Course Proposal Form

Please see Page 2 for instructions on which parts of this form to complete, whom to consult with to avoid unnecessary effort, and where to send the completed form.

Proposer(s): Milos Nikolic **Date: 05/12/2019**

Cover page: Basic permanent course information

Unless otherwise noted, items in this section are entered into EUCLID and **cannot** be changed without creating an entirely new course.

Course Name	Advanced Database Systems
Course Acronym *(used by the School only, e.g., for the Sortable Course List)*	ADBS
Course Level If the course is **only** available to MSc students, then it must be classed as Postgraduate. All other courses, regardless of level, are Undergraduate.	X Undergraduate __Postgraduate
Normal Year Taken	__UG1 __UG2 __UG3 X UG4 __UG5 __MSc
Also available in years *[This can be changed later if need be.]*	__UG1 __UG2 __UG3 __UG4 X UG5 X MSc
SCQF Credit Level Level 8 should normally be used for pre-honours courses. Level 10 should normally be used for optional UG3 courses (so UG4 students may also take them) and for courses aimed mainly at UG4 students. Level 11 should be used for courses aimed mainly at MSc students, whether or not UG4 students can also take them.	__7 __8 __9 __10 X 11
SCQF Credit Points	__10 X 20 __40 __60 __80 __Other:
Delivery Location	X Campus __On-line Distance Learning
Course Type	X Standard (default) __Dissertation __Online Distance Learning __Other (specify: Placement, Student Led Individually Created Course, Year Abroad)
Marking Scheme By default, courses use a numerical marking scheme. If you wish to use a grade-only marking scheme, your course proposal below should justify this.	X Standard (numerical) __Letter grade only

Guidance for remaining sections:

For an initial course proposal, please complete the **cover page and Section 1 (Case for Support)**, which asks you to describe the need for this course and to provide an overview of the course design, including the learning outcomes. **Please discuss your plans as early as possible with the head of Curriculum Review to avoid unnecessary effort.**

Send the form with these sections completed to the BoS Academic Secretary and head of Curriculum Review (listed on the BoS page) to obtain their comments before filling out the remainder of the form.

If a full proposal is invited, please complete the remaining sections and send to **iss-bos@inf.ed.ac.uk**.

2. Student-facing course description and additional feedback and assessment information.
This section provides most of the information students see in the DRPS entry for this course, as well as related details for BoS consideration.

3. Further information for BoS consideration: sample materials.

4. Additional Course Details required for DRPS. *[Administrative information such as delivery timing and prerequisites.]*

5. Placement in degree programme tables. *[Required for all level 9-11 courses; used to determine where the course will be added to existing degree programme tables.]*

6. Comments from colleagues. *[All course proposal should be sent to relevant colleagues in the area as well as to the appropriate year organizer and BoS Academic Secretary for comment in good time before the BoS meeting. Use this section to indicate what feedback has been solicited and received.]*

Colour coding and item-by-item guidance:

Guidance is provided in italics for each item. Please also refer to the guidance for new course proposals at http://www.inf.ed.ac.uk/student-services/committees/board-of-studies/course-proposal-guidelines. Examples of previous course proposal submissions are available on the past meetings page http://web.inf.ed.ac.uk/infweb/admin/committees/bos/meetings-directory but note that the proposal form was updated in Jan 2019.

Sections in gold are for student view and are required before a course can be entered into DRPS. You must complete these sections even if your course has already been approved based on other documentation.
Sections in orange are for School use but are still required for all courses (even those that have already been approved based on other documentation).
Section in gray are for consideration by the Board of Studies. They are normally required for all new course proposals but may be omitted in some circumstances (e.g., for invited course proposals) if you obtain permission in advance.

1. Case for support

This section is for consideration by the Board of Studies. The final two boxes (Learning Outcomes, Graduate Attributes) will also go into the student-facing course description.

Overall contribution to teaching portfolio and relation to existing curriculum

Please explain (a) what motivates the course proposal (e.g. a previous course having become outdated/inappropriate, an emergent or maturing research area or new research activity in the School, offerings of our competitors) and (b) how it relates to existing courses and degree programmes (including any prerequisite courses). Every new course should make an important contribution to the delivery of our Degree Programmes.

