates towards the competitive DB market.</p>	
3. Aims of the module	
<p>This module focuses on creating and manipulating databases using SQL and PL/SQL programming languages for Oracle databases. Advanced query capabilities and procedural constructs are described using SQL and PL/SQL. The theoretical foundation for using these capabilities is presented. Performance issues are discussed including indexing, key definitions, and data constraints. The role of application development in ease of use, query optimization, and system performance is discussed. The module aims to teach students to use advanced SQL statements and PL/SQL programming features such as IF statements, Loops, Stored Functions/Procedures, Tables, Cursors, Stored Packages, Stored Triggers and creating and maintaining various databases. SmartDraw and Designer of Oracle is used for ERD's. APEX, SQL Plus and SQL Navigator, SQL Server Management Studio are used as user interface of the databases.</p>	
4. Pre-requisite modules or specified entry requirements	
CS 312	

6. Indicative content.											
<table border="1"> <tr> <td>ERD and UML Modelling</td> </tr> <tr> <td>Practical Database Design Methods and use of</td> </tr> <tr> <td>Indexing Structures for Database Files</td> </tr> <tr> <td>Practical Database Design and Tuning</td> </tr> <tr> <td>Database Security and Authorization</td> </tr> <tr> <td>Enhanced Data Models (Triggers)</td> </tr> <tr> <td>Distributed Databases and Client-Server Architectures</td> </tr> <tr> <td>Emerging Technologies: XML and Internet Databases</td> </tr> <tr> <td>Data mining Concepts</td> </tr> <tr> <td>Emerging Database Technologies & Apps</td> </tr> </table>		ERD and UML Modelling	Practical Database Design Methods and use of	Indexing Structures for Database Files	Practical Database Design and Tuning	Database Security and Authorization	Enhanced Data Models (Triggers)	Distributed Databases and Client-Server Architectures	Emerging Technologies: XML and Internet Databases	Data mining Concepts	Emerging Database Technologies & Apps
ERD and UML Modelling											
Practical Database Design Methods and use of											
Indexing Structures for Database Files											
Practical Database Design and Tuning											
Database Security and Authorization											
Enhanced Data Models (Triggers)											
Distributed Databases and Client-Server Architectures											
Emerging Technologies: XML and Internet Databases											
Data mining Concepts											
Emerging Database Technologies & Apps											

1. Factual information			
Module title	COMPUTER SCIENCE 443 - Capstone Project I	Level	6
Module tutor	Dr. Alexander Astaras	Credit value	15
Module type	Taught: Lecture/Laboratory	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

The module is the second half of a year-long capstone project, concluding the students' four-year learning experience in the Computer Science and Business & Computing study programs. It is designed to foster research, autonomy and synthesis of concepts and skills acquired in all other modules. The first half of the capstone project (CS443) is devoted to research/analysis and design, while the second semester counterpart (CS444) places emphasis on implementation, experimental validation, thesis writing and final project presentation.

3. Aims of the module

To provide students with an opportunity to work in a guided but increasingly independent fashion, to explore a particular problem in depth, to make practical use of principles, techniques and methodologies acquired elsewhere in the module. To challenge students to form a scientific thesis, carry out a sustained piece of individual work to prove or disprove it, and to present their project work in a dissertation. To enhance communication skills, both oral and written.

4. Pre-requisite modules or specified entry requirements

CS312, CS450

8. Indicative content.

The module deals with the definition and formal proposal phase of a major Computer Science and/or Business Computing IT project. It involves learning to brainstorm for project topic ideas, filter and prioritize them, develop a thesis statement, build a brief project proposal description, seek a faculty member to act as a supervisor for the capstone project, write a thesis proposal, design a poster and deliver a final slide presentation outlining the proposed project.

1. Factual information			
Module title	COMPUTER SCIENCE 444 - Capstone Project II	Level	6
Module tutor	Project supervisors (various faculty)	Credit value	15
Module type	Taught: Lecture/Laboratory	Notional learning hours	150

2. Rationale of the module within the degree scheme/Prerequisites/other entry requirements

The module is the second half of a year-long capstone project, concluding the students' four-year learning experience in the Computer Science and Business & Computing study programs. It is designed to foster research, autonomy and synthesis of concepts and skills acquired in all other modules. The first half of the capstone project (CS443) is devoted to research/analysis and design, while the second semester counterpart (CS444) places emphasis on implementation, experimental validation, thesis writing and final project presentation.

3. Aims of the module

To provide students with an opportunity to work in a guided but increasingly independent fashion, to explore a particular problem in depth, to make practical use of principles, techniques and methodologies acquired elsewhere in the module. To challenge students to form a scientific thesis, carry out a sustained piece of individual work to prove or disprove it, and to present their project work in a dissertation. To enhance communication skills, both oral and written.

4. Pre-requisite modules or specified entry requirements

CS443

8. Indicative content.

The module deals with the development, experimental validation, thesis document writing and presentation phase of a major Computer Science and/or Business Computing IT project. It involves learning to brainstorm for project development solutions, filter and prioritize them in consultation with the project supervisor, organize and implement development, troubleshoot, build a proof-of-concept prototype, plan validation experiments, write a thesis document which provides a scientific literature context and outlines originality, design a poster and deliver a final slide presentation describing all capstone project work.

1. Factual information			
Module title	COMPUTER SCIENCE 450 - Systems Analysis and Design		
Module tutor	Mr. Chris Christodoulou	Level	5
Module type	Taught: Lecture/discussion	Credit Value	15
Mode of delivery	100% face-to-face		
Notional learning hours	150		

2. Rationale for the course and its links with other courses

The course serves as a capstone for the Junior year, aiming to unify all prior knowledge from the viewpoint of Systems Analysis and Design. As students have already been exposed to various aspects of computer systems, the course introduces them to the development life cycle of real-world, professional applications.

The course introduces the waterfall model for system/application development and the formal tools employed in its various stages with emphasis in prototyping, system implementation and testing.

Students entering the course are expected to have adequate knowledge of data modelling (CS205, CS312), hardware (CS310). The knowledge acquired in this course, will be applied on CS443 and CS444 where students are required to research and create a capstone project.

3. Aims of the course

The course introduces the waterfall model for system/application development and the formal tools employed in its various stages. The objectives of the course are to:

- Provide formal tools for functional and non-functional requirements collection and documentation (ERD, UML, DFD, STD's)
- Define the role of the systems analyst and designer.
- Build project management and interpersonal communication skills that the system analyst must have.
- Explain the methodologies that are used for systems analysis and design.
- Follow through the waterfall model (and discuss deviations therefrom), presenting the relevant tools at each stage.
- Provide the problem solving background for resolving trade-offs inherent in design.
- Present principles of quality and correctness testing.
- Provide students the opportunity to work as a team of analysts and designers in a project to implement the taught methodologies.

Students develop technical, analytical and business skills that support the pursuit of professional careers and advanced computer science studies.

4. Pre-requisite courses or specified entry requirements

CS312, CS310 or permission by the Department.

8. Indicative content

The course main focus is Systems analysis where students learn problem-solving techniques to decompose a system into its component pieces for the purpose of studying how well those component parts work and interact to accomplish their purpose.

- Scope analysis and feasibility study.
- Requirements collection and formal tools (ERD, UML, DFD, STD).
- Design: trade-off analysis and formal tools for decision making.
- Implementation phase: actions and deviations from Waterfall model.
- Unit and System testing.
- Deployment and Maintenance.
- Project management throughout the cycle.

