Computer Science Accreditation Council (CSAC)
Program Accreditation
Institutional Questionnaire and
Self-Assessment Report
For

Computer Science, Software Engineering, and Interdisciplinary programs

Incorporating Outcomes-Based Principles

June 9, 2011

THIS IS BEING CIRCULATED TO CACS-AIC
PLEASE DO NOT USE TO INITIATE YOUR ACCREDITATION. OBTAIN THE LATEST VERSION FROM THE CIPS ACCREDITATION OFFICE, SINCE IT MAY BE UPDATED

Table of Contents

1 Introduction

The questionnaire provides essential qualitative and quantitative input for the evaluation team as part of the overall accreditation process. It also serves as a tool for self-assessment.

The accreditation process consists of the following steps:

- Request by the institution for evaluation of its program(s);
- Completion and submission of this questionnaire and supplementary material described later;
- On-site visit by a team of program evaluators;
- Submission of a draft report by the team to the dean and/or program director approximately 6 weeks after the visit. The institution may respond to team's finding and/or note errors of fact or findings and respond with 14 days.
- Formal consideration by the Computer Science Accreditation Council resulting in a decision on accreditation.

In the evaluation process for accreditation as a Computer Science, Software Engineering, or Interdisciplinary program, the principal emphases are placed on the program administration, faculty and its qualifications, the students and graduates, the curriculum, and the resources (physical, fiscal, and human). Although the criteria are intended to specify minimum requirements, they also allow for and encourage two important characteristics of programs in computing. These characteristics are the diversity of programs that exist among the various institutions and the innovative features that have been typical in these programs.

1.1 Interdisciplinary programs

Definition of an interdisciplinary program: A program where the material in computer science is combined with material from one or more, frequently quite different disciplines, to form an interdisciplinary or joint degree program. Examples include programs with sciences such as physics, biology or humanities such as visual arts. Additional possibilities include double majors with computer science and mathematics, computer science and business. Joint programs with tightly integrated sets of courses from the different disciplines and a set of 'building blocks' (often called 'majors') that can be used to create a variety of interdisciplinary combinations also fall within this definition. The interdisciplinary criteria do not apply to programs in which students take courses in CS as a secondary discipline (e.g. a minor in CS). In this questionnaire the term "Other Discipline" is used to refer to the key non-CS topic(s) in interdisciplinary programs.

1.2 Completing this form

To simplify the task of the Council it is suggested that you complete the questionnaire by simply editing a copy of this document (Sections 3 and onward). Your responses will consist of filling in tables found below, and typing answers to certain free-form questions, also found below. The

free-form answers should be in **boldface,** so they can be easily located. To assist you we have placed the word '**Answer**' or '**Additional-comments**' wherever a free-form answer is requested. However, if you wish, you may submit your responses as a separate document. Please be sure to include question numbers with your responses. If an alternate format is used, at least all of the information requested in this questionnaire must be included.

This form is available electronically through the web at TOBEDETERMINED. A web template is also available from this site if you wish to host the questionnaire and related materials via a private secure website on your servers.

Whether using the web template or not, you are encouraged to submit the completed questionnaire and related documentation in electronic format.

1.3 Sample Questionnaire Available

To assist departments in preparing for accreditation, especially in the transition to outcomes-based accreditation, we have prepared a sample questionnaire for a fictitious university. This can be obtained from the same website as this form. The sample does not have every detail completed, but it should show what is generally expected.

1.4 Draft Accreditation Report Production Guidelines

The following draft accreditation report production guidelines are used by the accreditation team. The Department can expect to receive the draft report within the noted time lines.

Regular review single program	-	Draft ready within 6 to 8 weeks
Regular review multiple programs	-	Draft ready within 8 to 10 weeks

2 Supplemental information required

2.1 Material required at the time of the application

In addition to completing this questionnaire, you must also submit of each of the documents listed in the following table at the time of the application, preferably in digital form or as web links. If no digital version exists, then please submit five hardcopies of these materials.