This new course is a revised and extended version of the Advanced Databases course (INFR11011). There are three main reasons motivating this revision:

1. Bring the course content up to date with current trends in database systems.

The Advanced Databases course (INFR11011) has not seen significant changes for at least 10 years. The course content is still entirely based on the classical database textbook by Ramakrishnan and Gehrke from 2003. Although this textbook remains a valuable reference nowadays, the data management landscape has changed significantly in the meantime: we have seen new database architectures such as column stores and in-memory database systems, new query processing techniques such as query compilation, new indexing structures, new concurrency control protocols, and recent explosions of cloud databases, scientific databases, and systems combining databases and machine learning. Studying these new topics would allow students to get a better understanding of how modern database technologies are used in industry and research nowadays. This is a valuable learning outcome in our opinion.

2. Cover cutting-edge research topics

We believe that postgraduate students should be exposed not just to classical textbook material, but also to recent research literature to help them develop critical thinking and research skills. This revised course will include several lectures with discussions of recent research papers, which are normally not covered in any textbook yet. Examples of such topics include cloud databases, systems for in-database machine learning, scientific (array) databases, etc.

3. Introduce practical lab sessions

The current edition of Advanced Databases has no practical labs despite the implementation-oriented nature of the course (that has been the case for the last 4-5 years at least). To rectify this issue, this revised course will introduce lab sessions to allow students to gain practical experience with implementing different components of a prototypical database system.

The proposed changes in a nutshell:

- Update the content of the course to better reflect the modern trends in database management systems.

- Include lab sessions to give students practical experience with implementing various parts of a database system.

- Extend the course from a 10-credit to a 20-credit course. Covering the aforementioned new topics along with the existing material would not be possible within the scope of a 10-credit course.

- Change the name of the course to “Advanced Database Systems”. This name better reflects the system-oriented nature of the course and would hopefully mitigate confusions seen in previous years among students, who often think of “Advanced Databases” as being an advanced SQL course, which it is not.

This revision is part of an ongoing effort in the Database Group to revise the database courses offered by the School. The main objective is to offer a coherent and comprehensive curriculum that covers systems and foundational aspects of databases both at an introductory and advanced level. This revised course on Advanced Database Systems builds upon the introductory database course (former Database Systems) and complements the postgraduate course on the principles of data management (former Advanced Topics in Foundations of Databases). The three database courses have been revised in a coordinated effort.

Target audience and expected demand

Describe the type of student the course would appeal to in terms of background, level of ability, and interests, and the expected class size for the course based on anticipated demand. A good justification would include some evidence, e.g. by referring to projects in an area, class sizes in similar courses, employer demand for the skills taught in the course, etc

This course targets UG4, UG5, and MSc students in computer science who are interested in studying how modern database systems function internally. The students are expected to have a basic familiarity with database query languages (SQL and relational algebra) and a good level of programming experience (in any programming language). The course will cover both foundational topics (e.g., classical algorithms and data structures in DB systems) and recent topics from research papers. These topics will be essential to students who plan to work in the data science industry but also to students who want to do research in the area of data management. The programming component of this course will allow students to develop first-hand experience working with DB systems that goes beyond writing SQL queries.

The enrolment for Advanced Databases has been steadily around 55-60 students over the past few years. With the change to a 20-credit course, there is a risk that these numbers will go down slightly. On the other hand, we believe the course will be more attractive to students as the proposed new topics are highly sought-after on the Data Science job market (e.g., cloud databases, ML in databases, novel indexing data structures).

Anticipated Resource Requirements

Estimate how much lecturing, tutoring, exam preparation and marking effort will be needed in steady state, and any additional resources needed to set the course up initially. Provide estimates relative to class size where applicable and discuss how support staff will be recruited and supervised, if the class is likely to be very large. Please mention any scaling limits due to equipment or space. If equipment is required, say how it will be procured and maintained.]

The course organiser will deliver 24 lectures, 4 in-class tutorials, and 2 revision sessions. One or two guest lectures will be given by people from industry. The course will include four 1.5-hour lab sessions in weeks 3, 5, 7, 9 (roughly) and one assessed programming assignment. Both the practical lab material and the programming assignment will be set by the course organiser.

Assuming the class size of up to 50 students:

- The course organiser will prepare and mark the final exam; for larger class sizes, the course organiser will need help from additional markers (one marker per 25 extra students).
- One demonstrator will run 3-hour lab sessions per week and participate in coursework review meetings. The demonstrator will be recruited from existing PhD and MSc students. The expected workload for the demonstrator will be 30 hours.
- The course organizer and the demonstrator will mark the coursework assignment.