1. Factual information			
Module title	MATHEMATICS 120 – Calculus I		
Module tutor	Dr. Vassilis Daskalogiannis	Level	4
Module type	Taught: Lecture/discussion	Credit Value	15
Mode of delivery	100% face-to-face		
Notional learning hours: 150			

2. Rationale of the course within the degree scheme/Prerequisites/other entry requirements
<p>The purpose of this course is to give a solid foundation in Calculus concepts, tools and techniques for the student entering Science and Engineering fields.</p> <p>The course covers definition, calculation, and major uses of the derivative, as well as an introduction to integration. Topics include limits; the derivative as a limit; rules for differentiation; and formulas for the derivatives of algebraic, trigonometric, and exponential/logarithmic functions. Also discusses applications of derivatives to motion, density, optimization, linear approximations, and related rates. Topics on integration include the definition of the integral as a limit of sums, anti-differentiation, the fundamental theorem of calculus, and integration by the u-substitution and substitution by parts.</p> <p>The course emphasizes conceptualization, modelling, and skills. All three are of great importance. There is a concentration on multiple ways of viewing functions, on a variety of problems where more than one approach is possible, and on student activity and discussion.</p>

3. Aims of the course
<p>By the completion of this course it is the aim and hope of the mathematics faculty that students who work hard and apply themselves will be able to:</p> <ul style="list-style-type: none"> • help students acquire a solid foundation in Differential and Integral calculus skill and concept, in order to prepare them for other university courses that require such knowledge • show to students how math can model and solve authentic real world problems • enable students to develop problem-solving skills, fostering critical thinking, within a varied and interesting setting • encourage modelling and use of mathematics in Science and Engineering fields

4. Pre-requisite courses or specified entry requirements
MATH 101

8. Indicative content
<p>Functions and Models</p> <p>Limits and Derivatives</p> <p>Differentiation Rules</p> <p>Applications of Differentiation</p> <p>Integrals</p> <p>Integration by parts</p>

1. Factual information			
Module title	MATH 220 – Discrete Mathematics		
Module tutor	Dr. Kanakoglou Konstantinos	Level	5
Module type	Taught: Lecture/discussion	Credit Value	15
Mode of delivery	100% face-to-face		
Notional learning hours 150			

2. Rationale for the course and its links with other courses

Discrete mathematics can be defined as the study of structures consisting of a sequence of individual, separated steps. As such, they contrast with calculus, the latter describing processes which vary continuously or smoothly. If one can claim that the ideas of calculus were fundamental to the industrial revolution, then one can safely assume that the backbone of the science and technology of the computer age is discrete mathematics.

The purpose of this course is for the students to understand and use the aforementioned discrete backbones of computer science. In particular, this class is meant to introduce logic, proofs, sets, relations, functions, counting, and probability, with an emphasis on applications in computer science.

Further, this course will cover fundamental mathematical foundations required for conceiving, proving, and analyzing algorithms.

Upon completion of this course, it is the aim and hope of the mathematics faculty that students who work hard and apply themselves will be able to:

1. Become comfortable with mathematical thinking that allows them to write clean and logical proofs.
2. Learn the basic language and problem solving tools of discrete mathematics so they can use these in their further study.
3. Develop problem-solving skills, fostering critical thinking, within a varied and interesting setting.
4. Model and analyze computational processes using analytic and combinatorial methods.

3. Aims of the course

On completion of this course, students will be able to apply the methods of discrete (non- continuous) mathematics in Computer Science. They will be able to use these methods in subsequent Computer Science courses such as the design and analysis of algorithms, computability, and computer systems.

4. Pre-requisite courses or specified entry requirements

MATH 101, CS 105

8. Indicative content

The Foundations: Logic, Sets and Functions
 Basic Structures: Sets, Functions, Sequences, Sums and Matrices
 Algorithms
 Number Theory and Cryptography
 Induction and Recursion
 Counting
 Discrete Probability
 Advanced Counting Techniques
 Relations
 Graphs

1. Factual information			
Module title	STATISTICS 210 – Introduction to Statistics with R		
Module tutor	Dr. Mengoudi Kyriaki	Level	4
Module type	Taught: Lecture/discussion	Credit Value	15
Mode of delivery	100% face-to-face		
Notional learning hours 150			

2. Rationale of the course within the degree scheme/Prerequisites/other entry requirements

An application-oriented introduction to modern statistical analysis using R software that includes: study design, exploratory data analysis; random variables; probability models and sampling distributions; point and interval estimates; hypothesis tests and linear regression. A wide variety of applications are used.

3. Aims of the course

This course is an introduction to descriptive and inferential statistical methods. Students will master problem solving using both manual computations and R statistical software. The student will learn to formulate research questions, design data collection to answer the question, collect and analyse the data and interpret and report the results.

The course aims at the following:

- Give the student the statistical skills necessary to meet the needs of business and the real- world decision-making problems.
- Effectively communicate the results of a statistical analysis both orally and in writing.
- Provide fundamental statistical knowledge and skills required for a higher-level course in related fields.
- Analyze data using R software

4. Pre-requisite courses or specified entry requirements

Math 101

8. Indicative content.

Data and Statistics
Descriptive Statistics
Descriptive Statistics: Numerical Measures
Probability
Discrete Probability Distributions
Continuous Probability Distributions
Sampling Distributions
Interval Estimation
Hypothesis Testing
Linear Regression

8. STUDENT SUPPORT, GUIDANCE AND ADVICE

ACADEMIC ADVISING AND MENTORING

During the first 2 weeks of classes, students will be assigned the faculty member who will be their Academic Advisor for their first year at ACT and, in most cases, until graduation. The Academic Advisor helps students to plan their overall program of studies, as well as to select courses each semester. Advisors also provide information about ACT academic and support services, assist students in addressing problems in particular courses within a given semester, and offer ongoing advice concerning the students' long-term academic and career goals.

THE LEARNING HUB

ACT's Learning Hub is located on the upper floor of the Library. Students are invited to meet with the tutors and receive assistance with their English language and Mathematics needs. The Admissions and the Enrollment department offices are also located on the upper floor of the Library. Prospective students may obtain from there all the necessary information concerning their studies and financial support.

HEALTH SERVICES

The resident Anatolia High School nurse accepts student emergency visits as well as regular appointments in her office located on campus. The Anatolia/ACT is also within easy access to both a private clinic and several hospitals, all of which provide emergency services. A resident doctor at ACT is available to students on campus during the doctor's office hours.

9. OPPORTUNITIES FOR PERSONAL DEVELOPMENT PLANNING

CAREER SERVICES AND GUIDANCE

The Career Office equips students with the tools and know-how to successfully kickstart their careers. Specifically, we provide hands-on training on interviewing, resume & cover letter writing. The above are subsidized by guest lectures, which help students get an insight in their field of interest, and company visits, which give students the opportunity to meet company representatives and conduct informational interviews. The office also provides individual consultations in any career related issue, including job search tactics, career planning, resume preparation and mock interviews. Apart from this preparation stage, the career office offers internship and job opportunities, both in Greece and abroad, accessible to all students via the Career Office's job board (jobs.act.edu).

In addition, the Career Office also organizes the annual Career Week, a week-long event with lectures, company presentations, on-campus recruiting, networking opportunities and hands-on workshops.

The Office also serves as Liaison with the Business Community, and works actively with Industry and Academia to identify placement opportunities and keep students informed of local and regional trends. Leading representatives from the private and public sectors visit ACT regularly as guest speakers in classes and events, reinforcing ACT's strong ties with companies, institutions and organizations throughout the local, national and international business environment.

INTERNSHIPS

The Career Office gives special emphasis to students' internships, both with local and international organizations. Every semester there is a visiting program with local organizations with internship opportunities, all relevant to the academic programmes and focusing on enhancing in-class learning. Apart from the on-campus recruiting, we coordinate a series of internship opportunities with organizations located in other cities or even abroad, either with on-site placement or remote work. The school's Job Board and Facebook Career Group help disseminate the internship opportunities, while the Career Workshops (resume writing, cover letter preparation, mock interviews) prepare students for claiming those opportunities.

10. OPPORTUNITIES AND SUPPORT FOR STUDY ABROAD

ACT STUDENTS STUDYING IN THE US

ACT has signed a number of study abroad exchange agreements with partner colleges and universities that enable students to spend a semester studying in the US. Through these agreements, ACT students may spend a semester, normally in their second or third year of study, at a college in the US, and upon return to ACT receive full transfer credit for all courses successfully completed while abroad. Agreements with partner schools allow ACT students to enroll at collaborating institutions while continuing to be enrolled at the home school and pay tuition and fees at ACT. Students are encouraged to look into study abroad opportunities early in their academic career with the director of the I.P.O. Good academic standing is a pre-requisite for considering such a possibility.

11. WORK PLACEMENT INFORMATION

A number of opportunities for personal development are available to English majors on demand and on a voluntary basis within the program, ranging from opportunities for joining extracurricular activities and clubs on campus, serving on the Student Government Association, engaging in Service Learning, acquiring information literacy and CV writing skills to building professional expertise through a term's Internship training in their senior year, thus enhancing self-reflection, PR and communication skills, personal and professional responsibility, learning how to meet deadlines and working with others, etc.