Information required with the application	Check
The official **University calendar**. A pointer to the website is almost always sufficient.	
Teaching assignments for the current academic year.	
Complete or abbreviated **CVs for all faculty**[1], including information on grants received, recent evidence of scholarship, and professional involvement.	
Recruitment Brochures and Materials; examples of all materials other than the official Calendar or Program Handbook which are used for recruitment purposes should be included, e.g. brochures, flyers, data-sheets etc.	
Scholarships and Bursaries; list all scholarships and bursaries available to students enrolled in the program. Include the criteria associated with the award as well as the amount on the award.	
Course Outlines; provide copies of all Computer Science and/or Software Engineering course outlines for every course offered to students in the program(s). Course outlines are *not required* for disciplines outside of Computer Science and Software Engineering unless such a course is essential for meeting one of the graduate attributes discussed in Section 5.5. Course outlines should ideally contain learning objectives that can be mapped to the graduate attributes.	
All official **department handbooks** describing the undergraduate programs to be considered.	
All **guidance materials**, such as program checklists, distributed to undergraduate students	
Any relevant **salary policy documents, collective agreements**, and so forth	
Any publications describing **physical, computing, library, and other physical resources**.	
Any publications describing the **Department's organization or operations**, such as the latest annual report, descriptions of internship programs, and so forth.	

Additional materials that you feel may be helpful for the accreditation team should also be submitted. Where possible, as much of the material as possible should be submitted in digital

[1] CV's of all faculty should be submitted, including Computer Science and/or Software Engineering as well as, in the case of interdisciplinary programs, the faculty from any Other Discipline(s). CVs of the CS/SE faculty members will be examined, and the team will meet with groups of such faculty members. The faculty CV's for the Other Discipline(s) will only be used by the accreditation team to ensure that, in general, the faculty members in the Other Disciplines have appropriate backgrounds and teaching loads. If, due to the structure of an interdisciplinary program, the number of faculty in the Other Discipline(s) would be open-ended or extremely large, then a representative selection may be submitted.

form, using CD-ROMs or memory sticks. This is so they can be assessed while not connected to the Internet. Four copies of all materials should be provided, plus one extra copy if both computer science and software engineering programs are being evaluated. Examples of the types of materials to provide indicated throughout the questionnaire.

When new or updated material becomes available between the time the questionnaire is assembled and the date of the visit, it should be provided to the team members in advance or on arrival at the campus, with a copy to CIPS Accreditation Secretariat.

2.2 Material to be provided at the time of the visit

Specific material for each Computer Science and/or Software Engineering course must be made available to the visit team in their meeting room during the accreditation visit. Much of this can be provided electronically if desired. Confidential materials provided electronically should use a password-protected website, or else in five copies in the form of CD-ROMs or USB drives. Since not all members of the visit team may have suitable computers, a couple of computers should be made available in the meeting room with clear instructions regarding how to access any electronic material.

Above all, it is important that the visit team have quick and easy access to the material.

The following table should be replicated for every Computer Science and Software Engineering course. Note that in addition to the material listed in this table, course outlines and details of the graduate attributes will already have been provided with the original application for accreditation.

Per-course information requirements at time of visit	Check
Course Code:	
Course Name:	
Sample assignments	
Sample midterms, tests or quizzes	
Sample final examination	
Course textbook (provide a physical copy for examination or electronic access to an E-Book)	
Course notes given to students (a link to the course website will suffice if the notes are there)	

Demonstrable evidence must be provided regarding how this course ensures certain graduate attributes are fulfilled. This evidence will include samples of the actual work of students (projects, marked tests and exams, etc.).

Note that, other than physical textbooks, much of the above material *could* be provided at the time the application is submitted. However it is only *required* at the time of the visit.

3 The University Environment

3.1 Size of the academic unit

Please complete the following table to indicate the size of your operations (i) in absolute terms, (ii) as an approximate percentage of the operations of all units reporting to the same Dean (typically a *Faculty*), and (iii) as an approximate percentage of the total University operations.

	Your Unit	as % of Faculty	as % of University	Comments or clarifications
Faculty FTE	_	%	%	_
Undergraduate full-time enrollment	_	%	%	_
Undergraduate part-time enrollment	_	%	%	_
Graduate full-time enrollment	_	%	%	_
Graduate part-time enrollment	_	%	%	_
Annual student-courses taught at all levels	_	%	%	_
Total salary budget	_	%	%	_
Total non-salary budget	_	%	%	_

3.2 University structure

How do the Department's programs fit into the structure of the University? Please include explicit references to the University calendar or other submitted materials.

Answer

4 Faculty

4.1 Financial resources

Please complete the following table to indicate the financial resources allocated in support of Computer Science and/or Software Engineering faculty members:

	Minimum for Faculty Members funded	Maximum	Number of Faculty Members included	Comments or clarifications
Salaries				
Professional or other allowances				
NSERC research grant				
Other research funding				
Other financial support				

4.2 Non-regular faculty.

How many courses are taught annually by non-regular faculty? How are non-regular faculty hired?