Quotas, special arrangements or unusual characteristics

Please specify if this course requires any special arrangements such as quotas or other registration arrangements; is a collaboration with another school or institution, or has other atypical characteristics that may affect finances or student registration. Further justification/information may be requested for such courses.

Narrative description of the course aims and structure

Please describe the main goals of the course and how the course design will allow students to achieve those goals. This section should be consistent with the student-facing information provided below, but should provide additional information to help colleagues at BoS understand the vision and structure of the course. This description may refer to the learning outcomes and graduate attributes (next two boxes) and should explain how activities such as tutorials, labs, or in-lecture activities will support them, and how the proposed assessments will assess them.

For courses that are important pre-requisites for other courses, this section may also provide content/syllabus information which is too detailed for the student-facing description, such as a lecture-by-lecture syllabus.

This course extends the syllabus of the Advanced Databases course with new topics focused on modern database system architectures (in-memory databases, column stores, cloud databases) and new data processing paradigms (parallel, distributed, streaming); none of these are covered by the existing Advanced Databases course. Another goal of this course is to expose students to hot research questions (e.g., ML over databases, write-optimised data structures).

The course will include practical lab sessions where students will have a chance to work on concrete mini-projects. Each project will ask students to implement one component of a prototypical database system, for instance, to implement the buffer manager, a particular join algorithm, or a B+ tree index. The projects will require programming in C++ and will have increasing difficulty to allow students to familiarise with the language and environment.

Summary of Intended Learning Outcomes (MAXIMUM OF 5)

List the learning outcomes of the course. These must be assessable (i.e., observable), so must specify what the student should be able to do concretely, not simply what they should "understand". Use concrete verbs that indicate (a) what type of assessment would be appropriate, and (b) what level of knowledge/thinking is expected (from recall to analysis to novel creation). ***Example verbs:*** *define, explain, implement, compare, justify. Assessments (described later) should be tied to the learning outcomes.*

Outcomes should typically focus more on the types of thinking/skills developed than on the detailed course content, and the level of thinking should be appropriate to the level of the course: outcomes for a Level 11 course should include more higher-level thinking skills than for a Level 8 course. Further guidance on writing learning outcomes can be found at https://www.ncl.ac.uk/ltds/assets/documents/res-writinglearningoutcomes.pdf

On completion of this course, the student will be able to

1) Describe how database management systems function internally. Interpret and comparatively criticise database systems architectures.

2) Implement major components of a database management system and analyse their performance.

3) Analyse and compare the fundamental query evaluation and concurrency control algorithms. Identify strengths and weaknesses of query evaluation plans. Optimise query evaluation plans.

4) Identify trade-offs among database systems techniques and contrast distributed/parallel techniques for OLTP and OLAP workloads.

Graduate Attributes, Personal & Professional Skills

List the personal attributes and generic transferrable skills this course will help develop. Examples include

Cognitive skills: *problem-solving, critical/analytical thinking, handling ambiguity*

Responsibility, autonomy, effectiveness: *independent learning, self-awareness and reflection, creativity, decision-making, leadership, organization and time management, flexibility and change management, ethical/social/professional awareness and responsibility, entrepreneurship*

Communication: *interpersonal/teamwork skills, verbal and/or written communication, cross-cultural or cross-disciplinary communication*

2. Student-facing course description and additional feedback and assessment information

Except where noted, all fields are required and will go into the DRPS entry for the course (for use by students). Important: any text in DRPS is effectively a contract with students, so should not include details that are likely to change from year to year.

