Department of Computer Science and Information Systems

MSc Advanced Computing Technologies Programme Arrangements 2019–2020

Version of October 17, 2019

This is not a final document and minor changes may be introduced until the end of October. If in doubt, please contact igor@dcs.bbk.ac.uk. This statement will be removed once the version is final.

Contents

1 General Information

1.1 Contacts

Programme Director:	Igor Razgon, igor@dcs.bbk.ac.uk
Programme Administrator:	Stacey Hine, pg@dcs.bbk.ac.uk
Admissions Tutor:	Hubie Chen, hubie@dcs.bbk.ac.uk
Projects Tutor:	Oded Lachish, oded@dcs.bbk.ac.uk
Disability Officer:	Oded Lachish, oded@dcs.bbk.ac.uk

1.2 Web presence

Detailed and updated information about the programme is available from the

- programme internet page:

  www.dcs.bbk.ac.uk/study/postgraduate-specialist/msc-advanced-computing-technologies/

- departmental internet page for current students:

  www.dcs.bbk.ac.uk/current-students/

- programme Moodle page for enrolled students:

  moodle.bbk.ac.uk/course/view.php?id=3292

The Moodle Virtual Learning Environment (moodle.bbk.ac.uk) is used to provide detailed information and post announcements about each module on which you are enrolled.

It is your responsibility to familiarise yourself with the contents of this booklet as well as the internet pages listed above. You should also consult Moodle and read your College email on a regular basis.

2 Student Support

Every student is allocated a personal tutor in the first weeks of the programme. The personal tutor is someone students can contact to discuss any problems of a non-academic nature. These may relate to special needs or personal problems that may affect the student's academic performance. The Department also has a disability officer whom students can contact.

Academic problems should first be addressed to the lecturer concerned. If the problem is not resolved or it does not relate to a specific module, then the Programme Director should be contacted.

For more general information about Birkbeck, student services and regulations have a look at

`https://www.bbk.ac.uk/student-services`.

It is expected that students familiarise themselves with these pages so that they are aware of the services and regulations.

The School of Business, Economics and Informatics has Learning Co-ordinators. Their role is to support students in their studies. They can offer help and support on a variety of topics ranging from writing skills to basic maths. See

`http://www.bbk.ac.uk/business/current-students/learning-co-ordinators`

for details.

3 Important Dates

Lectures will commence in the week starting on **Monday 30 September 2019**. The teaching (i.e., not including exams and project) covers two terms of eleven weeks each (autumn and spring term). The summer term is given over to revision (including revision lectures), exams, and the project.

- Autumn term: 30 September 2019 – 13 December 2019.
- Spring term: 13 January 2020 – 27 March 2020.
- Summer term: 27 April 2020 – 10 July 2020.

Please refer to

`http://www.bbk.ac.uk/about-us/term-dates`

for the College holiday closing times.

Students should attend lectures during term time as shown in the timetables in Section 4.2. If students are unable to attend lectures, they should arrange with lecturers or fellow-students to obtain copies of any material distributed in class.

Any student who decides to withdraw from the programme should inform the Programme Administrator(s). Students who simply stop turning up for lectures without formally withdrawing from the programme will still be held liable for fees. It is especially important for international students that they inform the department about any absence.

4 Syllabus

4.1 List of Modules

We give a general overview of the content of the programme here; detailed descriptions of the individual modules are given in Section 5.

The MSc in Advanced Computing Technologies is a specialist programme of study, focusing on areas of expertise and research specialisation within the Department of Computer Science and Information Systems:

- Data Analytics
- Information and Web Technologies
- Learning Technologies

As an alternative to the award of MSc Advanced Computing Technologies, the modules and project may be chosen with a main focus in one of these areas to be awarded an MSc Data Analytics, MSc Information and Web Technologies or MSc Learning Technologies (see Section 7.1).

Full-time students follow 8 taught modules and undertake a 3–4 month project. Part-time students are normally expected to take 4 taught modules in each of the two years and the project component in the second year.

Students select their taught modules from the following:

- Advances in Data Management (ADM)
- Applied Machine Learning (AML)
- Big Data Analytics Using R (BDA)
- Cloud Computing (CC)
- Component-Based Software Development (CBSD)
- Data Science Techniques and Applications (DSTA)
- Data Warehousing and Data Mining (DWDM)
- Information and Network Security (INSEC)
- Natural Language Processing and Information Retrieval (NLP)
- Interactive Systems (IRS)
- Internet and Web Technologies (IWT)
- Machine Learning (ML)
- Mobile Computing and the Internet of Things (MCIT)

- Programming Paradigms and Languages (PPL)
- Semantic Technologies (ST)
- Software Design and Programming (SDP)

If you are not a proficient Java programmer or not familiar enough with object-oriented design principles, you are strongly advised to study material such as that available through the following link: `https://docs.oracle.com/javase/tutorial/java/concepts/`

Not all modules will necessarily be offered each year. In addition, the availability of modules is subject to timetabling constraints and student demand. In the event that a module is over-subscribed, available places will be allocated on a first-come, first-served basis determined by the date you submit your module choices.

Information about any subsequent changes and more detailed information about aspects of the programme are available through the Moodle Virtual Learning Environment (`moodle.bbk.ac.uk`) at

`moodle.bbk.ac.uk/course/view.php?id=3292`.

Moodle is also used to provide detailed information and post announcements about each module.

4.2 Timetables

The teaching venues will be announced online at:

`http://www.dcs.bbk.ac.uk/current-students/`

as well as in the My Birkbeck portal. Below is the timetable for the modules. Note that occasionally there might be changes (e.g. swapping of lectures between modules, or additional tutoring sessions). Please consult the web pages of the modules regularly for up-to-date information.

Module abbreviations used in the following timetables are given in Section 4.1 above.

4.2.1 Timetable for 2019/20

MSc Advanced Computing Technologies

Timetable for 2019/20

Part-time students must select 4 modules each year (maximum of 5) in such a way as to ensure that they can complete their chosen 8 modules in 2 years.

Day	Autumn		Spring	
	Module	Time	Module	Time
Monday	**MCIT***	**14:00–17:00**	ADM[1] (FT**)	14:00–17:00
	ST/INSEC	18:00–21:00	ADM (PT)	18:00–21:00
Tuesday	BDA (FT**)	13:30–17:00	SDP (FT**)	13:30–17:00
	BDA (PT)	18:00–21:00	NLP/IWT	18:00–21:00
Wednesday	PPL	18:00–21:00	DSTA/DWDM	18:00–21:00
Thursday	AML/IRS	18:00–21:00	ML/SDP (PT)	18:00–21:00
Friday	Project[2] (FT)	14:00–17:00	CC	18:00–21:00
	CBSD/Project[2] (PT)	18:00–21:00		

*Note that some modules are offered in the evening on alternate years only.