Answer

4.3 Workload

How is the administrative load distributed among the faculty? How is the teaching load distributed among the faculty? How are teaching assignments made?

Answer

4.4 Quality indicators

Please provide any additional data you have, not covered above, demonstrating the high or continually-improving quality of faculty members. This could include a high proportion of full-time faculty, high NSERC Discovery Grant renewal rates, recent or planned hiring to renew faculty, good gender distribution, surveys showing high job satisfaction among faculty, and good student evaluations.

Answer

5 Students

5.1 Enrollment and graduates in each program

Please complete the following tables for each program for which you are seeking accreditation, using one row per program and including the name of the program in the first column:

Programs to be considered	
Official Program Name	Program Code*

*"Program Code" should be an acronym or short name you will use to identify the entries in subsequent tables. This may be an acronym or similar code. It may be commonly used in your institution or invented for the purposes of this questionnaire.

Current Enrollment				
Program Code	Year 1	Year 2	Year 3	Year 4

Numbers of graduates in each of the last five years					
Program Code	5 years ago	4 years ago	3 years ago	2 years ago	most recently

5.2 Admission requirements, promotion requirements and passing averages

What are the requirements to enter into or continue in the programs? Please include explicit references to the University calendar or other submitted materials.

Answer:

If you have articulation agreements in place with non-university educational institutions, describe what policies and processes there are in place to assure the equivalency of courses that are recognized under the articulation agreements.

Answer

5.3 Student counseling and advising

How are students advised about course and career selection?

Answer:

5.4 Quality indicators

Approximately what percentage of graduates continue in post-graduate studies?

Answer:

How many graduating students received scholarships for post-graduate studies?

Answer:

Provide any other data you have indicating the high and/or continuously improving quality of students. This could include prizes awarded, high levels of job placement, feedback from employers, low attrition rates, high admission averages and high graduation averages.

Answer:

5.5 Graduate attributes:

Central to the outcomes-based accreditation process is demonstrating that all students have met certain learning and competency objectives by the time of graduation. CSAC has developed a set of graduate attributes, which it is suggested you adopt[2]. However, you may provide an alternative set of graduate attributes by simply editing the following section. Any edits must be underlined. One example of a situation where you may need to adopt different graduate attributes would be if

[2] The CSAC graduate attributes are designed to be consistent with those of the Seoul Accord, the international agreement governing mutual recognition of accreditation in computing. Accreditation agencies for other disciplines also use very similar graduate attributes.

your programs are also being reviewed by another agency that has specific requirements. A second example would be if your program focuses on a specialized area of computer science, and you wish to refine the attributes to take this into account.

A graduate of a computer science, software engineering or interdisciplinary program must be able to:

GA1. Demonstrate Knowledge: Competently apply knowledge in a) software engineering, b) algorithms and data structures, c) systems software, d) computer elements and architectures, e) theoretical foundations of computing, f) discrete mathematics and g) probability and statistics.

GA2. Analyse and Solve Problems: Use appropriate knowledge and skills, including background research and experimentation, to identify, investigate, abstract, conceptualize, analyse, and solve complex computing problems, in order to reach substantiated conclusions.

GA3. Design Software and Systems: Design and evaluate solutions for complex open-ended computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, as well as economic, cultural, societal, and environmental considerations

GA4. Use Appropriate Resources: Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of their strengths and limitations.

GA5. Work Individually and in a Team: Function effectively as an individual and as a member or leader in diverse teams and in multi-disciplinary settings

GA6. Communicate Effectively. Communicate with the computing community and with society at large about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions

GA7. Act Professionally. Act appropriately with respect to ethical, societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and with regard to the consequential responsibilities relevant to professional computing practice.

GA8. Be Prepared for Life-Long Learning: Learn new tools, computer languages, technologies, techniques, standards and practices, as well as be able to identify and address their own educational needs in a changing world in ways sufficient to maintain their competence and to allow them to contribute to the advancement of knowledge.

GA9. Demonstrate Breadth of Knowledge. Possess knowledge in areas other than computer science and mathematics so as to be able to communicate effectively with professionals in those fields.

If you have adopted a different set of graduate attributes than the CSAC defaults, please describe the reasons for doing so, and a mapping to the CSAC default attributes, justifying any omissions.