Summary Description *Provide a brief official description of the course, around 100 words. This should be worded in a student-friendly way, it is the part of the descriptor a student is most likely to read. If this course replaces another course, please say so in this summary.*	Database management systems are at the core of computer applications that need to store, manipulate, and query data. This course takes a deep dive into how modern database systems function internally, from studying their high-level design to understanding the underlying data structures and algorithms used for efficient data processing. The course covers a range of data management techniques from both commercial systems and cutting-edge research literature, enabling students to apply these techniques to other fields of computer science. The covered topics include database architecture, storage manager, data models (row, columnar), indexing (tree-based, hash tables), transaction processing (ACID, concurrency control), crash recovery, parallel architectures (multi-core, distributed), cloud databases, and ML in databases. These topics will be valuable to students who plan to work in the data science industry but also to students who want to do research in the area of data management. The programming component of this course will allow students to develop first-hand experience working with database systems that goes beyond writing SQL queries. This course replaces Advanced Databases (INFR11011).
Keywords *Give a list of searchable keywords.*	Database system architecture, query optimization, query evaluation, indexing, transactions, parallel databases
Course Description *A more detailed student-facing description of the course, which should normally include (a) a more in-depth academic description of the learning aims, nature and context of the course, (b) a rough outline of the content or syllabus, often as bullet points, and (c) a description of how the course will be taught, how students are expected to engage with their learning and how they will be expected to evidence and demonstrate their achievement of the intended learning outcomes.]*	Lectures will cover roughly the following topics: - Database systems architectures, row stores and column stores, OLTP vs. OLAP, in-memory database systems. - Storage: secondary-storage devices. - Indexing: tree-based and hash-based techniques, multi-dimensional indexing, learning indices from data. - Write-optimised data structures: LSM trees, LSM hash tables, B^eps trees. - Query evaluation: sorting and join processing, selection, projection, aggregation, query compilation.

	- Query optimisation: cardinality estimation, cost-based query optimisation, dynamic programming, rule-based optimisation, learning query plans. - Transaction management: ACID properties, concurrency control, locking and multi-version protocols, crash recovery. - Distributed database systems: parallel query evaluation, distributed transaction processing. - Data warehousing and decision support: OLAP, materialised views, incremental view maintenance. - Stream processing systems, data streaming algorithms. - Scientific (array) databases, cloud databases, database systems for machine learning. In practical lab sessions, students will implement various components of a prototypical database system. The labs will require programming in C++ and will have increasing difficulty to allow students to familiarise with the language and environment.
Assessment Weightings: *These should correspond approximately to the proportion of learning outcomes that each component assesses. More than 30% coursework requires specific justification.* *The expectation for a 10pt course is 20% coursework with the equivalent of one 15-20hr assessed assignment (but possibly split into smaller pieces). See 'components of assessment' below.*	Written Exam 70% Practical Exam 0% (*for courses with programming exams)* Coursework 30%
Further Assessment Information *Provide any further information that should go on DRPS for students. E.g., if the assessment includes required group work or if students must pass some individual component of assessment as well as the course overall.*	
Components of assessment and time spent on assignments (for BoS only) *If not already included in the course narrative description, please describe the type of assessments (oral presentation, report, programming, etc) and* ***how each component of assessment will assess the intended learning outcomes****. Where coursework involves group work, it is important to remember that every student has to be assessed individually for their contribution to any jointly produced piece of work.* *Also estimate* ***how many hours*** *students will spend on assignments. Please see the School policy on Workload and Assessment, which states that students should not be expected to spend*	The coursework includes one programming assignment worth 30% where students will design and implement a data processing task, experimentally evaluate their work, and write a report on their findings (learning outcome 2). Students are expected to spend around 40 hours working on this programming assignment.

more than 6-7 hrs/wk per 10 credits, including contact hours. *Note that it often desirable to include formative assignments which are not formally assessed but submitted for feedback, often in combination with peer assessment.*	
Feedback Information *Provide a high-level description of how and what type of feedback will be provided to students, for inclusion in DRPS.*	Students will receive feedback from demonstrators during practical sessions and from the instructor on at least one piece of formative assessment similar to the final exam.
Additional Feedback Information (for BoS use only) *If not already included in the course narrative, provide further details on planned feedback arrangements. This includes how course feedback is solicited from the class and responded to, as well as what feedback students will get (either on work that contributes to their final mark, or not).* *The University is committed to a baseline of principles regarding feedback that we have to implement at every level, and the School encourages submission of at least one piece of written work for formative feedback.* *In general, formative feedback:* • *Should say how students can improve.* • *Need not be on individual work (e.g., consider a lecture or document summarizing common issues.)* • *Can include oral feedback during labs/tutorials* • *Can include feedback from peers* • *Clickers/TopHat/equivalents can provide in-class feedback for both students and lecturer.* • *Is returned in time for other forms of assessment to which it relates, to allow feedforward.*	The course will include four in-class sessions where students will get oral feedback on tutorial sheets and recent exam papers. Students will also get feedback from demonstrators during lab sessions.