In addition, through ACT's Careers Office, targeted English-specific workshops, company visits and presentations further enhance opportunities for personal development. Short term internships have also been introduced. Here are some concrete examples:

- Communications & copywriting internship opportunities
- Resume, cover letter and interview workshops
- Private consultations for preparing individual students' resumes & cover letters.
- Mock interviews - private advising on interviewing
- Job board which renews weekly and includes - among others - internship and entry level opportunities in the field of communication.
- Guest lectures of professionals in the field (media, teaching, etc.)

The programme does not require students to undertake a compulsory work placement but there is the option of internship, if they wish to do so. Through the internship module students will be able to make visible connections between community service, their own learning, personal and professional development and citizenship skills, values and practices.

12. FACILITIES AND SERVICES

BISELL LIBRARY - GENERAL INFORMATION

The Bissell Library offers a vast collection of books in print, electronic books, videos, and DVDs. Already one of the largest English language libraries in Greece, its collection is rapidly growing into a space designed to accommodate the institution's needs for years to come. The collection includes subscriptions to periodicals in hard copy as well as access to numerous full-text scholarly journals, magazines, and newspapers.

The Bissell Library offers, to currently enrolled students, on site and remote access to research databases to support inquiry and research. Business databases include Ebsco's Business Source Elite, Regional Business News, Hoover's, and ProQuest ABI Inform Global. Databases for research include: Academic Search Premier, E-books collection, Encyclopaedia Britannica, ERIC, GreenFILE, Columbia International Affairs Online (CIAO), JSTOR, Oxford English Dictionary and Oxford Music Online. We also subscribe to the Ebsco A-Z service, providing listing of all the electronic resources accessible from the library. The EBSCO Discovery Service TM brings together the most comprehensive content providing to users an easy, yet powerful means of accessing all of the library's information resources through a single search.

The Bissell Library shares an integrated library management system with the Socrates Eleftheriades and Olga Mavrophidou-Eleftheriades Library of Anatolia College. Access to both collections is available through the web-based library catalog. Library users can search the catalog, databases or the Internet

through public access terminals available on both floors of the library. Network ports are available for laptops and the entire building is Wi-Fi enabled.

COMPUTING SERVICES & NETWORKING FACILITIES

ACT has state of the art computer infrastructure and facilities. All computer facilities are connected to a high-speed campus network, which is based on fiber optic cables connecting all buildings. In addition a large high speed wireless network access (WI-FI) covers large areas of the campus giving students the ability to use the school's resources or access the internet on their laptop.

The computer facilities include the Stavros S. Niarchos Technology Center in Bissell Library and a number of other computer laboratories located in various buildings. Many high-speed servers are present in the network infrastructure, along with Intel® Core™ latest technology workstations, connected to the Internet, available to students in multiple laboratories.

The laboratories are used both as general access and instructional computer labs. They are equipped with data projectors and black and white or color laser printers. All stations are networked with full Internet Access and run the latest software such as MS office, Oracle, Power-builder, Visible Analyst, Java, Visual Basic, 3-D Max, Adobe Photoshop, Adobe CS Production Studio Premium, Macromedia Studio, Macromedia Authorware, PanaView Image Assembler, Mathematica Player, Minitab, MathCad, Daedalus, etc.

SCIENCE LABORATORIES

All science courses are accompanied by laboratory work. The purpose of the laboratories offered is to expose students to hands-on experience regarding concepts and principles learned in classroom. The College's new Science Facilities are located in the ground floor of Constantinidis Hall. The facilities include three laboratories (Biology/Ecology, Physics, Chemistry) covering a total area of 300 square meters.

FOOD SERVICES

The ACT Cafe, rented on a contract to a professional food service provider, is also located in the Constantinidis Hall and operates weekdays from 10:00 - 18:30 (Fall – Spring semesters) and 11:00 - 14:00 (Summer term—hours flexible) when classes are in session. The cafe offers an assortment of cold and hot sandwiches, coffee, salads and beverages.

HOUSING

ACT housing is available on a first come, first served basis and priority is always given to freshmen. Apartments are all shared and have both single and double rooms, common area, kitchen and bathroom. All interested students must complete a Housing Application in order to be considered for on-campus housing.

The Student Services Coordinator will assist students in locating off-campus housing in local residential areas. A list of trusted real estate agencies and property owners who speak English will be made available for interested students. ACT does not have any official relationship with housing agencies and does not endorse any specific agency. With all off-campus housing, students are responsible for personally contracting with the landlord but ACT will provide guidance and assistance. Regular announcements about available flats around the city are made on the ACT housing Facebook group. Students can also refer to this Facebook group in order to find roommates or shared housing.

13. DETERMINATION OF RESULTS ([LINK TO OU REGULATIONS](#))

MINIMUM REQUIREMENTS FOR PASS

To obtain an Open University award students are required to complete all parts of the programme's approved assessment and comply with all regulations relating to their programme of study.

The minimum aggregate pass marks for The Open University validated awards are:

- 40% for undergraduate programmes
- 50% for postgraduate programmes

These minima apply to assessments, modules, stages and qualifications.

DETERMINING MODULE OUTCOMES

The overall module mark or grade shall be determined as set out in the assessment strategy detailed in the module specification and published in the Programme Handbook.

A student who passes a module shall be awarded the credit for that module. The amount of credit for each module shall be set out in the programme specification and published in the Programme Handbook.

In order to pass a module a student must achieve the requirement of the module as set out in the module specification and published in the Programme Handbook.

Where a student is registered only for a module (rather than a qualification) the resit will apply.

BACHELOR HONOURS DEGREE CLASSIFICATION

Classification of bachelor degrees will be based on the average mark across all modules within Stage 3 (usually Credit Level 6) and Stage 2 (usually Credit Level 5) at a ratio of 2:1 respectively unless the requirements of a Professional, Statutory and Regulatory Body (PSRB) state otherwise.

Honours degrees are classified as:

- First class Aggregate mark of 70% or above
- Upper Second class Aggregate mark between 60% and 69%
- Lower Second class Aggregate mark between 50% and 59%
- Third class Aggregate mark between 40% and 49%

Where students have directly entered a Qualification Level 6 top-up award (e.g. having previously undertaken a Higher National Diploma (HND) or Foundation Degree (FD) award) the calculation for the honours classification will be based solely on all credits at Credit Level 6.

Performance in work for which an award of credit for prior learning has been made is not taken into account in the calculation of the final award.

Where the final result of the classification calculation creates a mark of 0.5% or greater this will be rounded up to the next full percentage point (e.g. 69.5% is rounded to 70; 59.5% to 60%; and so on). Where the calculation creates a mark below 0.5% this will be rounded down to the next full percentage point (e.g. 69.4% is rounded to 69%; 59.4% to 59%; and so on). For the purposes of rounding up or down, only the first decimal place is used.

14. **ASSESSMENT AND PROGRESSION REGULATIONS** ([LINK TO OU REGULATIONS](#))

SUBMISSION OF ASSESSED WORK

Work submitted for a summative assessment component cannot be amended after submission, or re-submitted.

Student requests for extensions to assessment deadlines will not be approved unless made in accordance with published partner institution guidelines as approved by The Open University.

Where coursework is submitted late and there are no accepted extenuating circumstances it will be penalized in line with the following tariff:

Submission within 6 working days: a 10% reduction for each working day late down to the 40% pass mark and no further.

Submission that is late by 7 or more working days: submission refused, mark of 0. A working day is defined by the partner and submission after the deadline will be assumed to be the next working day.

ASSESSMENT SCORES

All undergraduate assessment will be marked on a percentage scale of 0-100.

% Scale Score	Performance Standard
70+	Excellent pass
60-69	Very Good pass
50-59	Good Pass
40-49	Pass
0-39	Fail

The final grade for an individual assessment component will be determined after completion of a quality assurance process (e.g. moderation, remarking) as detailed in the partner institution's OU approved policy for moderation.

Where the result of the assessment calculation creates a mark of 0.5% or greater this will be rounded up to the next full percentage point (e.g. 69.5% is rounded to 70; 59.5% to 60%; and so on). Where the calculation creates a mark below 0.5% this will be rounded down to the next full percentage point (e.g.

69.4% is rounded to 69%; 59.4% to 59%; and so on). For the purposes of rounding up or down, only the first decimal place is used.

DETERMINING PROGRESSION AND QUALIFICATION OUTCOMES

The paths through which students are required to progress through the programme, and the elements identified as compulsory or optional, are set out in the programme specification and approved in the validation process.