** FT and PT students are required to take module according to mode of study.

Indicative timetable for 2020/21

Day	Autumn		Spring	
	Module	Time	Module	Time
Monday	**ST (FT**)**	**14:00-17:00**	ADM[1] (FT**)	14:00–17:00
	MCIT/INSEC	18:00–21:00	ADM (PT)	18:00–21:00
Tuesday	BDA	18:00–21:00	SDP (FT**)	13:30–17:00
			NLP/IWT	18:00–21:00
Wednesday	PPL	18:00–21:00	DSTA/DWDM	18:00–21:00
Thursday	AML/IRS	18:00–21:00	ML/SDP (PT)	18:00–21:00
Friday	Project[2] (FT)	14:00–17:00	CC	18:00–21:00
	CBSD/Project[2] (PT)	18:00–21:00		

*Note that some modules are offered in the evening on alternate years only.

** FT and PT students are required to take module according to mode of study.

[1]Lab sessions for ADM are repeated in the afternoons for full-time students.

[2]Lectures giving guidance on projects are held in Week 3 of the autumn term. Lectures are repeated to allow attendance on either Friday afternoon or Friday evenings

5 Module Descriptions

Lectures aim to introduce the key ideas of each module. The specific objectives of each module and the principal readings are circulated at the start of the term. The reading lists for individual modules given below are only indicative. Lecturers will specify, usually at the first lecture, whether or not books need to be purchased for particular modules. Independent study is a key learning objective of the programme.

Most modules have dedicated web pages that provide links to relevant online literature. Depending on the nature of the material, some lecturers use 'lecture outlines' to support their teaching and may even distribute these outlines via their web pages. However, there is no expectation that written notes will be provided for the modules.

Students can also contact lecturers outside the classroom to discuss the material. They can meet the lecturers during scheduled 'office hours' or can contact them via email either to discuss a problem or to make an appointment. Lecturers' contact details are given on the Department web site.

Students are expected to attend all lectures. Any student who decides to withdraw from the programme should inform the Programme Administrator, in writing or by email. Students who simply stop turning up for lectures without formally withdrawing from the programme will still be held liable for fees.

A number of modules require students to submit coursework as part of the assessment. More details are given in Section 7.5 of this handbook.

5.1 Advances in Data Management (ADM)

Teaching Staff

Alex Poulovassilis, Richard Brownlow (2019/20)

Online material

`http://moodle.bbk.ac.uk/`

Aims

To study advanced aspects of database management and recent advances in data management technologies in three major directions: performance, distribution of data, and heterogeneity of data.

The module examines the technologies underlying modern database management systems. It studies advanced aspects of query processing, transaction management, distributed data management, and recent developments in web data, big data and alternative database architectures.

Syllabus

- Review of the fundamental principles of modern database management systems, relational databases and SQL.
- Query processing and query optimisation.
- Transaction management: ACID properties, concurrency control, recovery.
- Beyond records and objects: stored procedures and functions, triggers, semantic technologies.
- Distributed databases: data fragmentation and replication, distributed query processing, distributed transaction management.
- Heterogeneous data integration.
- XML data management.
- Linked Data.
- Parallel databases.
- NoSQL/NewSQL stores.
- Graph databases.

Prerequisites

A first module in Database Systems (e.g. as taught in a typical UK undergraduate degree in computer science) or the Birkbeck module "Data and Knowledge Management". Knowledge of computer programming.

Assessment

By 2-hour written examination and coursework, with weighting 90% and 10%, respectively.

Reading

- R. Ramakrishnan and J.Gehrke, Database Management Systems, McGraw-Hill 2003 (3rd Edition)
- A.Silberschatz, H.F.Korth and S.Sudarshan, Database System Concepts, McGraw-Hill 2011 (6th Edition)
- M. T. Oszu, P. Valduriez, Principles of Distributed Database Systems, Prentice-Hall 2011 (3rd Edition)
- W.Lemahieu, S. Vanden Broucke, B. Baesens, Principles of Database Management - The Practical Guide to Storing, Managing and Analyzing Big and Small Data, Cambridge University Press, 2018
- Research papers will be distributed to students; students will also be directed to Web resources on the subject.

5.2 Applied Machine Learning (AML)

Teaching Staff
Paul Yoo

Online material
http://moodle.bbk.ac.uk/

Aims
This module covers the fundamental concepts and techniques of applied machine learning using Python and how to use the existing tools to analyse data. Students develop the hands-on and practical skills needed for applied machine learning including the use of existing Python libraries and tools (e.g. Scikit-Learn and TensorFlow) and the use of the techniques needed to analyse data (e.g. pre-processing, feature selection and classification). The module will use Python the most popular machine learning language to solve practical problems based on use cases extracted from real domains such as financial forecasting and computer vision.. . .

Syllabus
- Introduction to Python for machine learning
- Preparing data
- Feature selection for machine learning
- Evaluation and resampling
- Rule-based algorithms: decision tree and random forest
- Regression-based algorithms: logistic regression and neural networks
- Large-scale machine learning using TensorFlow
- Real-life case studies (e.g. financial forecasting and computer vision)

Prerequisites
No specific module is pre- or co-requisite but knowledge of mathematical concepts and basic python will be assumed.

Assessment
By 2-hour written examination and practical coursework, weighting 70% and 30%, respectively.
29

Reading
- G´eron, A., 2017. Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems. " O'Reilly Media, Inc."
- Students will also be directed to papers available online and other Web resources on the subject.

5.3 Big Data Analytics Using R (BDA)

Teaching Staff

Tingting Han

Online material

`http://moodle.bbk.ac.uk`

Aims

This module covers the principle concepts and techniques of data analytics and how to apply them to large-scale data sets. Students develop the core skills and expertise needed by data scientists, including the use of techniques such as linear regression, classification and clustering. The module will show you how to use the popular and powerful data analysis language and environment R to solve practical problems based on use cases extracted from real domains.

Syllabus

- Introduction to big data analytics: big data overview, data pre-processing, concepts of supervised and unsupervised learning.
- Basic statistics: mean, median, standard deviation, variance, correlation, covariance.
- Linear regression: simple linear regression, introduction to multiple linear regression.
- Classification: logistic regression, decision trees, SVM.
- Ensemble methods: bagging, random forests, boosting.
- Clustering: K-means, K-medoids, Hierarchical clustering, X-means.
- Evaluation and validation: cross-validation, assessing the statistical significance of data mining results.
- Selection of advanced topics such as: scalable machine learning, big data related techniques, mining stream data, social networks.
- Tools: R.

Prerequisites

Experience with a modern programming language.

Assessment

By 3-hour written examination and practical coursework, weighting 80% and 20% respectively.

Reading

- Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani. An Introduction to Statistical Learning: With Applications in R. Springer, 2017. Available online at `http://www-bcf.usc.edu/~gareth/ISL/`.

5.4 Cloud Computing (CC)

Teaching Staff

Stelios Sotiradis

Online material

`http://www.dcs.bbk.ac.uk/~stelios/cc/`

Aims and Outline

The Cloud Computing module aims to introduce back-end Cloud Computing techniques for deploying applications, processing Big Data and developing scalable systems. The module focuses on (a) the theoretical aspects of Distributed Systems and Cloud Computing, and (b) the practical aspects by using Python to develop and deploy cloud-based applications.