Breakdown of Learning and Teaching Activities

State how many hours students spend on each part of the course. The total should be 10 x course credits, but please also see the School policy on Workload and Assessment.which states that students should not be expected to spend more than 6-7 hrs/wk per 10 credits, including contact hours.

Assume 10 weeks of lectures slots and 10 weeks of tutorials, but these need not all be used. As a guideline, a 10-pt course typically has 18-20 lecture hours, but should have only around 15 lectures of examinable material; the rest should be used for guest lectures, revision sessions, introductions to assignments, etc.

Contact hours

Hours	Type
28	Lecture Hours
0	Seminar/Tutorial Hours
0	Dissertation Project Supervision Hours
6	Supervised practical/Workshop/Studio hours
2	Feedback/Feedforward hours
0	Summative assessment hours
2	Revision Session Hours

Non-contact hours

Hours	Type
162	Directed Learning & Independent Learning hours

Total hours: 200

Reading List/Learning Resources

You are encouraged to create resource lists using LEGANTO

- Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems (Third Edition) McGraw-Hill 2003.
- Research papers on the topics of the syllabus.

3. Further information for BoS consideration: sample materials

A full proposal for a new course must include examples of exercises and assessment. Please provide these below, along with publicity information if the course is to be advertised outwith the School.

Course information and publicity *The course web page (typically the Learn landing page) will be linked from the Sortable Course List, and information such as timetables and assignment deadlines must be made available prior to the start of the academic year. Please specify here if any additional info/publicity is needed for your course: typically only if it is aimed largely at non-SoI students.*	
Sample tutorial/lab sheet questions *Provide a list of tutorial questions and answers and/or samples of lab sheets. These need not be fully fleshed out but should indicate what sort of exercises will be provided to help students learn the material.*	The tutorial questions from the current edition of Advanced Databases will be revised and extended to include the new material covered in this course. The lab sheets will include three programming exercises asking students to implement various components of a prototypical database system developed for educational purposes.
Sample assessment materials *If the course is primarily assessed by **exam**, provide a sample exam question with model answers. Any non-standard exam format must be justified. The online list of past exam papers gives an idea of typical and alternative exam formats: http://www.inf.ed.ac.uk/teaching/exam_papers/.* *If the course is largely or primarily assessed by **coursework**, provide a sketch of a possible assignment with an estimate of effort against each sub-task and a description of marking criteria.*	The exam will be similar to that of Advanced Databases. The exam questions will be revised and extended to include the new material covered in this course. https://exampapers.ed.ac.uk/search/INFR11011
Any other relevant materials *Include anything else that is relevant, possibly in the form of links. If you do not want to specify a set of concrete readings for the official course descriptor, please list examples here.*	

4. Additional Course Details for DRPS

Except where otherwise noted, these fields are required for entry into EUCLID and will be visible to students in the DRPS entry.

Planned Academic Year of Delivery *(The first year you anticipate the course running, e.g. AY 2019-20)*	2020-21
Course Organiser *(By default, the course proposer)*	Milos Nikolic
Intended Delivery Period	___Semester 1 _X_Semester 2 ___Full Year ___Summer ___Other (please specify):
Timetable considerations/conflicts *For School use. Please specify any constraints to be considered (e.g. overlap of popular combinations, other specialism courses, external courses etc). Include whether the semester delivery is constrained or could be flexible.*	
Is this course available to visiting students?	_X_Yes (default) ___No **If no, please provide a justification here:**
Required pre-requisite courses *Use sparingly: these are enforced in PATH and can only be waived by approval from the School's Curriculum Approval Officer. Note that cross-year required pre-requisites may prevent MSc students from registering; consider using recommended pre-requisites or "other requirements" instead.*	_X_No ___Yes (please specify full course name(s) and code(s)):
Recommended pre-requisite courses	___No _X_Yes (please specify full course name(s) and code(s)): Database Systems (INFR10070)