Pathways through any programmes offered by partner institutions may require students to complete prerequisite or co-requisite modules.

STAGE REQUIREMENTS

Each of the stages of an undergraduate programme is expected to consist of a total of 120 credits.

In cases where Programmes are not divided into stages (for example, most postgraduate programmes and undergraduate programmes of 120 credits or less) the provisions below apply to the whole programme.

In order to complete and pass a stage of a programme, a student must acquire the total credit set out in the programme specification at the approved qualification level for the award, or have been exempted through advanced standing, or through the implementation of the processes covering extenuating circumstances (see Section F).

The credit value of each module contributing to a stage determines its weighting in the aggregation of credit for a stage.

Where a student fails a module, the following may apply in the first instance:

- Resit, a second attempt at an assessment component following a failure at first attempt.
- Compensation, the award of credit by the Board of Examiners for a failed module(s) on account of good performance in other modules at the same credit level where the learning outcomes have been met.

RESIT PROVISION

Resit provision is subject to all the following conditions:

- The maximum number of retakes allowed in a programme leading to an Open University award is 10.
- A student who does not complete the resit by the date specified shall not progress on the programme, except in cases where the process for allowing extenuating circumstances has been followed.
- Resits can only take place after the meeting of the Board of Examiners or following agreement by the Chair and the External Examiner of the Board.
- A student who successfully completes any required resits within a module shall be awarded the credit for the module and the result capped at the minimum pass mark for the module.

15. DISSERTATIONS AND PROJECTS

HOW THE BISSELL LIBRARY SUPPORTS STUDENTS WITH THEIR THESES

The Bissell library in its effort to compliment and facilitate the educational process and the curriculum, always strives to provide the best services possible safeguarding that all students across all Divisions receive the same learning experience and support, as well as fair access to all available resources and services. More specifically when it comes to the writing up stage of their theses, the Bissell Library provides the following Resources and Support Services.

Resources :

- A collection of thesis writing books, academic writing and general study skills. This collection is housed in the Learning Hub
- Subject guides corresponding to the Divisions also provide a thematic information gateway of trustworthy information resources (e-journals, databases, etc.) to assist students in their quest for information gathering for their theses.
- The website of the Library, attempts to familiarize and provide introductory guidance on how to use various online tools such as Zotero, EasyBib and Diigo, which can become indispensable tools when it comes managing information and bookmarks, and citing resources.
- Guides have been composed to assist students with the OU Harvard referencing style (available on the library website)
- The Library in conjunction with all Divisions has streamlined the procedure of gathering all theses so as to develop in the near future an Institutional Repository. In the meantime students can drop in any time and can have access to study good examples of theses on site.
- Last but not list a plethora of printed material in a variety of subjects is also available to students.
- The students are able to ask for Interlibrary loans of journal articles we do not have online access to. The service is provided by the British Library. They are also able to suggest books relevant to their research to the library to buy.

Support Services:

- The Learning Hub strives to help students to become strong, independent learners through a variety of services. The assistance stretches through all stages of the learning process whether it is revising thesis writing, math, reading, or just refining those digital skills. Consultation is offered on one to one basis on prearranged meetings, or drop in sessions, as well as via e-mail.
- The allocation of two Academic Liaison Librarians as well as Division Academic staff reps allow for better communication among students, academic staff and the Library. This helps students and academic staff to convey in a more efficient manner their needs to the Library.
- Information Literacy lessons conducted by the Academic Liaison Librarians offer students a reminder of research, evaluation and referencing skills to support them in their project. This come as a continuation of previous Information literacy teaching offered in the previous years of study, more specifically in English 101 which includes 6-Information literacy sessions. These sessions include: techniques on how to locate formal and informal information sources on subjects, evaluation of resources, plagiarism and paraphrasing, and referencing.
- Finally the Academic Liaison Librarians provide one to one support on how to search, evaluate and locate materials and referencing, either in drop in sessions or after appointments.

16. OTHER INSTITUTIONAL POLICIES AND REGULATIONS ([LINK TO OU REGULATIONS](#))

ACADEMIC MISCONDUCT

Academic misconduct is defined as any improper activity or behavior by a student which may give that student, or another student, an unpermitted academic advantage in a summative assessment. In investigating and dealing with cases of suspected misconduct, partner institutions will follow the policies and processes approved at Institutional approval or review.

The following is a non-exhaustive list of examples of academic misconduct which will be considered under these Regulations:

- Plagiarism: representing another person's work or ideas as one's own, for example by failing to follow convention in acknowledging sources, use of quotation marks, etc. This includes the unauthorized use of one student's work by another student and the commissioning, purchase and submission of a piece of work, in part or whole, as the student's own.

Note: Where a student has an acknowledged learning disability, a proof-reader may be used to ensure that the student's meaning is not misunderstood as a result of the quality and standard of writing, unless a partner institution policy specifically prohibits this. Where permitted, a proof-reader may identify spelling and basic grammatical errors. Inaccuracies in academic content should not be corrected nor should the structure of the piece of work be changed.

- Collusion: cooperation in order to gain an unpermitted advantage. This may occur where students have consciously colluded on a piece of work, in part or whole, and passed it off as their own individual efforts or where one student has authorized another to use their work, in part or whole, and to submit it as their own.

Note: legitimate input from tutors or approved readers or scribes is not considered to be collusion.

- Misconduct in examinations (including in-class tests).

ACADEMIC OFFENCES

An academic offence (or breach of academic integrity) includes any action or behavior likely to confer an unfair advantage, whether by advantaging the alleged offender or by disadvantaging another or others. Examples of such misconduct are plagiarism, collusion, cheating impersonation, use of inadmissible material and disruptive behavior. Responsibility for reviewing breaches of academic integrity is held by the college's Academic Standards and Performance Committee (AS & PC).

Charges against a student for violating academic integrity may originate from any source: a faculty member, an administrator, a staff member, a fellow student, or from the community at large. The charges are to be submitted in writing to the chair of the AS&PC. If a member of the Committee originates the charge, then that member will be excluded from the decision-making process, and any other process related to the case.

On receipt of the allegation of a breach of academic integrity, the Chair of the AS&PC must inform the Chair of the Board of Examiners that is responsible for the assessment of the course(s) that are affected by the alleged offence. The Board should then suspend its decisions on the candidate's grade(s) until the facts have been established.

The AS&PC will either itself investigate the charge or establish from its own membership a panel to conduct the investigation. In establishing whether a breach of academic integrity has occurred, the Committee (or panel) should consider oral and/or written evidence supplied by the individual(s) making the charge and the alleged offender. The alleged offender shall have the right to appear before the Committee (or panel).

Once the AS&PC has considered the allegation and reached a conclusion on whether an offence has occurred, it should issue a report with a recommendation regarding the outcome for the student to the Chair of the relevant Board of Examiners. If it has been established that an offence has occurred, the Board will judge the significance of the misdemeanor and exercise its discretion as appropriate to the case. If it is established that a student has attempted to gain an unfair advantage, the examiners shall be given the authority to rule that the student has failed part or all of the assessments, and the authority to determine whether or not the student should be permitted to be reassessed.

Independently on the assessment decisions made by the Board of Examiners, the AS&PC is empowered to consider a wider range of sanctions that might be applied when a student is found guilty of a breach of academic integrity. The following list of sanctions is indicative and can be imposed by majority vote of the Committee:

- Admonishment Letter (or Letter of Warning): The student is advised in writing that her/his behavior violates rules of academic and/or personal integrity and that a recurrence will lead to more serious sanctions. The Committee will deliberate on whether the letter should or should not appear in the student's file permanently or for a lesser period of time.
- First Offense File: The student's name and a description of the offense is filed in a shared electronic folder, accessible by the Provost, department chairs and area coordinators. Second offenses automatically result in a hearing.
- Disciplinary Probation: The student is advised in writing that his/her behavior violates rules on academic and/or personal integrity and is given a probationary period (to be decided upon by the Committee) to show by good behavior that a more stringent penalty should not be imposed. During the period of the probation, the student is required to terminate association with all extra-curricular activities and resign from any student office.
- Suspension: The student's relationship with the College will be discontinued until the end of the semester or term. The student will forfeit any fees involved with the College.
- Dismissal: The student's relationship with the College will be terminated indefinitely. The right to apply for re-admission shall be denied.