Students will understand the emerging area of Cloud Computing and how it relates to traditional models of computing, while at the same time gaining competence in developing cloud-based services and applications. The students will also be introduced to Big Data processing systems and applications.

Syllabus

- Introduction to Cloud Computing and Virtualisation
- Cloud services and Cloud software engineering
- Parallel Computing and Distributed Systems
- Cloud data storage systems
- Deploying and using Cloud storage systems
- Introduction to Container systems
- Introduction to Big Data systems
- Introduction to Hadoop MapReduce framework
- Developing Hadoop Map Reduce applications
- Introduction to Internet of Things and Service-Oriented Architectures

Each class will include a laboratory session (that usually takes half the time) with practical exercises and tutorial sessions to provide hands-on experience of cloud application development and deployment. Students will have the opportunity to develop RESTful web services using Python and access virtualised Linux servers. The students will be introduced into various distributed computing systems and technologies such as APIs, NoSQL systems, container systems, Hadoop MapReduce, Amazon WS, as well as other systems and platforms.

Prerequisites

Very good knowledge of programming with Python.

MSc students who did not have much experience in software development before joining their respective postgraduate programmes should have already taken the Principles of Programming I (POP1) module.

Coursework

One programming assignment with report.

Assessment

By 2-hour written examination and practical coursework, weighting 80% and 20%, respectively.

Reading

- J. Rosenberg and A. Mateos, *The Cloud at Your Service*, Manning, 2010.
- Extensive use is made of other relevant book chapters and research papers that are distributed in class or provided online.

5.5 Component-Based Software Development (CBSD)

Teaching Staff

Keith Mannock

Online material

`http://moodle.bbk.ac.uk/`

Aims

This module introduces the theory and practice associated with implementing large-scale distributed information systems in heterogeneous environments. The student will develop the technical knowledge necessary to analyse the scalability and interoperability problems associated with large-scale heterogeneous systems and will experience the design and implementation of enterprise-level computer applications. Industry standard frameworks such as Java Enterprise Edition (JEE), Spring, Enterprise Service Bus (ESB), etc. will be explored utilising practical workshops. The development of web services will also be discussed through the use of standards such as XML, SOAP, WSDL and UDDI. How web services can be used to implement a Service Oriented Architecture (SOA) will be described. Note: this course requires a great deal of time outside class. Students should take this into account when considering taking other modules or if they encounter a lot of travel for work.

Syllabus

- Introduction to Component Based Software
- Java approaches to n-tier architectures (JEE, Spring, etc.)
- Enterprise Computing in the real world (Case Study)
- Message-Oriented Middleware (RabbitMQ, etc.)
- Virtualization and Cloud Computing
- Design Patterns and Enterprise Architectures
- The .NET model for distributed computing
- Persistence layers
- User Interfaces
- Web Services
- RESTful services

Prerequisites

Introduction to Software Development with a distinction level grade, or Principles of Programming I and II, or a similar course or relevant experience (as approved by the module leader).

Assessment

By 2-hour written examination and practical coursework, weighting 80% and 20%, respectively.

Reading

- C. Szyperski, Component Software: Beyond Object-Oriented Programming 2nd Edition, Addison Wesley, 2002.
- M. Hailperin, Operating Systems and Middleware, Course Technology Inc.. 2007.
- M. Fowler, Patterns of Enterprise Application Architecture, Addison Wesley, 2002.
- A. Lee Rubinger and B. Burke, Enterprise JavaBeans 3.1, O'Reilly, 2010.
- H. H. Liu, Developing Enterprise Java Applications with Spring Frameworks: An End-to-End Approach, PerfMath, 2012.

5.6 Data Science Techniques and Applications (DSTA)

Teaching Staff

Alessandro Provetti

Online material

`http://moodle.bbk.ac.uk/`

Aims

This module has been designed as the final module for MSc Data Science students. It presents to them a set of topics in data analytics, focussing on a range of applied data analysis techniques to convert information into knowledge. In particular, the "complex network" model of data organization is studied in some depth.

Syllabus

This module is designed for a minimal overlapping with the Machine Learning/Applied Machine learning modules, which are available as electives for MSc Data Science students.

- introduction to the module; Data Science as 9 computational problems;
- refresh some concepts of Statistics, Linear Algebra and Information theory;
- the geometric view of data, the curse of dimensionality, spectral and decomposition techniques;
- traditional Data Mining techniques such as Dimensionality reduction, e.g., PCA, SVD, SVMs; some kernelization;
- lab experience with Python modules for Data Analytics such as NumPy, Pandas and Scikit-learn;
- from data to graphs, and their relevant properties;
- centrality measures; communities (if time allows).

Prerequisites

The ability to program in Python, SQL and a basic knowledge of statistics normally obtained by taking the PoP I, PwD and BDA modules or as approved by the module leader.

Assessment

By a 2-hour written examination and courseworks, weighting 80% and 20% (aggregated), respectively.

Reading

The needed study materials may be made available electronically during the term.

- F. Provost and T. Fawcett, Data Science for Business. O'Reilly, 2013 (2nd edition is expected).
- M. Zaki and W. Meira, Data Mining and Analysis. CUP, 2014.
- G. Caldarelli and A. Chessa, Data Science and Complex Networks. Oxford University Press, 2016.
- J. Grus, Data Science from Scratch – First principles with Python (2nd ed.). O'Reilly, 2019.
- J. Voss, An Introduction to Statistical Computing: A Simulation-based Approach. Wiley, 2013.
- J. Vanderplas, A Whirlwind Tour of Python. O'Reilly, 2016.

5.7 Data Warehousing and Data Mining (DWDM)

Teaching Staff

Nigel Martin

Module URL

`http://www.dcs.bbk.ac.uk/~nigel/teaching/dwdm/`

Aims and Outline

To study advanced aspects of data warehousing and data mining, encompassing the principles, research results and commercial application of the technologies.

This module covers the organisation, analysis and mining of large data sets to support business intelligence applications. Students study the principles and commercial application of the technologies, as well as research results and emerging architectures underpinning the analysis and mining of "big data".

Syllabus

- Data warehousing requirements.
- Data warehouse conceptual design.
- Data warehouse architectures.
- Data warehouse logical design: star schemas, snowflake schemas, fact tables, dimensions, measures.
- OLAP architectures, OLAP operations. SQL extensions for OLAP.
- Data warehouse physical design: partitioning, parallelism, compression, indexes, materialized views, column stores.
- Data warehouse construction: data extraction, transformation, loading and refreshing. Warehouse metadata. Continuous ETL.
- Data warehouse architecture trends. MapReduce and warehouse architectures: Pig, Hive, Spark.
- Data mining concepts, tasks and algorithms.
- Data mining technologies and implementations. Techniques for mining large data sets, stream mining, architecture trends, standards, products.
- Research trends in data warehousing and data mining.