Required co-requisite courses *Specify any courses that must be taken in parallel with the existing course. Note that this leads to a timetabling constraint that should be mentioned elsewhere in the proposal.*	_X_ No ___Yes (please specify full course name(s) and code(s)):
Prohibited Combinations *Specify any courses that may not be taken in combination with the proposed course].*	___No ___Yes (please specify full course name(s) and code(s)):
Other Requirements/Additional Information *This information is often used by MSc students and students from other Schools to see if they have appropriate background without having done our School's courses. So please avoid course titles, instead list specific knowledge and skills (such as mathematical concepts, programming ability or specific languages, etc).* *Also list any other constraints on registration, for example: "Only available to 4th Year Informatics students including those on joint degrees." or "This course is open to all Informatics students including those on joint degrees, and to students in the School of Mathematics. Other external students whose DPT does not list this course should seek permission from the course organiser."*	___No _X_ Yes (please specify): This course is open to all Informatics students including those on joint degrees. For external students where this course is not listed in your DPT, please seek special permission from the course organiser. The course assumes an understanding of algorithms and data structures (e.g., quick sort, merge sort, binary trees, hash tables, big-O notation). It is recommended that students have basic knowledge of databases (e.g., SQL and relational algebra) although key concepts will be covered in the course. A good level of programming is assumed and will not be covered during lectures. The coursework will involve implementing different parts of a database system in C++.
Visiting Student Pre-requisites	_X_ Same as "other requirements" ___Different than "other requirements" (please specify):

5. Placement in degree programme tables: for level 9-11 courses only

This section is for consideration by the Board of Studies and will be used later by ITO to determine where the course will be added to existing degree programme tables.

Is this course restricted to students on a specific degree? *E.g., some courses are only available to students on a specific CDT or MSc.*	_X_ No ___Yes (please specify and provide justification):
Is this course compulsory for students on any degree(s)?	_X_ No ___Yes (please specify and provide justification):
Any issues for part-time students? *Normally, part-time students have access to the same courses as full-time students on the equivalent degree. If you anticipate any problems with this, please specify here.*	

For optional courses:

If this course is available but non-compulsory for students on various degrees (most courses), please fill in this section. The choices here determine where the course appears in degree programme tables (DPTs) and the 2-3 character tags are displayed in the Informatics sortable course list.

Should this course be tagged as 'ML' (machine learning foundations and methods)? *Courses with the ML tag are typically very high-demand and most degrees limit the number of ML credits. If your course might appeal to a similar audience but draw off students from these large courses, please select 'no' and choose one of the tags below.*	_X_ No ___Yes
If you chose 'no', please choose at least one of the following tags… *Ideally, select exactly one, unless there is a good argument for more than one. These three are used in various combinations for many of our degrees.*	_X_ **FSS** (CS foundations, systems, and software) ___**AIA** (artificial intelligence applications and paradigms) ___**COG** (cognitive science: including HCI and NLP courses, but not most other AI courses. Please restrict to courses most relevant to natural cognition.)
…and also tick if any of the following tags or categories apply. *Do not tick any of these if you selected 'ML' already.*	___**NS** (natural systems: e.g., computation by or about biological or social systems. Many COG courses are also NS. This tag is mainly relevant for MSc in Informatics.) _X_ **SE** (software engineering: including courses that are highly relevant to SE degrees. All SE courses should also be FSS. This tag is mainly relevant for UG SE degrees.)

	_X_ Databases and data management systems (used for Data Science MSc and MSc(R)) __ Unstructured data and applications (used for Data Science MSc and MSc(R)) __ Level 11 Security courses (used for Security MSc)
If you are not sure which tags are most appropriate or have other questions about this section, please note any comments/issues here.	

6. Comments from colleagues

All course proposal should be sent to relevant colleagues in the area as well as to the appropriate year organizer and BoS Academic Secretary for comment in good time before the BoS meeting. Please indicate here what feedback has been solicited and received.

Additional Comments *Summarise any comments received from relevant individuals prior to proposing the course. If you have not discussed this proposal with others please note this.*	
Year Organiser Comments *Year Organisers are responsible for maintaining the official Year Guides for every year of study, which, among other things, provide guidance on available course choices and specialist areas. The Year Organisers of all years for which the course will be offered should be consulted on the appropriateness and relevance on the course. Issues to consider here include balance of course offerings across semesters, subject areas, and credit levels, timetabling implications, fit into the administrative structures used in delivering that year.]*	
BoS Academic Secretary Comments *Proposals must be checked by the Secretary of the Board of Studies prior to discussion at the actual Board meeting. This is a placeholder for their comments, mainly on the formal quality of the content provided above.*	