Within five working days of receipt of the decision, either party (plaintiff or student) has the right to make a formal written appeal against the decision of the Committee. The appeal is addressed first to the AS&PC. If the Committee does not deem any change to the decision is warranted subsequent to consideration of the appeal, the appeal may then be brought to the Academic Council, and subsequently to the President whose decision is final.

EXTENUATING CIRCUMSTANCES

The Open University recognizes that students may suffer from a sudden illness, or other serious and unforeseen event or set of circumstances, which adversely affects their ability to complete an assessment, or the results they obtain for an assessment. In such cases the partner institution's extenuating circumstances procedures will be applied, as approved in institutional review.

A student who is prevented from attending or completing a formal assessment component or who feels that their performance would be (or has been) seriously impaired by extenuating circumstances, may submit a deferral request to the AS&PC.

MITIGATING CIRCUMSTANCES

The following regulations distinguish between factors or circumstances which were known to the student in advance of taking an assessment and which affect his or her ability to attend an examination or submit work by the published deadline, and those which have not impaired the student's ability to attend for examination or meet a deadline for the submission of work but which may have affected his or her performance. In all cases, it is the responsibility of the student to ensure the timely disclosure of any factors or circumstances which may affect the assessment of his or her learning and responsibility for the consideration of these factors and circumstances will lie with the AS&PC.

Students whose circumstances may affect (or may have affected) their ability to meet a program's assessment requirements must submit a completed Mitigating Circumstances Extension Form together with verifiable documentation to the Registrar's Office. This form can be completed electronically or in person and may, if necessary, be signed retrospectively.

In the case of factors or circumstances which were known to the student in advance of taking an assessment and which affect his or her ability to attend an examination or submit work by the published deadline:

- the AS&PC will consider the evidence submitted by the student;
- if the mitigating circumstances are accepted by the Committee it will determine the extension to be granted to the student or, in the case of examinations, the date on which the student shall be assessed; in such cases the grades will not be capped at 40%.
- the Chair of the Board of Examiners, the Registrar and the appropriate department head/area coordinator will be informed of the Committee's decision.
- The student will have the right to apply for a further extension, or for a rescheduling of an examination, if the mitigating circumstances persist.

In the case of factors or circumstances having prevented a student from attending for examination or meeting a deadline for the submission of work but which may have affected his or her performance:

- the AS&PC will review the evidence submitted by the student and make a recommendation for consideration by the appropriate Board of Examiners;
- the Board of Examiners is responsible for considering that action that it should take in the light of the recommendations of the AS&PC;
- the actions available to the Board of Examiners include: the deferral of an assessment to a later date; compensation for the failure in a course; agreement that the student should either retake the course or be reassessed with the grade achieved being recorded in the student's transcript and therefore

contributing to the classification of the award; and, exceptionally a decision that the student be assigned a higher grade for the course or courses on which his or her performance has been affected.

Students are responsible for ensuring that the partner institution is notified of any extenuating circumstances at the time they occur and for supplying supporting documentation by the published deadline.

If a student is unable to attend an examination or other assessment event because of extenuating circumstances, they must inform the Partner institution as soon as possible and provide supporting evidence before published deadlines or within 7 calendar days, whichever is sooner. If a student cannot submit evidence by published deadlines, they must submit details of the extenuating circumstances with an indication that evidence will be submitted within 7 calendar days.

Medical evidence submitted in support of a claim for extenuating circumstances should be provided by a qualified medical practitioner.

Upon receipt of recommendations from the panel or body responsible for investigating extenuating circumstances, the Board of Examiners, or its subsidiary board, will decide whether to:

- provide a student with the opportunity to take the affected assessment(s) as if for the first time i.e. a 'sit' or 'submit', allowing them to be given the full marks achieved for the examination or assessment, rather than imposing a cap;
- waive late submission penalties;
- determine that there is sufficient evidence of the achievement of the intended learning outcomes from other pieces of assessment in the module(s) for an overall mark to be derived;
- note the accepted extenuation for the module(s) and recommend that it is taken into account at the point of award and classification.

The Board of Examiners, depending on the circumstances, may exercise discretion in deciding on the particular form any reassessment should take. Options are a viva voce examination, additional assessment tasks designed to show whether the student has satisfied the programme learning outcomes, review of previous work, or normal assessment at the next available opportunity. The student will not be put in a position of unfair advantage or disadvantage: the aim will be to enable the student to be assessed on equal terms with their cohort.

The module marks released following the meeting of the Board of Examiners should clearly identify results where extenuation has been considered and applied.

If a student fails, without good cause, to provide the responsible body with information about extenuating circumstances within the timescales specified in the partner institution policy, the responsible body has authority to reject the request on those grounds.

17. STUDENT PARTICIPATION AND EVALUATION

STUDENT PARTICIPATION

A member of the Student Government Association (SGA) must be present at all meetings of the Academic Standards and Performance Committee (ASPC) of the college, and participate in the discussions and voting for all cases examined. Furthermore, there are scheduled weekly meetings between the Associate Dean of Students and the SGA, where students present their views on the operation and development of the College. In addition, ACT may invite students to Academic Council meetings, where they can express their views and opinions to the top-level administration. The SGA is also involved in co-organizing major on-campus events.

MODULE EVALUATION

Module evaluation is conducted through the student evaluation forms. These forms measure the teaching quality and assessment methods, learning materials, delivery methods, course objectives, thought-provoking activities, comprehension of the subject matter, grading, degree of intellectual challenge and stimulation and draw comparisons with other courses.

The collection of student feedback is made at the office of the Associate Dean of Students, who has the general overview of the procedure. Then, modules are classified according to the programme they belong, and the feedback is sent to the corresponding department head. The results are also forwarded to the individual instructors. The outcomes of module evaluations are discussed between the Associate Dean for Academics and the department heads, and also in the departmental meetings of all divisions. In all these meetings, measures that need to be taken to improve student experience in future offerings of the modules are discussed.

COLLEGE-WIDE FEEDBACK

At the college level, ACT is administering another survey to measure both educational and other aspects of student life and behaviors. The survey, named College Student Experiences and Learning Outcomes (CSELOA) is aiming at measuring self-perceptions of students and has two parts. The first part measures student learning outcomes and the second measures student behaviors and experiences. The questionnaire includes a diverse spectrum of variables relating to academics, faculty, student services, student-to-student and student-to-faculty interactions, sense of community, use of campus facilities, academic skills, communication, after-college preparation for graduate studies or work, and off campus study and life behaviors.

18. GENERAL READING LIST (NOT MODULE SPECIFIC), INCLUDING ELECTRONIC RESOURCES

- Fundamentals of data structures in C++ /; Horowitz, Ellis; 1995; Bissell Library, General Stacks (005.73 HOR)
- Cryptography's role in securing the information society / Kenneth W. Dam and Herbert S. Lin, editors.; 1996; Bissell Library, General Stacks Upper Level (652.8 CRY)
- Data structures and algorithms /; Aho, Alfred V.; 1983; Bissell Library, General Stacks (005.73 AHO)
- Fundamentals of computer algorithms /; Horowitz, Ellis; 1978; Bissell Library, General Stacks (519.4 HOR)
- Web-teaching : a guide to designing interactive teaching for the World Wide Web /; Brooks, David W.; 1997; Eleftheriades Library (025.06 BRO)
- Principles of database and knowledge-base systems /; Ullman, Jeffrey D.; 1988; Eleftheriades Library (005.74 ULL)
- Research methods in political science : an introduction using MicroCase.; Corbett, Michael; 2003; Bissell Library, Reference Desk (320.0285 COR)
- Young scientist;; 1999; Eleftheriades Library (CD 503 YOU)
- Understanding information systems : what they do and why we need them /; Ratzan, Lee; 2004; Bissell Library, East Wing (LSDC 020 RAT)
- Information management using dBase /; Yerkey, A. Neil; 1991; Eleftheriades Library (025.3 YER)
- Encyclopedia of job-winning resumes /; Fournier, Myra; 2006; Bissell Library, General Stacks Upper Level (650.14 FOU)
- Object-oriented programming in architecture; 1999; Bissell Library, Staff Office (VC 005.26 OBJ)
- Global software programmers; 1999; Bissell Library, Staff Office (VC 005.26 GLO)
- StudyWorks! for science / Mathsoft; 1986; Eleftheriades Library (CD 503 STU)
- Consider a spherical cow : a course in environmental problem solving /; Harte, John; 1988; Bissell Library, General Stacks (363.7 HAR)
- Computer images /; 1986; Eleftheriades Library (006.6 COM)
- Object-oriented client/server application development using ObjectPAL and C++ /; Ayer, Steve J.; 1995; Bissell Library, General Stacks (005.758 AYE)
- Mind tools : the science of artificial intelligence /; Bortz, Alfred B.; 1992; Eleftheriades Library (006.3 BOR)
- Computationalism : new directions /; 2002; Bissell Library, General Stacks (004 COM)
- Toward autonomous, adaptive, and context-aware multimodal interfaces : theoretical and practical issues : third COST 2102 International Training School Caserta, Italy, March 15-19, 2010 : revised selected papers /; 2011; Bissell Library, General Stacks (004.019 COS)
- Learning PHP, MySQL & JavaScript : with jQuery, CSS & HTML5 /; Nixon, Robin; 2015; Bissell Library, Reference Desk (005.2762 NIX)
- Object-oriented programming in the arts; 1999; Bissell Library, Staff Office (VC 005.26 OBJ)
- Cite right : a quick guide to citation styles--MLA, APA, Chicago, the sciences, professions, and more /; Lipson, Charles; 2006; Bissell Library, East Wing (LSDC 808.027 LIP)
- Sets, logic and maths for computing /; Makinson, David; 2012; Bissell Library, General Stacks (004.01 MAK)
- Electronic payment systems for e-commerce /; O'Mahony, Donal; 2001; Bissell Library, General Stacks (332.1028 O'MA)
- Data structures and problem solving using Java /; Weiss, Mark Allen; 2002; Bissell Library, General Stacks (005.133 WEI)
- Java design : objects, UML, and process /; Knoernschild, Kirk; 2002; Bissell Library, General Stacks (005.133 KNO)