Prerequisites

A first module in Database Systems (e.g. as taught in a typical UK undergraduate degree in computer science) or the Birkbeck module Data and Knowledge Management.

Assessment

By 2-hour written examination and practical coursework, weighting 90% and 10% respectively.

Reading

- R. Ramakrishnan, J. Gehrke, Database Management Systems (3rd ed.), McGraw Hill, 2003, ISBN 0-07-246563-8.
- M. Golfarelli, S. Rizzi, Data Warehouse Design: Modern Principles and Methodologies, McGraw Hill, 2009, ISBN 978-0-07-161039-1.
- J. Celko, Joe Celko's Analytics and OLAP in SQL, Morgan Kaufmann, 2006, ISBN 978-0-12-369512-3.
- J. Han, M. Kamber, J Pei, Data Mining Concepts and Techniques (3rd ed.), Morgan Kaufmann, 2011, ISBN 978-0-12-381479-1.
- Research papers will be distributed to students; students will also be directed to Web resources on the subject.

5.8 Information and Network Security (INS)

Teaching Staff

Igor Razgon

Aims

Information security is about protecting information (and information systems) against unauthorised access and tampering. Avoiding security breaches has a high priority for organisations storing and handling confidential data.

The main aim of this module is to provide broad coverage of the field of information security. This course covers the technical as well as the management side of security in information systems. Despite being an essential part of security, technical methods such as cryptography are not enough to guarantee a high level of security. They have to be embedded into a wider context in order to make them more effective. Users of technology have to understand the underlying principles and follow certain policies to avoid security breaches. This module introduces the fundamental approaches to security engineering and includes a detailed look at some important applications.

Syllabus

- Overview of Information Security
- Access Control Matrix Model
- Security Policies
- Social Engineering
- Basic Cryptography
- Identity Management
- Access Control Mechanisms
- Confinement
- Assurance and Trust
- Network Intruders and Intrusion Detection
- Firewalls and Malicious Software
- Cryptographic Protocol Concepts
- Authentication
- Key Exchange
- Economics of Information Security

Assessment

Two-hour written examination (80%) and practical coursework (20%).

Reading

- Keith M. Martin, Everyday Cryptography: Fundamental Principles and Applications, 2012, ISBN 978-0-19-969559-1
- William Stallings, Cryptography and Network Security 5th edition, Pearson, 2010, ISBN 0136097049
- Matt Bishop, Computer Security: Art and Science, Addison-Wesley, 2002, ISBN 0201440997
- Ross Anderson, Security Engineering 2nd edition, John Wiley & Sons, 2008, ISBN 0470068523
- Bruce Schneier, Applied Cryptography, John Wiley & Sons, 1996, ISBN 0-471-11709-9

Online material

`https://moodle.bbk.ac.uk/`

5.9 Natural Language Processing and Information Retrieval (NLP)

Teaching Staff

Dell Zhang

Online material

`http://www.dcs.bbk.ac.uk/~dell/teaching/ir/`

Aims and Outline

The aim of this module is to introduce modern NLP & IR concepts and techniques, from basic text indexing to advanced text analysis. Both theoretical and practical aspects of NLP & IR systems will be presented and the most recent issues in the field of NLP & IR will be discussed. This will give students an insight into how modern search engines work and are developed.

Due to the explosive growth of digital information in recent years, modern Natural Language Processing (NLP) and Information Retrieval (IR) systems such as search engines have become more and more important in almost everyone's work and life (e.g. see the phenomenal rise of Google). NLP & IR research and development are one of the hottest research areas in academia as well as industry. This module will convey the basic principles of modern NLP & IR systems to students.

Syllabus

- Boolean Retrieval
- The Term Vocabulary and Postings Lists
- Regular Expressions and Text Normalization
- Dictionaries and Tolerant Retrieval
- Edit Distance
- Index Compression
- Scoring, Term Weighting and the Vector Space Model
- Evaluation in Information Retrieval
- Probabilistic Information Retrieval
- Language Models for Information Retrieval
- Language Modeling with N-Grams
- Spelling Correction and the Noisy Channel
- Text Classification, Naive Bayes, and Sentiment Analysis
- Vector Space Classification
- Logistic Regression
- Matrix Decompositions and Latent Semantic Indexing
- Vector Semantics
- Neural Nets and Neural Language Models
- Sequence Processing with Recurrent Networks

Assessment

By 2-hour written examination and practical coursework, weighting 80% and 20%, respectively.

Reading

- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, *Introduction to Information Retrieval*, Cambridge University Press, 2008, ISBN 0521865719. `https://nlp.stanford.edu/IR-book/`
- Dan Jurafsky and James H. Martin, *Speech and Language Processing*, 3rd ed draft. `https://web.stanford.edu/~jurafsky/slp3/`

5.10 Interactive Systems (IRS)

Teaching Staff

George Magoulas

Aims

The module aims to present a coherent introduction to the practical issues of creating interactive systems and products from a human-centred perspective. It covers fundamental concepts of interactive systems design, essential processes, and techniques for the design, development, and evaluation of interactive systems in different contexts.

Syllabus

The module covers theory, methods, and techniques used for the design of interactive systems. Indicative topics are listed below:

- Essentials of designing interactive systems: key concepts and how these are applied to different types of systems.
- The process of human-centred design
- Usability
- Techniques for designing interactive systems: understanding the requirements, prototyping and evaluating design ideas.
- Methods for understanding users
- Design methods
- Evaluation methods
- Task analysis
- Contexts for designing interactive systems: case studies of interaction design in contexts that are dominating the subject today.
- Web-based interactive systems
- Agents and avatars
- Mobile computing

Prerequisites

None.

Assessment

By 2-hour written examination and practical coursework, weighting 80% and 20%, respectively.

Reading

- D. Benyon, Designing Interactive Systems, 3rd edition, Pearson.

Online material

`http://moodle.bbk.ac.uk/`

5.11 Internet and Web Technologies (IWT)

Teaching Staff

Peter Wood

Aims

To provide students with an understanding of how network protocols work, particularly those used on the Internet, and the ability to present and manipulate information on the World Wide Web, with an emphasis on XML and JSON.

Syllabus

- Introduction to the Internet and its applications
- Web languages (e.g. HTML, XHTML, XML, JSON)
- Languages for defining Web document types (e.g. DTDs)
- Web query and transformation languages (e.g. XPath, XSLT)
- Client-side processing (e.g. using Javascript, jQuery)
- Server-side processing (e.g. using PHP)
- The transport layer (e.g. TCP, UDP)
- The network layer (e.g. IP, DHCP, ICMP)
- The link layer (e.g. Ethernet, ARP)

Prerequisites

A first module in programming.

Assessment

By 2-hour written examination and coursework exercises, weighted 80% and 20% respectively.