Answer:

6 Curriculum

In judging curricula for accreditation, the main objective is to ensure that, taken together, the material taught leads students to have met the graduate attributes by the time they graduate. Ensuring this is the case requires courses with breadth and depth, and a variety of teaching strategies.

For each of the questions in this section, when you are asked to identify a course, it would be very helpful to give the course code *plus* a very short abbreviation of the title of the course. For example, if CS202 is your data structures and algorithms course, you might right CS202-DSA. The added abbreviation will greatly speed comprehension by the evaluators.

Please provide information for each program, as taken by students who will be graduating *within two academic semesters*. If the lists of required courses have officially changed for students currently enrolled (but who will be graduating at a later date) then please explain the rationale for the changes and add rows to the tables with labels such as 'NEW CS (for students starting in 2014)'. If the code for a course has changed, but the content remains essentially the same, then please indicate this as oldCode [newCode]. If a requirement can be met by the student choosing from a small list or category of courses, then please list the codes of courses the students may choose from, separating each by 'or'. Finally, if there are differences in what appears below and the official calendar(s), please explain the discrepancies.

6.1 Contribution to the graduate attributes achieved in each course

For each computer science or software engineering course that is *required in any of the programs or options*, please complete the following table. Please replicate the table for each course. For each graduate attribute please give, where possible, quantitative indicators showing how that course contributes to that attribute. If the tables provide insufficient space, you may attach additional information as an appendix to this questionnaire. If you do that, please provide clear pointers to the supplementary material.

If you have given a different set of graduate attributes in Section 5.5, then please edit the rows here accordingly.

In some situations, you may want to group 2 or 3 courses together and create a table for the group. That might be the case, for example, if the calendar says student must choose "one of X, Y or Z". If X, Y and Z are sufficiently similar such that you can give the same answers regarding the graduate attributes, then grouping these three courses would simplify analysis.

Concise information is encouraged in the 'Details' column. One paragraph per cell will often suffice.

Per-course information requirements	Details
Course Code(s):	
Course Name:	
Program Codes:	*(In this cell, indicate 'all' if this course or group is required in all programs; otherwise list the programs in which it is required, or indicate "all except …")*
Names of professors who have most recently taught the course	
GA1 Demonstrate Knowledge.	*(Simply paste the course description here. Accreditation team members will verify that this material is indeed taught by looking at the course notes, exams, textbooks, etc.)*
GA2. Analyse and Solve Problems:	*(In the cell, briefly describe in this cell the analytic and problem-solving techniques learned or practiced, and how the assessment process ensures that students will in fact become proficient in these techniques. During the visit, accreditation team members will look for evidence to corroborate what you say in assignments, exams and student interviews.)*
GA3. Design Software and Systems:	*(In this cell, briefly describe in this cell the kinds of open-ended design experiences in the course, if any. Identify how students are taught to evaluate their designs, and how the assessment process guarantees that students will learn to be effective designers. Note that programming is considered a type of design. The visit team will corroborate what you say by looking at student wor and interviewing students.)*
GA4. Use Appropriate Resources:	*(In this cell, list the tools and practices learned, including software, platforms and hardware; briefly describe how the educational process ensures that students definitely learn to use these and that they are state-of-the-art. The visit team will look for evidence of various kinds to confirm what you say.)*
GA5. Work Individually and in a Team:	*(In this cell, describe any teamwork and how team performance is assessed, as well as how independent study is promoted.)*
GA6. Communicate Effectively.	*(In this cell, describe oral presentations, written work and other activities that teach communication)*
GA7. Act Professionally.	*(In this cell, describe elements of the course that promote professionalism and ethics, and how students obtain are assessed in this material)*
GA8. Be Prepared for Life-Long Learning	*(In this cell, describe ways in which students are given skills that will enable them to learn on their own later; this may include research tasks, significant individual reports, etc.)*
GA9. Demonstrate Breadth of Knowledge.	*(Describe any integration of non-computing learning into the course)*

6.2 Grid, summarizing how courses contribute to each of GA2 through GA9

Complete the following table, with each blank cell being replaced with:

*** if the course (or course group) contributes *greatly* to the GA in question;
** if the course or group contributes *substantially* to the GA; and
* if the course or group contributes in a *minor* way to the GA.
Leave the cell blank if the course does not contribute to the GA at all.