- Data structures, algorithms, and applications in Java /; Sahni, Sartaj; 2000; Bissell Library, General Stacks (005.74 SAH), Bissell Library (005.74 SAH)
- Java the UML way : integrating object-oriented design and programming /; Lervik, Else; 2002; Bissell Library, General Stacks (005.133 LER)
- An introduction to object-oriented programming with Java /; Wu, C. Thomas; 2001; Bissell Library, General Stacks (005.133 WU), Bissell Library (005.133 WU)
- Physics for computer science students : with emphasis on atomic and semiconductor physics /; Garcia, Narciso,; 1991; Bissell Library, General Stacks (530 GAR)
- Encyclopedia of computer science and technology /; Henderson, Harry; 2003; Bissell Library, Reference Stacks (R 004.03 HEN)
- Essential discrete mathematics for computer science /; Feil, Todd; 2003; Bissell Library, General Stacks (004.01 FEI)
- Invitation to computer science /; Schneider, G. Michael; 2013; Eleftheriades Library, General Stacks (004 SCH)
- Fundamentals of discrete math for computer science : a problem-solving primer /; Jenkyns, T. A.; 2013; Bissell Library, General Stacks (004.01 JEN)
- GRE : practicing to take the computer science test.; 1997; Bissell Library, General Stacks (004.071 GRE)
- Computing tomorrow : future research directions in computer science /; 1996; Bissell Library, General Stacks (004.072 COM)
- Concrete mathematics : a foundation for computer science /; Graham, Ronald L.; 1994; Eleftheriades Library (510 GRA)
- Discrete Mathematics /; Graham, Ronald L. ;2011; Eleftheriades Library, General Stacks (510 GRA)
- Thesis projects : a guide for students in computer science and information systems /; 2008; Bissell Library, Reference Desk (004.072 THE)
- Pascal Algorithms : a Pascal-based introduction to computer science /; Reilly, Edwin D.; 1989; Bissell Library, General Stacks (005.262 PAS)
- Security systems and software; 1999; Bissell Library, Staff Office (VC 005.26 SEC)
- The computer : a very short introduction /; Ince, D.; 2011; Eleftheriades Library, General Stacks (004.16 INC)
- Eyewitness encyclopedia of science the essential multimedia reference guide to science and technology; 1994; Eleftheriades Library (CD 503 EYE)
- Java collections : an introduction to abstract data types, data structures, and algorithms /; Watt, David A.; 2001; Bissell Library, General Stacks (005.133 WAT)
- Fundamentals of data structures in Pascal /; Horowitz, Ellis; 1984; Bissell Library, General Stacks (005.262 PAS)
- Neural networks and fuzzy systems : a dynamical systems approach to machine intelligence /; Kosko, Bart; 1992; Bissell Library, General Stacks (006.3 KOS)
- The First Book of Information Science /; Becker, Joseph; 1973; Eleftheriades Library (004.09 BEC)
- Fuzzy sets, neural networks, and soft computing /; 1994; Bissell Library, General Stacks (006.3 FUZ)
- Partitioning data sets : DIMACS workshop, April 19-21, 1993 /; 1995; Bissell Library, General Stacks Upper Level (621.367 PAR)
- Cognitive vision systems : sampling the spectrum of approaches /; 2006; Bissell Library, General Stacks (006.37 COG)
- Creating successful software for your customer; 1999; Bissell Library, Staff Office (VC 005.26 CRE)
- The global network and object-oriented programming; 1999; Bissell Library, Staff Office (VC 005.26 GLO)
- The history of software; 1999; Bissell Library, Staff Office (VC 005.26 HIS)

- Object-oriented data structures using Java /; Dale, Nell B.; 2002; Bissell Library, General Stacks (005.133 DAL)
- Darwin among the machines /; Dyson, George; 1998; Eleftheriades Library (006.3 DYS)
- Data structures & algorithms in Java /; Lafore, Robert; 2003; Bissell Library, Reference Desk (005.73 LAF)
- In the image of the brain : breaking the barrier between the human mind and intelligent machines /; Jubak, Jim; 1992; Eleftheriades Library (006.3 JUB)
- Intelligent systems for finance and business /; 1995; Bissell Library, General Stacks Upper Level (658.0563 INT)
- Data structures in Java : a laboratory course /; Andersen, Sandra; 2002; Bissell Library, General Stacks (005.133 AND)
- JavaScript : the definitive guide /; Flanagan, David; 2002; Bissell Library, General Stacks (005.2762 FLA)
- Computer models for operations management /; Hall, Owen P.; 1993; Bissell Library, General Stacks Upper Level (658.5 HAL)
- Statistical principles of research design and analysis /; Kuehl, R. O.; 1994; Bissell Library, General Stacks (001.422 KUE)
- Theoretical models in biology : the origin of life, the immune system, and the brain /; Rowe, Glenn; 1994; Bissell Library, General Stacks (570.0113 ROW)
- Simulating neural networks with Mathematica /; Freeman, James A.; 1994; Bissell Library, General Stacks (006.3 FRE)
- The Internet for scientists and engineers : online tools and resources /; Thomas, Brian J.; 1996; Bissell Library, General Stacks (004.67 THO)
- HAL's legacy : 2001s computer as dream and reality /; 1997; Bissell Library, General Stacks (004 HAL)
- Software engineering with B /; Wordsworth, J. B.; 1996; Bissell Library, General Stacks (005.133 WOR)
- Compiler construction /; Wirth, Niklaus; 1996; Bissell Library, General Stacks (005.453 WIR)
- The F programming language /; Metcalf, Michael; 1996; Bissell Library, General Stacks (005.133 MET)
- Principles of artificial intelligence and expert systems development /; Rolston, David W.; 1988; Bissell Library, General Stacks (006.3 ROL)
- The art of computer programming; Knuth, Donald Ervin; 1973; Bissell Library, General Stacks (005.1 KNU)
- Object-oriented programming via Fortran 90/95 /; Akin, J. E.; 2003; Bissell Library, Reference Desk (005.133 AKI)
- The elements of UML style /; Ambler, Scott W.; 2003; Bissell Library, General Stacks (005.117 AMB)
- Programming with objects : a comparative presentation of object-oriented programming with C++ and Java /; Kak, Avinash C.; 2003; Bissell Library, General Stacks (005.133 KAK)
- Memory as a programming concept in C and C++ /; Franek, F.; 2004; Bissell Library, General Stacks (005.435 FRA)
- Game architecture and design /; Rollings, Andrew; 2004; Bissell Library, General Stacks (005.26 ROL)
- Algorithms /; Sedgewick, Robert ;1983; Bissell Library, General Stacks (519.4 SED)
- Virtual reality /; Rheingold, Howard; 1991; Eleftheriades Library (501.13 RHE)
- Algorithms and data structures /; Wirth, Niklaus; 1986; Bissell Library, General Stacks (005.73 WIR)
- Information, randomness & incompleteness : papers on algorithmic information theory /; Chaitin, Gregory J.; 1990; Eleftheriades Library (511.3 CHA)
- Algorithmic information theory /; Chaitin, Gregory J.; 1987; Eleftheriades Library, General Stacks (004 CHA)