Reading

- Sas Jacobs, Beginning XML with DOM and AJAX. Apress, 2006, ISBN 1-59059-676-5.
- Anders Moller and Michael Schwartzbach, An Introduction to XML and Web Technologies. Addison Wesley, 2006, ISBN 0-321-26966-7.
- James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach (7th edition), Pearson, 2016, ISBN 1-292-15359-8
- Kevin R. Fall and W. Richard Stevens, TCP/IP Illustrated, Volume 1, Second Edition, Addison-Wesley, 2012, ISBN 0-321-33631-3

Online material

`https://www.dcs.bbk.ac.uk/~ptw/teaching/IWT/index.html`
`https://moodle.bbk.ac.uk`

5.12 Machine Learning (ML)

Teaching Staff

George Magoulas

Online material

`http://moodle.bbk.ac.uk/`

Aims

The module covers computational algorithms for intelligent information management, decision making and complex problem solving. It provides an introduction to machine learning methods such as neural networks, fuzzy logic, fuzzy clustering, natural computing, and covers basic concepts of feature selection and generalisation.

Syllabus

- Learning from data
- Feature selection and generalisation
- Supervised learning, unsupervised learning and clustering
- Fuzzy logic and fuzzy clustering
- Deep networks: architectures and learning algorithms
- Natural computing: genetic algorithms, evolutionary algorithms, swarm intelligence
- Advanced learning and evolution schemes

Prerequisites

No specific module is pre- or co-requisite but knowledge of mathematical concepts (algebraic concepts, vector, matrix, function and graph, gradient, trigonometry concepts, statistical concepts and the notion of probability), and data structures and algorithms, as taught in a typical undergraduate degree in computer science or engineering, is essential.

Assessment

By 2-hour written examination and practical coursework, weighting 80% and 20%, respectively.

Reading

- S. Marsland, Machine Learning: an Algorithmic Perspective, CRC Press.
- M. Negnevitsky, Artificial Intelligence: a Guide to Intelligent Systems, Addison Wesley.
- E. Alpaydin, Introduction to Machine Learning, MIT Press.

- Rojas, Neural Networks - A Systematic Introduction. Available online at: `http://page.mi.fu-berlin.de/rojas/neural/`
- S. Theodoridis, K. Koutroumbas, Pattern Recognition, Academic Press.
- Students will be directed to papers available online and other Web resources on the subject.

5.13 Mobile Computing and the Internet of Things (MCIT)

Teaching Staff

George Roussos

Online material

`http://www.dcs.bbk.ac.uk/~gr/muc/`

Aims

Students taking this module will:

- study the novel aspects of mobile, ubiquitous and pervasive computing systems
- study the principles, research problems and applications of the Internet of Things
- acquire a range of design skills for software development in Android
- acquire systems development experience with mobile and ubiquitous computing technologies
- help students develop self-study skills so that they can keep up with the rapidly changing technologies, tools and techniques in the area

Syllabus

- Wireless and mobile networks
- Routing and mobility aspects of IP networks
- Smartphone components
- Radio Frequency Identification (RFID) and the IoT
- Processing sensor streams
- Location sensing technologies
- Privacy in mobile location sensing systems
- Programming with Android

Prerequisites

Prerequisites: a first course in networks and a first course in software engineering (e.g. as taught in a typical UK undergraduate degree in computer science). Significant experience in Java programming including networking, data access and concurrent programming techniques.

Assessment

By 2-hour written examination and practical project, weighting 80% and 20%, respectively.

Reading

- J. Schiller, Mobile Communications, Addison Wesley, 2003.
- G. Roussos, Networked RFID: Systems, Software and Services, Springer, 2008.
- A. La Marca and E. de Lara, Location Systems: An Introduction to the Technology Behind Location Awareness, Morgan and Claypool Publishers, 2008.

5.14 Programming Paradigms and Languages (PPL)

Teaching Staff

Keith Mannock and Trevor Fenner

Online material

`http://moodle.bbk.ac.uk/`

Aims

To create a basic understanding of different programming paradigms and how they can be used in developing software. To provide with the opportunity to further develop the scope of their problem solving skills by studying advanced programming languages and new programming paradigms.

To enable students to understand the key differences between the various programming paradigms and the applicability of these paradigms to different programming problems. To enable students to deal with the changing requirements of the current day (and future) programmer where "polyglot" programming (the ability to mix and match languages and technologies appropriately) is fast becoming the norm.

Syllabus

- Definition of syntax and semantics
- Variable scoping and binding - lexical/static and dynamic
- Normal and applicative order evaluation, parameter passing mechanisms
- Storage allocation: run-time stack, heap storage and garbage collection
- Language support for abstract data types - arrays, maps, lists, etc.
- Data abstraction and libraries, collections, iterators
- Recursion, including tail recursion optimisation
- Continuations, threads, tasks, actors, co-routines, etc.
- The imperative (procedural and object-oriented) language paradigm
- The functional and logic language paradigms: lambda expressions, environments, predicates, etc.
- Process oriented languages
- Domain Specific Languages: scientific computation, symbolic computation, web and Internet computing.

Prerequisites

Introduction to Software Development with a distinction level grade, or Principles of Programming I and II, or a similar course or relevant experience (as determined by the module leader).

Assessment

By 2-hour written examination and practical coursework, weighting 80% and 20%, respectively.

Reading

- B. A. Tate, Seven Languages in Seven Weeks: A Pragmatic Guide to Learning Programming Languages, Pragmatic Bookshelf, 2010.
- K. Henney (ed), 97 Things Every Programmer Should Know: Collective Wisdom from the Experts, O'Reilly, 2010.
- M. L. Scott, Programming Language Pragmatics (3rd Edition), Morgan Kaufmann, 2009.
- A. B. Tucker and R. Noonan, Programming Languages (2nd Edition), McGraw-Hill, 2007.
- B. C. Pierce, Types and Programming Languages, The MIT Press, 2002.
- M. Gabbrielli and S. Martini, Programming Languages: Principles and Paradigms, Springer, 2010.

5.15 Semantic Technologies (ST)

Teaching Staff

Michael Zakharyaschev

Online material

`https://moodle.bbk.ac.uk/mod/page/view.php?id=595902`

Aims

Semantic Technologies provide easier ways to find, share, reuse and combine information. Prominent examples include Google's Knowledge Graph, Wikipedia's knowledge base sister Wikidata, numerous artificial intelligence projects from Alexa to Siri, and the healthcare ontology SNOMED CT. Various companies are exploring the use of their own enterprise knowledge graphs for improving internal knowledge management with the help of tools such as the graph database management system Neo4j, virtual knowledge graph engine Ontop, and the enterprise knowledge graph platform Stardog.

Semantic Technologies are a cross-cutting topic in computer science that involves aspects of data management, publication (exchange formats, data integration), knowledge organisation (ontologies), and advanced analytics (expressive query languages). This module gives an introduction to Semantic Technologies for data analysis, knowledge representation, and data management.