GA1 does not appear in the table since there are specific questions that address it. You may need to adjust the table if you have used a set of graduate attributes that differs from the CSAC defaults.

Please list courses or course groups that are required in all programs, then courses or course groups required only in certain programs

Course code(s)	GA2	GA3	GA4	GA5	GA6	GA7	GA8	GA9

6.3 Additional quality indicators

In addition to the per-course information given in Section 6.2, please summarize any other qualitative or quantitative assessment you have done which provides evidence that the graduate attributes have been met and that your curriculum is of high and/or continually improving quality. This may include surveys of students, surveys of employers, special tests given to students, interviews with students, etc.

Answer:

6.4 Expected minimum numbers of courses in various categories

Complete the following table to indicate how the *minimum* requirements in each program compare to CSAC's general expectations. Use one row for each program being considered. In each

cell please give the coeds and abbreviations for the required courses (or groups of courses) in your program(s) that fall in each category.

Please note that under outcomes-based accreditation, fulfilling graduate attributes takes precedence over strict counting of numbers of courses, so the guideline numbers given in rows two and three may not need to be fully adhered to in all cases. The material in this table will help assess GA1 (knowledge taught) and GA9 (breadth of knowledge).

Program Code	**CS/SE**	**Math**	**non-CS / non-Math**	**Unspecified**
Guidelines for CS and SE programs	*>=15*	*>=5*	*>=10*	*<=10*
Guidelines for Interdisciplinary programs	*>=10*	*>=3*[3]	*10 (at least 5 in each Other Discipline)*	*At least 3*

Additional-comments

[3] In case where mathematics is one of the Other Disciplines the total required courses is 10 (not 13). Courses in column three and four can be double counted.

6.5 Coverage of areas of computing

Complete the following table to indicate the codes and abbreviations for the *required* courses taken by students that significantly address each of the identified sub-areas within Computer Science, again using one row per program.

For general CS and SE programs, it is the general expectation that there will be material taught in all of the categories. However for interdisciplinary programs, it may be that only algorithms and data structures taught. As before, fulfilling the graduate attributes is key. The material in this table will help assess GA1 (knowledge taught) and GA9 (breadth, within computing).

Program Code	Software engineering	Algorithms and data structures	Software systems	Computer elements and architecture	Theoretical foundations

Additional-comments

6.6 Special types of knowledge

What knowledge and skills at an advanced level will students learn in each program? (Used to assess GA1)

Answer

How do the programs guarantee students are exposed to multiple programming languages and paradigms? (Used to assess GA1 and GA4)

Answer

How do the programs expose students to new areas of computing? (Used to assess GA1 and GA9)

Answer

6.7 Coverage of areas of SE, for SE programs

For programs to be considered under the **Software Engineering Accreditation** guidelines, please indicate the required courses in the program that cover the following areas of Software Engineering. This is to assess GA1.

Part 1

Program Code	Software requirements	Software design/ architecture	Software construction and maintenance	Software testing and QA

Part 2

Program Code	Software engineering management and process	Application areas	Human-computer interaction	Standards

Additional-comments

6.8 Coverage of areas of mathematics

Complete the following table to indicate the *required* courses taken by students that significantly address each of the identified sub-areas of mathematics, again using one row per program. This is to assess GA1.

It would generally be expected that in CS and SE programs, there is material taught in each of columns 2-7. For interdisciplinary programs Discrete Mathematics and Probability and Statistics would be considered normally essential, combined with some grounding in logic, Boolean algebra and the basics of graph theory.

Program Code	Discrete math	Calculus	Probability and statistics	Logic	Boolean algebra	Graph theory	Other math

Additional-comments

6.9 Breadth requirement

Complete the following table to indicate the *required* courses (or categories of courses) taken by students in each of the broad disciplines outside of computer science and mathematics/statistics, again using one row per program. This is to assess GA9.

Program Code	Science	Engineering	Business	Humanities	Social science

Note: For CS and SE programs, a minimum of ten courses is suggested in these areas, with a minimum of three courses in humanities or social science. For interdisciplinary programs a minimum of three courses are suggested in these areas.

Additional-comments

6.10 Required courses in Other Disciplines for interdisciplinary programs

For **Interdisciplinary** programs complete the following table to indicate the *required* courses (or categories of courses) taken by students in the Other Discipline(s) of the programs. Again using one row per program. This is to assess GA9.