- Information-theoretic incompleteness /; Chaitin, Gregory J.; 1998; Eleftheriades Library (511.3 CHA)
- The limits of mathematics : a course on information theory and the limits of formal reasoning /; Chaitin, Gregory J.; 2003; Eleftheriades Library (511.3 CHA)
- Artificial intelligence and intelligent systems /; Padhy, N. P.; 2005; Bissell Library, General Stacks (006.33 PAD)
- Affective, interactive and cognitive methods for e-learning design : creating an optimal education experience /; 2010; Eleftheriades Library, General Stacks (371.334 AFF)
- Engineering problem solving with C++ /; Etter, Delores M.; 2012; Bissell Library, Reference Desk (620.00285 ETT)
- Head first Java /; Sierra, Kathy; 2005; Bissell Library, New Books Area (005.133 SIE)
- Introduction to programming in Java : an interdisciplinary approach /; Sedgewick, Robert; 2014; Bissell Library, New Books Area (005.133 SED)
- Digital design and computer architecture /; Harris, David Money; 2013; Bissell Library, Reference Desk (621.381 HAR)
- Joomla! 3 beginner's guide /; Tiggeler, Eric; 2014; Bissell Library, Reference Desk (006.78 TIG)
- Would-be worlds : how simulation is changing the frontiers of science /; Casti, J. L.; 1997; Bissell Library, General Stacks (003.7 CAS)
- Thinking in complexity : the complex dynamics of matter, mind, and mankind /; Mainzer, Klaus; 1997; Bissell Library, General Stacks (501 MAI)
- Visions : how science will revolutionize the twenty-first century /; Kaku, Michio; 1998; Eleftheriades Library (501.72 KAK)
- The computer and the mind : an introduction to cognitive science /; Johnson- Laird, P. N.; 1988; Bissell Library, General Stacks (153 JOH)
- Decoding the universe : how the new science of information is explaining everything in the cosmos, from our brains to black holes /; Seife, Charles; 2006; Bissell Library, General Stacks (006.33 SEI)
- The innovators : how a group of inventors, hackers, geniuses, and geeks created the digital revolution /; Isaacson, Walter; 2014; Bissell Library, General Stacks (004.0922 ISA)
- Discrete mathematics /; Lipschutz, Seymour; 2007; Bissell Library, General Stacks (511.1 LIP)
- Objects first with Java : a practical introduction using BlueJ /; Barnes, David J.; 2012; Bissell Library, New Books Area (005.133 BAR)
- Cognitive behavioural systems : COST 2102 International Training School, Dresden, Germany, February 21-26, 2011 : revised selected papers/; 2012; Bissell Library, General Stacks (006.3 COG)
- Philosophy and computing : an introduction /; Floridi, Luciano; 1999; Bissell Library, General Stacks (004.01 FLO)
- Mind design II : philosophy, psychology, artificial intelligence /; 1997; Bissell Library, General Stacks (006.3 MIN)
- The philosophy of artificial intelligence /; 1990; Bissell Library, General Stacks (006.3 PHI)
- The metaphorical brain 2 : neural networks and beyond /; Arbib, Michael A.; 1989
- The great interactive dream machine : another adventure in cyberspace /; Peck, Richard; 1996; Elementary Library (JUNIOR PECK)
- Virtual reality : computers mimic the physical world /; Grady, Sean M., 1998
- Neural networks and natural intelligence /; 1988; Bissell Library, General Stacks Upper Level (612.81 NEU)
- The paperback computer; 1992; Bissell Library, Staff Office (VC 004.09 PAP)
- Computational financial mathematics using Mathematica : optimal trading in stocks and options /; Stojanovic, Srdjan; 2003; Bissell Library, Reference Desk (332.63 STO)
- Imitation of life : how biology is inspiring computing /; Forbes, Nancy; 2004; Bissell Library, General Stacks (570 FOR)

- Evolutionary computation : a unified approach /; De Jong, Kenneth A.; 2006; Bissell Library, General Stacks (005.1DE)
- The Blackwell guide to the philosophy of computing and information /; 2004; Bissell Library, General Stacks (004.01 BLA)
- Getting started with MATLAB : a quick introduction for scientists and engineers /; Pratap, Rudra.; 2010; Bissell Library, General Stacks Upper Level (620.001 PRA)
- Operating system concepts /; Silberschatz, Abraham; 2014; Bissell Library, Reference Desk (005.43 SIL)
- Projects in computing and information systems : a student's guide /; Dawson, Christian W.; 2015; Bissell Library, Reference Desk (004 DAW)
- Invent to learn : making, tinkering, and engineering in the classroom /; Martinez, Sylvia Libow; 2013; Eleftheriades Library, General Stacks (507.1 MAR)
- Research methodology : a step-by-step guide for beginners /; Kumar, Ranjit; 2014; Bissell Library, Reference Desk (300.72 KUM)
- Android application development cookbook : over 100 recipes to help you solve the most common problems faced by Android developers today /; Boyer, Rick; 2016; Bissell Library, Reference Desk (005.3 BOY)
- The thrilling adventures of Lovelace and Babbage : with interesting & curious anecdotes of celebrated and distinguished characters : fully illustrating a variety of instructive and amusing scenes; as performed within and without the remarkable difference engine /; Padua, Sydney; 2015; Eleftheriades Library, Fiction Stacks (GN PADUA)
- Ada Lovelace, poet of science : the first computer programmer /; Stanley, Diane; 2016; Elementary Library (510.92 STA)
- Ada Byron Lovelace and the thinking machine /; Wallmark, Laurie; 2015; Elementary Library (92 LOVELACE)
- Object-oriented methods : a foundation /; Martin, James; 1998; Bissell Library, General Stacks (005.12 MAR)
- Object-oriented modeling and design /; 1991; Bissell Library, General Stacks (005.1 OBJ)
- Systems analysis and design /; Kendall, Kenneth E.; 2002; Bissell Library, General Stacks (004.21 KEN)
- An introduction to object-oriented programming /; Budd, Timothy; 2002; Bissell Library, General Stacks (005.117 BUD)
- Lectures in e-commerce /; 2001; Bissell Library, General Stacks Upper Level (658.8 LEC)
- A first course in discrete mathematics /; Anderson, Ian; 2001; Eleftheriades Library (510 AND)
- Business information systems : analysis, design, and practice /; Curtis, Graham; 2002; Bissell Library, General Stacks Upper Level (658.4038 CUR)
- Fundamentals of neural networks : architectures, algorithms, and applications /; Fausett, Laurene V.; 1994; Bissell Library, General Stacks (006.3 FAU)
- The cognitive brain /; Trehub, Arnold; 1991; Bissell Library, General Stacks (153 TRE)
- Introduction to operations research /; Hillier, Frederick S.; 1995; Bissell Library, General Stacks Upper Level (658.4034 HIL)
- Discrete mathematics with algorithms /; Albertson, Michael O.; 1988; Bissell Library, General Stacks (510 ALB)
- Neural network time series forecasting of financial markets /; Azoff, E. Michael; 1994; Bissell Library, General Stacks (332.645 AZO)
- Robot vision /; Horn, Berthold; 1986; Bissell Library, General Stacks Upper Level (629.892 HOR)
- Learning algorithms : theory and applications in signal processing, control, and communications /; Mars, P.; 1996; Bissell Library, General Stacks Upper Level (629.8 MAR)
- A first course in fuzzy logic /; Nguyen, H. T.; 1997; Bissell Library, General Stacks (511.3 NGU)
- Corpus processing for lexical acquisition /; 1996; Bissell Library, General Stacks (006.35 COR)
- Distributed database systems /; Bell, D. A.; 1992; Bissell Library, General Stacks (005.74 BEL)