The aims of this module are to

- introduce the theoretical foundations of Semantic Technologies and Knowledge Graphs, including the W3C standard languages RDF/S, SPARQL, the Web Ontology Language OWL;
- provide the students with practical skills of modelling data using knowledge graphs, querying knowledge graphs, and building ontologies;
- overview the current applications of Semantic Technologies in Wikidata, health care, media management, and industry;
- demonstrate standard reasoning algorithms for classification of concepts in ontologies.

Syllabus

- Introduction to the module: Knowledge Graphs, Semantic Web, Linked Data, Wikidata; ontologies and their applications. Practical work: building ontologies with the Protege editor.
- Is XML a semantic technology? The tree model of XML documents, XML Schema, JSON. Querying XML documents with XPath. Practical work: (1) querying XML using an online XPath tool; (2) building a pizza ontology.

- Resource Description Framework (RDF). RDF Schema. Terse RDF Triple Language Turtle. Practical work: building Knowledge Graphs.
- SPARQL Query Language. Practical work: querying Knowledge Graphs using SPARQL; setting up and querying Apache Jena triplestore.
- Virtual Knowledge Graphs, ontology-based data access (OBDA). Practical work: ontology-based access to the IMDB database using Ontop.
- The OWL 2 Web Ontology Language. Practical work: understanding the OWL 2 language using Protege.
- Ontology engineering. OWL ontologies in life sciences and industry. Practical work: designing a travel agent's ontology.
- Reasoning with OWL. Introduction to formal semantics. Practical work: using the reasoner in Protege.
- Introduction to Description Logic and reasoning algorithms. Practical work: applying the reasoning algorithm.

Prerequisites

None

Assessment

By 2-hour written examination and coursework exercises, weighting 80% and 20% respectively.

Reading

- P. Hitzler, M. Kroetzsch and S. Rudolph. Foundations of Semantic Web Technologies. Chapman & Hall, 2009.
- E. Kendall and D. McGuinness. Ontology Engineering. Synthesis Lectures on the Semantic Web: Theory and Technology, 2019

5.16 Software Design and Programming (SDP)

Teaching Staff

Keith Mannock

Online material

`http://moodle.bbk.ac.uk/`

Aims

The main aim of the module is to provide students with the necessary skills for developing software utlising the object-oriented and functional programming paradigms utilising the Java programming language.

Syllabus

A selection from the following topics:

- The object model and how it is realised in various object-oriented languages (e.g., Java, Scala, Ruby, C++, ...)
- Further development the ideas of inheritance and polymorphism and abstraction
- Language features: inner classes, closures, higher-order functions, meta-objects, etc.
- The functional paradigm. Abstract data types, polymorphic types, static typing and type inference. Recursion and induction. List processing. Higher-order functions. Eager and lazy evaluation. Imperative features. Signatures, structures, functors, typeclasses, monads
- An introduction to Test Driven Design (TDD) and Behavioural Driven Design (BDD)
- The use of an Integrated Development Environment (IDE) for software development: e.g., editing, debugging, compilation, etc.
- Modularity, versioning, packaging, and managing the build process
- Design Patterns and Anti-Patterns and their application to software design
- The SOLID (Single responsibility, Open-closed, Liskov substitution, Interface segregation and Dependency inversion) approach to object oriented programming and design
- Code refactoring and analysis
- Concurrency and agents/actors

Please note that the materials for this module are presented using a combination of blended learning techniques together with the inverse curriculum approach to teaching.

Prerequisites

Introduction to Software Development with a distinction level grade, or Principles of Programming I and II, or an appropriate level of experience with a modern programming language otherwise (requires approval by the module leader).

Assessment

By 3-hour unseen written and practical examination and coursework exercises; weighting 80% and 20% respectively.

Reading

- Scala for the Impatient by Cay Horstmann, 2nd Edition, Addison-Wesley Professional, 2016. ISBN-10: 0-13-454063-8
- Scala Design Patterns, Patterns for Practical Reuse and Design by John Hunt, Springer International Publishing, 2013. ISBN-13: 978-3-319-02191-1
- Scala Design Patterns by Ivan Nikolov, Packt Publishing; 2016. ISBN-10: 1785882503

Through extensive course notes and example code. See the module webpage for details.

6 MSc Project

Please refer to the Moodle pages

`moodle.bbk.ac.uk/course/view.php?id=3292` (ACT programme)
`moodle.bbk.ac.uk/course/view.php?id=25178` (ACT project)

for a detailed description and guidelines.

Aims

Each student is required to undertake an individual project, under the supervision of a staff member, which should represent one-third of the student's effort for the degree (60 credits).

The main aims of the project are to offer students the opportunity to:

- develop a systematic understanding and critical awareness of an agreed problem relevant to the MSc programme as described in a project proposal form
- plan and execute a major piece of programming work appropriate to the MSc programme
- critically present existing approaches in the problem area, place their own approach in the wider area and evaluate their contribution
- gain experience in communicating complex ideas/concepts and approaches/techniques to others by writing a comprehensive, self-contained report.

Choosing a Project

Students are encouraged to come up with their own ideas for projects. In order to arrange supervision for the project, a student should discuss possible projects with the Projects Tutor, Programme Director or with the lecturer who seems the most appropriate for the topic.

The project must relate to one or more modules on the MSc programme. Additional requirements depend on the particular MSc Programme.

- For the MSc in Advanced Computing Technologies, MSc in Data Analytics, MSc in Information and Web Technologies and MSc in Learning Technologies, the project should build on advanced topics in computing technologies in order to develop a system and/or algorithms whose design is by no means obvious at the outset of the project.
- For the MSc in Data Analytics, the project should have a main focus in the area of data analytics.
- For the MSc in Information and Web Technologies, the project should have a main focus in the area of information and web technologies.

- For the MSc in Learning Technologies, the project should have a main focus in the area of learning technologies.

Syllabus

The main part of the project will be done by students on their own (supported by the supervisor). There is a small taught part of the module (see timetables in Section 4.2) in which the students are acquainted with

- how to formulate the objectives/aims of an MSc project
- how to write a project proposal
- how to organise and plan the project
- how to research literature
- how to write a project report.

Assessment

Written project proposal (of about 2000-3000 words) and written project report (of about 10,000 words), weighting 20% and 80%, respectively.

Reading

- As recommended by the supervisor.

Ethical Implications

All activities carried out by Birkbeck staff and students that involve one or more of:

- intervention or interaction with human participants;
- the collection and/or study of data derived from human participants;
- a potential impact on animals or the environment;

requires ethical consideration and approval.

For details see

`http://www.bbk.ac.uk/committees/research-integrity/GuidelinesonResearchwithEthicalImplications.pdf`

7 Administration and Assessment

For detailed College rules and regulations see

`http://www.bbk.ac.uk/registry/policies`

and, in particular,

`http://www.bbk.ac.uk/registry/policies/documents/CAS-regs-18-19.pdf`

Below we summarise the most relevant rules for the MSc Advanced Computing Technologies.

7.1 Requirements for the Award of the MSc

Each taught module available on the programme has a value of 15 credits while the project has a value of 60 credits, giving a total of 180 credits for the 8 taught modules and the project.