Program Code	Courses

Note: A minimum of 10 courses is required in these areas, with at least 5 in each Other Discipline (two of which must be advanced courses)

Additional-comments

6.11 Additional questions regarding curriculum

How does the Department manage and review its curriculum?

Answer

Are there other innovative aspects of the programs that deserve special mention?

Answer

7 Resources

Answers to any of the following questions can be in the form of explicit reference to one or more pages in the submitted materials

7.1 Physical facilities

Briefly summarize the physical facilities (including offices, laboratories, and classrooms) available to meet program needs.

Answer

7.2 Computing resources

Briefly describe the computing resources (hardware and software) available to your undergraduate students. What policies and procedures are in place for maintaining and upgrading equipment?

Answer

7.3 Library

Please complete the following table to indicate the size of the library budget:

Library budget	**Computer Science**	**as % of Faculty**	**as % of University**	**Comments or clarifications**
for monographs	$	%	%	
for serials	$	%	%	

7.4 Additional information

Please attach your completed Canadian Association of Computer Science (CACS/AIC) survey for the most recently completed academic year, or complete the following tables:

FACULTY:

Faculty	(as of this past April 1)
Number of full-time faculty:	
Number of other faculty:	
Number of open positions:	
Number of additional projected positions:	
Starting salary you did/would offer a fresh Ph.D.:	

STAFFING:

Departmental Staff (as of this past April 1)	**University-funded positions**	**Funded by other sources**
Number of clerical staff:		
Number of administrative staff:		
Number of system software staff:		
Number of programming staff:		
Number of hardware staff:		

RESEARCH FUNDING:

Research Funding (last 12 months)	**Amount**
NSERC Research (operating) Grants:	$
NSERC Capital Grants:	$
NSERC Infrastructure Grants:	$
NSERC Strategic Grants:	$
Other Capital or Equipment Grants:	$
Other Operating Grants:	$
Total Value of Research Contracts:	$
Overhead Recovered by Department:	$

DEPARTMENTAL BUDGET:
Annual budget for last complete fiscal year, ending on (date):

	Budget	**Actual**
salaries (regular faculty):	$	$
salaries (sessionals):	$	$
salaries (clerical):	$	$
salaries (admin. support):	$	$
salaries (tech. support):	$	$
salaries (student T.A.s):	$	$
salaries (other):	$	$
equipment:	$	$
non-equipment capital:	$	$
computing:	$	$

NOTES:
Student T.A.s includes all marking, lab demonstration, etc. exclusive of sessional teaching.
Computing is all expenditures in support of computing separate from Departmental equipment and support (e.g. funds for purchase of computer time on a central Computing Services facility)

GRADUATE PROGRAM:
(most recent complete calendar year)

	Masters	**Doctoral**
Total full-time enrollment		
Total part-time enrollment		
Number of degrees awarded		
Average length of program (years)		
Minimum guaranteed funding		
Average funding per year		

UNDERGRADUATE PROGRAMS:
List *all* undergraduate programs offered through your Department, whether or not considered for accreditation:

Program Code (or full name if not being accredited	**Date program introduced**	**Most recent number of graduates**	**Total current enrollment**

8 Privacy Code Statement

The information supplied in the submitted in this questionnaire is for the confidential use of the Computer Science Accreditation Council and its authorized agents, and will not be disclosed without authorization of the institution concerned, except for summary data not identifiable to a specific institution.

The CIPS Accreditation Secretariat collects personal information through the accreditation questionnaire and during the site visit. This information is used solely by certain approved members of the CIPS national office staff, member volunteers who are related to the accreditation process, and external accreditors from the Seoul Accord, for the purpose of:
1) assessing a faculty's ability to meet the accreditation criteria;
2) assessing student compliance with the accreditation criteria;
2) performing an audit of Accreditation Councils ' procedures and processes.

The Canadian Information Processing Society is committed to the principles of the Canadian Standards Association's "Model Code for the Protection of Personal Information." A copy of the full "Accreditation Information Privacy" document can be obtained by contacting CIPS Executive Director at info@cips.ca or (905) 602 1370

9.0 Mailing Instructions and Contact Information

All information should be submitted to:

CIPS Accreditation Secretariat
5090 Explorer Drive, Suite 801
Mississauga, Ontario L4W 4T9

For questions and for electronic submission of the document, contact:

Gina van Dalen, Manager Professional Standards
accreditation@cips.ca
(905) 602-1370 ext. 329 or
(905) 486-1079