- Object-oriented database systems : concepts and architectures /; Bertino, Elisa; 1993; Bissell Library, General Stacks (005.74 BER)
- Object technology strategies and tactics /; Singer, Gilbert L.; 1996; Bissell Library, General Stacks (005.12 SIN)
- Discrete mathematics and its applications /; Rosen, Kenneth H.; 1991; Bissell Library, General Stacks (511 ROS)
- Introduction to formal languages /; Revesz, Gyorgy E.; 1991; Bissell Library, General Stacks (511.3 REV)
- Applied iterative methods /; Hageman, Louis A.; 1981; Bissell Library, General Stacks (519.4 HAG)
- Artificial intelligence : a beginner's guide /; Whitby, Blay;; 2003; Bissell Library, General Stacks (006.3 WHI)
- Artificial intelligence and natural man /; Boden, Margaret A.; 1977; Bissell Library, General Stacks (006.3 BOD)
- Cross-curricular word problems : developing children's problem-solving skills in the daily maths lesson.; Clarke, Peter;; 2010; Elementary Library, Reference Desk (510 CLA)
- Data made flesh : embodying information /; ; 2004; Bissell Library, General Stacks (306.4 DAT)
- Discrete mathematics : elementary and beyond /; Lovasz, Laszlo;; 2003; Bissell Library, General Stacks (510 LOV)
- Discrete mathematics demystified /; Krantz, Steven G.; 2009; Bissell Library, General Stacks (511.1 KRA)
- Discrete structures /; Fell, Harriet; 2017; Bissell Library, Reference Desk (511.1 FEL)
- E-Learning and the science of instruction : proven guidelines for consumers and designers of multimedia learning /; Clark, Ruth Colvin; 2003; Bissell Library, East Wing (FDC 658.3124 CLA)
- Essentials of discrete mathematics /; Hunter, David James;; 2012; Bissell Library, General Stacks (004.01 HUN)
- Expert systems : tools and applications /; Harmon, Paul; 1988; Bissell Library, General Stacks Upper Level (620.028 HAR)
- Guide to the successful thesis and dissertation : a handbook for students and faculty /; Mauch, James E.;; 2003; Bissell Library, Learning Hub (SSC 808.02 MAU)
- JavaScript : the definitive guide /; Flanagan, David; 1998; Bissell Library, General Stacks (005.2 FLA)
- Mathematics and physics for programmers /; Kodicek, Danny; 2005; Bissell Library, Reference Desk (510.24 KOD)
- Mobile applications development with Android : technologies and algorithms /; Qiu, Meikang; 2017; Bissell Library, Reference Desk (005.365 QIU)
- On intelligence /; Hawkins, Jeff;; 2004; Bissell Library, General Stacks Upper Level (612.82 HAW)
- Privacy on the line : the politics of wiretapping and encryption /; Diffie, Whitfield; 1998; Bissell Library, General Stacks (342.730858 DIF)
- Programming with Microsoft Visual Basic 6.0 : an object-oriented approach ; comprehensive /; Ekedahl, Michael V.; 1999; Bissell Library, General Stacks (005.1313 VIS)
- Statistical mechanics of learning /; Engel, A.; 2001; Bissell Library, General Stacks (006.3 ENG)
- Technology supported learning and teaching : a staff perspective /; ; 2006; Bissell Library, General Stacks (371.33 TEC)
- The code book : the secret history of codes and code-breaking /; Singh, Simon; 2000; Bissell Library, General Stacks Upper Level (652.8 SIN)
- The wrong side of the galaxy /; Thomson, Jamie;; 2014; Elementary Library (JUNIOR THOMSON)
- UML xtra-light : how to specify your software requirements /; Kratochvil, Milan; 2003; Bissell Library, General Stacks (005.1 KRA) Bissell Library, General Stacks (005.1 KRA)
- Understanding search engines : mathematical modeling and text retrieval /; Berry, Michael W.; 2005; Bissell Library, East Wing (LSDC 025.04 BER)

- E-commerce and web technologies : 5th International Conference, EC-Web 2004, Zaragoza, Spain, August 31-September 3, 2004 : proceedings /; ; 2004; Bissell Library, General Stacks Upper Level (658.84 INT)
- The tools for successful online teaching /; Dawley, Lisa; 2007; Bissell Library, Reference Desk (371.334 DAW)
- Data analysis using SQL and Excel /; Linoff, Gordon; 2008; Bissell Library, General Stacks (005.75 LIN)
- JavaScript & jQuery : interactive front-end web development /; Duckett, Jon; 2014; Bissell Library, Reference Desk (005.133 DUC)
- Neuromancer /; Gibson, William; 1984; Bissell Library, Fiction Stacks (FIC GIBSON)
- Thing explainer : complicated stuff in simple words /; Munroe, Randall; 2015; Eleftheriades Library, General Stacks (500 MUN)
- Beginning PHP 5.3 /; Doyle, Matt; 2010; Bissell Library, Reference Desk (006.76 DOY)
- Android UI design : plan, design, and build engaging user interfaces for your Android applications; Thornsby, Jessica; 2016; Bissell Library, Reference Desk (005.437 THO)
- Design patterns : elements of reusable object-oriented software /; Gamma, Erich; 2016; Bissell Library, New Books Area (005.117 GAM)
- Adobe Photoshop CC for photographers : version 2015.5 : a professional image editor's guide to the creative use of Photoshop for the Macintosh and PC /; Evening, Martin; 2017; Bissell Library, General Stacks (006.696 EVE)
- Cisco CCENT/CCNA ICND1 100-101 official Cert guide /; Odom, Wendell; 2013; Bissell Library, General Stacks (004.6076 ODO)
- The hidden power of blend modes in Adobe Photoshop /; Valentine, Scott; 2013; Bissell Library, General Stacks (006.686 VAL)
- The cybercultures reader /; 2007; Bissell Library, General Stacks (303.4833 CYB)
- Doing a successful research project : using qualitative or quantitative methods /; Davies, Martin; 2014; Bissell Library, Reference Desk (001.42 DAV)
- Applications of discrete mathematics /; 1991; Bissell Library, General Stacks (511.1 APP)
- The singularity is near : when humans transcend biology /; Kurzweil, Ray; 2005; Bissell Library, General Stacks (153.9 KUR)
- Στοιχεία διακριτών μαθηματικών /; Liu, C. L.; 2002; Eleftheriades Library, General Stacks (510 LIU)
- Quantitative methods for business decisions : with cases /; Lapin, Lawrence L.; 1994; Bissell Library, General Stacks Upper Level (658.4033 LAP)
- Ideas that changed the world; Ingpen, Robert R.; 1996; Eleftheriades Library (CD 609 ING)
- Fade to Blue : a novel /; Beaudoin, Sean; 2009; Eleftheriades Library, Fiction Stacks (YA BEAUDOIN)
- Blogs, wikis, podcasts, and other powerful Web tools for classrooms /; Richardson, Will; 2010; Eleftheriades Library, General Stacks (025.04 RIC)
- Calculus : early transcendentals /; Stewart, James; 2012; Bissell Library, General Stacks (515 STE) Bissell Library, General Stacks (515 STE)
- Mona Lisa overdrive /; Gibson, William; 1988; Bissell Library, Fiction Stacks (FIC GIBSON)
- Internet research skills /; Ó; 2012; Bissell Library, Reference Desk (025.04 ODO)
- Fundamentals of database systems /; Elmasri, Ramez; 2016; Bissell Library, Reference Desk (005.74 ELM)
- Bebob to the boolean boogie : an unconventional guide to electronics /; Maxfield, Clive; 2009; Bissell Library, Reference Desk (621.3815 MAX)
- Adobe Audition CC /; Anderton, Craig; 2013; Bissell Library, Reference Desk (621.3893 AND)
- Adobe Premiere Pro CC 2015 release /; Jago, Maxim; 2016; Bissell Library, General Stacks (006.686 JAG)
- Digital media : concepts and applications /; Crews, Tena B.; 2017; Bissell Library, Reference Desk (006.7 CRE)

- 100 diagrams that changed the world : from the earliest cave paintings to the innovation of the iPod /; Christianson, Scott; 2012; Eleftheriades Library, General Stacks (608 CHR)
- Introduction to research methods : a practical guide for anyone undertaking a research project /; Dawson, Catherine; 2009; Bissell Library, Reference Desk (001.42 DAW)

Electronic resources (databases) added to our collections:

- ACM Digital Library which includes access to 45 journal titles for Computer Science among other resources
- Business Source Complete expanding the subscription of Business Source Elite to provide 2,725 more titles to support the Business Division