The programme regulations follow the College Common Awards Scheme. To pass a taught module or the project, a student must obtain a mark of at least 50%. A student may be offered a compensated fail mark when obtaining a mark between 40-49% in at most two 15-credit taught modules.

To be awarded the MSc, students must pass the project and at least 6 of their 8 taught modules; they must in addition obtain an average mark of at least 50% over the 8 taught modules, and at least 40% in any taught modules failed (30 credits maximum).

To be awarded the MSc with Merit, students must obtain an average mark of at least 60% over the taught modules and project.

To be awarded the MSc with Distinction, students must obtain an average mark of at least 70% over the taught modules and project.

In calculating the average mark, the taught module marks and project mark are weighted to reflect their credit value.

Students may be awarded a Postgraduate Certificate in Advanced Computing Technologies provided they have passed modules of at least 60 credits (no compensation for failed modules is allowed) or a Postgraduate Diploma in Advanced Computing Technologies provided they have passed modules of at least 120 credits (with at most 30 credits of compensation for failed modules).

Students who satisfy the MSc criteria may be made an award of MSc Advanced Computing Technologies. Students wishing to follow pathways leading to the awards MSc Data Analytics, MSc Information and Web Technologies or MSc Learning Technologies must satisfy the following additional criteria.

7.1.1 MSc Data Analytics

A minimum of 5 modules must be chosen from the following list:

- Advances in Data Management

- Big Data Analytics Using R
- Cloud Computing
- Data Science Techniques and Applications
- Data Warehousing and Data Mining
- Information Retrieval and Organisation
- Machine Learning
- Search Engines and Web Navigation

Additionally, the project should have a main focus in the area of data analytics.

7.1.2 MSc Information and Web Technologies

A minimum of 5 modules must be chosen from the following list:

- Advances in Data Management
- Component-Based Software Development
- Data Warehousing and Data Mining
- Information Retrieval and Organisation
- Internet and Web Technologies
- Mobile and Ubiquitous Computing
- Search Engines and Web Navigation
- Semantic Technologies

Additionally, the project should have a main focus in the area of information and web technologies.

7.1.3 MSc Learning Technologies

Any 8 modules on the MSc Advanced Computing Technologies programme can be chosen, but the project should have a main focus in the area of learning technologies (see `www.dcs.bbk.ac.uk/study/modules/project-in-learning-technologies/`).

7.2 Announcement of Results

The Examination Board meets in July to consider the results of the written exams and coursework, and in November to consider the results of the projects and to recommend the award of degrees.

Shortly after the meeting of the exam board, you will receive a letter from the Department about your results. Your results and grades will be officially confirmed by the College.

Students who have not paid their fees will not get any information about their examination results.

7.3 Choosing Modules

You will choose modules through the My Birkbeck portal. You have to make your choices by the specified deadline so that you can be enrolled on your chosen modules and have access to the material for them on Moodle.

Module availability is subject to timetabling constraints and student demand. In the event that a module is over-subscribed, available places will be allocated on a first-come, first-served basis determined by the date on which you submitted your module choices.

7.4 Examinations

All examinations are held during the daytime in May/June. Your examination timetable will be made available through the My Birkbeck portal.

7.5 Coursework

A number of modules require students to submit coursework as part of the assessment. Please consult the web page of the relevant module or contact the teaching staff of the module for particular details.

Submitted coursework must always be the students' own work, except where explicitly noted. Students are required to confirm in writing or via e-mail that each item of coursework submitted is indeed their own work.

The Department and College have strict guidelines and penalties associated with plagiarism, and routinely submit students' work to plagiarism detection services. More details are given in Section 7.10 on "Assessment Offences and Plagiarism" of this booklet.

College policy dictates how Departments will treat work that is due for assessment but is submitted after the published deadline. Any work that is submitted for formal assessment after the published deadline but before the cut-off date (normally ten working days after the deadline) is given two marks: a penalty mark of 50% for postgraduate students, assuming it is of a pass standard, and the "real" mark that would have been awarded if the work had not been late. Both marks are returned to the student. If the work is not of a pass standard a single mark is given.

If you are late submitting work that is to be considered for assessment, then you should submit a mitigating circumstances form (see Section 7.7) and provide written

documentation, medical or otherwise, to explain why the work was submitted late. The case will then be considered by the appropriate sub-board or delegated panel of the Board of Examiners. If no case is made then the penalty mark will stand. If the case is made and accepted then the examination board may allow the "real" mark to stand. The college policy about the provision of feedback on assessment is as follows.

`http://www.bbk.ac.uk/mybirkbeck/services/rules/Feedback-on-Assessment.pdf`

Unlike other disciplines, feedback on Computer/Data Science coursework/tests is often given in the form of model answers/solutions (e.g., program code), rather than textual comments.

7.6 Projects

Please consult the Moodle page for the MSc Advanced Computing Technologies Project:

`moodle.bbk.ac.uk/course/view.php?id=25178`

7.7 Mitigating Circumstances and Deferral

A Mitigating Circumstances claim should be submitted if valid detrimental circumstances result in:

- the late or non-submission of assessment;
- non-attendance of examination;
- poor performance in assessment.

If a student feels their circumstances warrant consideration by the Board of Examiners, they should notify the Programme Administrator, in writing, in advance, at the earliest opportunity (within 7 days of the assessment deadline or examination) using a *Mitigating Circumstances Claim Form*, which can be downloaded from:

`http://www.bbk.ac.uk/registry/policies/documents/MitCircs.pdf`

On the form, students should state whether the circumstances relate to non-attendance at an examination or late submission of an assignment and should include supporting evidence (e.g. a medical certificate giving the nature and duration of any illness). Students should be aware that discussing their claim with a member of staff does not constitute a submission of a claim of mitigating circumstances.

For a claim to be accepted a student must produce independent documentary evidence to show that the circumstances:

- have detrimentally affected their performance/submission/attendance in assessment or will do so;

- were unforeseen;
- were out of their control and could not have been prevented;
- relate directly to the timing of the assessment affected.

In **exceptional cases**, students may be permitted to defer the written exams and/or the project to the following year. They must apply by filling in a *Mitigating Circumstances Claim Form*. Students have to do this before **1st May** for exams and by **1st September** for the project.

A student who defers an element of assessment normally has to enter for that element the following year; usually no further deferrals are permitted. Simply not turning up for an exam or failing to submit a coursework or project, without permission to defer, will be considered to be the same as failing it, in the sense that it will count as one of the two attempts that you are permitted to make at passing that element.

7.8 Retake, Reassessment and Progression

One reassessment, and **only one**, is allowed for each element. You may be reassessed in a failed coursework, written exam or the project if your marks for that module are below 50%.

Students who fail an assessment and are awarded a reassessment opportunity have their reassessment mark subject to a cap of 50% for the reassessed element. The cap does not apply to a retake of a whole module or to students with accepted mitigating circumstances.

There are no special resit exams; students resit alongside the other candidates in May/June the following year. They normally do so a year after their first attempt. Where the syllabus has changed, we set a paper that is suitable for resit candidates, providing alternative questions where necessary. Note, however, that we do this only for candidates from the previous year, not from further in the past.

First year part-time students must normally pass at least three modules in order to proceed to the second year of study. Students who do not achieve this will not be able to complete their studies in two years. Instead they will have to spend at least one year as a repeating student, retaking the failed modules. Under normal circumstances this would take place the following year and students would not be allowed to take any new modules until they had passed the failed modules. However, because some modules on this programme are taught in the evenings on alternate years only, doing so would mean that students in their third year might have no new modules available to them in the evenings. This would have the effect of extending the duration of the degree to at least four years.

As a result, we permit students in such circumstances to enrol on new modules in their second year, postponing their second attempts at the failed modules to the third year. This is not ideal, but seems preferable to extending the duration of the degree.

7.9 Re-enrolment

Repeat students, i.e., students who have to retake some modules (and are not taking any new modules) will be charged pro-rata based on the number of credits they retake.

Assessment only students, i.e. those students who

- are being reassessed for coursework and/or examinations only
- have deferred their examinations and are not taking any new modules
- have deferred the project and do not require supervision (resubmitting only)

pay a reduced fee that will allow them access to College facilities (Library and workstation rooms). While deferred students are classed as assessment only they are allowed to attend lectures for revision purposes. They should formally seek the permission of module tutors to ensure classes are not oversubscribed.

Dissertation only students, i.e. students who retake the project with supervision, pay one third of full fees. Note that

- a student who has to resubmit the dissertation and be reassessed for examination or coursework will be progressed as dissertation only
- a student who has to resubmit the dissertation and also repeat modules will be progressed as repeat and fees are based pro-rata on the number of credits.

7.10 Assessment Offences and Plagiarism

See

`http://www.bbk.ac.uk/student-services/exams/assessment-offences`

for the College Policy on Assessment Offences.

One particular assessment offence is *plagiarism* that is defined as

> "[...] copying a whole or substantial parts of a paper from a source text (e.g., a web site, journal article, book or encyclopedia), without proper acknowledgement; paraphrasing of another's piece of work closely, with minor changes but with the essential meaning, form and/or progression of ideas maintained; piecing together sections of the work of others into a new whole; procuring a paper from a company or essay bank (including Internet sites); submitting another student's work, with or without that student's knowledge; submitting a paper written by someone else (e.g., a peer or relative), and passing it off as one's own; representing a piece of joint or group work as one's own."

Also,

> "[a] student who knowingly assists another student to plagiarise (for example by willingly giving them their own work to copy from) is committing an examination offence."

The College considers plagiarism a serious offence, and as such it warrants disciplinary action. This is particularly important in assessed pieces of work where plagiarism goes so far as to dishonestly claim credit for ideas that have been taken by someone else. The College also provides learning support for exams and assessments, please see

`http://www.bbk.ac.uk/student-services/learning-development`

and guidelines on plagiarism

`http://www.bbk.ac.uk/student-services/exams/plagiarism-guidelines`

8 Student Services

The College provides various services to students, see:

http://www.bbk.ac.uk/student-services

In particular, there are the Counselling Service, the Disability and Dyslexia Service, and the Mental Health Service. They provide specialist support to students.

8.1 Counselling Service

The Counselling Service

http://www.bbk.ac.uk/student-services/counselling-service

provides assistance to students who are experiencing emotional difficulties which may be impacting upon their studies or overall experience at Birkbeck.

8.2 Mental Health Service

Many students experience mental health difficulties at some point in their time at university. Whether you have a formally diagnosed psychiatric condition or other form of mental health difficulty such as anxiety or depression, we encourage you to seek support in your studies. Birkbeck's Mental Health Service

http://www.bbk.ac.uk/student-services/mental-health-advisory-service

is a first point of contact for students experiencing mental health issues at any stage during their studies.

8.3 Disability and Dyslexia Service

At Birkbeck we welcome students with disabilities. We aim to provide all of our students with a study environment that enables them to participate fully in our courses.
The Disability and Dyslexia Service:

http://www.bbk.ac.uk/student-services/disability-service

can provide advice and support to students with conditions that impact their ability to study, such as:

- specific learning difficulties (dyslexia, dyspraxia, dyscalculia, AD(H)D)
- sensory impairments (blind/partially sighted, deaf/hearing impaired)
- mobility conditions (including RSI, arthritis, neck back and knee conditions etc.)

- medical conditions (e.g. HIV, CFS, diabetes, cancer, chest and respiratory conditions etc.)
- autism-spectrum conditions (autism or Asperger's syndrome)

They can provide support during your studies including

- Your Study Support Plan
- The Disabled Students' Allowance
- Access to Learning Fund
- Charities and trusts
- Dyslexia screening test
- Government benefits
- Personal emergency evacuation plans
- Pager alert system
- Rest Room
- Toilet facilities
- Car parking
- Disability and Dyslexia Support in the Library and IT Services

9 Career Development

Most students are interested in developing their careers, either within their current field of work or in a completely new direction. The Careers Group, University of London

`http://www.thecareersgroup.co.uk/`

offers great expertise and experience in working with students and graduates of all ages and at all stages of career development.

The Careers and Employability Service

`http://www.bbk.ac.uk/careers/careers-service`

is our in-house service for enhancing career development and employability throughout your time at Birkbeck, from enrolment through to graduation.

There is also Birkbeck Talent, a professional recruitment service aimed exclusively at assisting Birkbeck students to find work whilst studying and after graduation. They work with top employers in and outside London to offer innovative internships, prestigious job vacancies and exciting graduate opportunities. To find out more please visit

`http://www.bbk.ac.uk/student-services/birkbeck-talent-service`

10 The Business Engagement Team

The School of Business, Economics and Informatics has a dedicated Business Engagement team to provide you with extra support. The team delivers the following initiatives to support you in your career aspirations:

10.1 Mentoring Pathways

Mentoring Pathways pairs successful applicants with industry professionals for individual advice and guidance. There are a number of places available for final year undergraduates and postgraduate students. We have partnerships with a number of key organisations and work alongside Birkbeck alumni who provide mentors. Applications open in the autumn. Please email `mentoring@bbk.ac.uk` or visit

`http://www.bbk.ac.uk/business/business-services/mentoring-external`

for more information.

10.2 Enterprise Pathways

Whether you are setting out in your journey as an entrepreneur or have already established a thriving business, we offer various pathways to support you. These include a non-credit bearing module with workshops once a month throughout the academic year, access to digital resources, enterprise boot camps and inspirational talks to help you to develop your ideas and network with other students. Please email `enterprise@bbk.ac.uk` or visit `www.bbk.ac.uk/enterprise`.

10.3 Keeping in Touch

You can also follow BEI on social media for information and conversations:

- Twitter: @BirkbeckBEI
- Facebook, Google+ and LinkedIn: Search 'BirkbeckBEI'

Please visit our website

`www.bbk.ac.uk/business/business-services`

for more resources and information.