Computing and IT Accreditation Canada

Accreditation Criteria

Last Revised: March 2026

Table of Contents

Abstract 4

1. Introduction 5

2. Method of Evaluation 6

2.1 Overall Evaluation 6

2.2 Submission Guidelines 7

3. Glossary 8

4. Accreditation Process 10

5. Objectives and Outcomes 11

5.1 The CIPS Graduate Attributes 11

5.2 Quality Indicators and Outcomes 13

5.3 Quality Improvement/Enhancement Process 13

6. Faculty 14

7. Facilities and Resources 16

8. Administration, Planning, and Internal Process 17

9. Students 18

10. Industry Support 19

11. Curriculum 20

11.1 Evidence that Graduate Attributes Have Been Met (Quality Indicators) 21

12. University Curriculum Accreditation Specific *Criteria* 23

12.1 Overall Distribution of Courses 23

12.1.1 Expected Minimum Number of Courses for Computer Science and Software Engineering Programs 23

12.1.2 Expected Minimum Number of Courses for Interdisciplinary Programs 23

12.2 Breadth of Coverage in Computer Science Topics 24

12.3 Mathematics [Graduate Attribute 1f to 1g] 26

12.4 Breadth and Depth in Topics Outside Computing and Mathematics [Graduate Attribute 9]

..... 27

12.5 Non-Trivial Problem Solving in Teams [Graduate Attributes 2 and 5]..... 28

12.6 Written and Oral Communication Skills [Graduate Attribute 6]..... 29

12.7 Professionalism [Graduate Attribute 7] 29

12.8 Coordination of Program Schedules in Interdisciplinary Programs 29

13. College Curriculum Accreditation Specific *Criteria*..... 30

13.1 Technical Curriculum - Overall Distribution of Courses 30

13.2 Program Characteristics Typically Associated with Meeting Graduate Attribute..... 30

14. Business Technology Management Curriculum Accreditation Specific 34

14.1 Hierarchical Structure of the BTM 2.0 34

14.2 BTM Competency Standard Framework 34

14.3 BTM Program Characteristics Typically Associated with Meeting Graduate Attribute 35

15. Innovation and Research..... 37

Appendix A - Graduate Attributes (as per Seoul Accord)..... 38

Appendix B - CIPS Accreditation Process 42

B.1 CIPS Accreditation Process 42

B.2 Accreditation Decisions and Term..... 42

B.3 The Appeal Process..... 43

B.4 Changes During Period of Accreditation 45

B.5 Publication of Accreditation and Public Release Policy..... 45

B.6 Definition of Report Terms..... 46

Abstract

These guidelines are written to aid faculty and administrators involved in the accreditation of Computer Science, Software Engineering and Inter-Disciplinary, Information and Communications Technology (ICT), Business Technology Management programs within public and private universities, colleges, polytechnics, and institutes of technology. These guidelines are administered by the CIPS Accreditation Council and apply to programs leading to degrees, diplomas, applied degrees, certificates, micro-credentials who are typically one to four years of duration. They specify the objectives of accreditation, the various steps in the process, and the essential and highly desirable qualities of accreditable programs. Specific criteria are applicable depending upon the duration and intent of the programs undergoing accreditation.

For further information related to the Accreditation Council and Criteria for a specific program or set of programs, please contact the Canadian Information Processing Society (CIPS) Accreditation Secretariat.

The following sections specify the objectives of accreditation, the various steps in the process, and the essential and desirable qualities of accreditable programs. Questions and suggestions for improvements may be sent directly to the CIPS Accreditation Secretariat (accreditation@cips.ca).

1. Introduction

The Accreditation Council is an autonomous body established by CIPS.

The Council has as its objectives:

1. To formulate and maintain high educational standards for universities, colleges and institutes of technology offering computer science, software engineering and Inter-Disciplinary, information and communications technology, business technology management programs, and to assist those institutions in planning and carrying out educational programs.
2. To promote and advance all phases of computer education with the aim of promoting public welfare through the development of better educated computer practitioners and professionals.
3. To foster a cooperative approach to computer science and software engineering, and/or ICT or BTM education among industry, government, and educators both nationally and globally to meet the changing needs of society.

The purpose of accreditation is to recognize programs whose graduates will have received an outstanding undergraduate computer education – an education informed by state-of-the-art professional practice, sound underpinnings of information and computer technologies, and the needs and applications of industry. Accreditation can also be an important component in an Institution's quality monitoring and improvement program.

CIPS Accreditation Council accredits programs primarily in Canada but welcomes institutions from outside Canada wishing to undergo accreditation based on council criteria and standards. CIPS accreditation criteria incorporates principles of *outcomes-based* accreditation. This contrasts with an emphasis on educational *inputs,* such as number of courses taught, and lists of topics in the curriculum. The emphasis of these criteria is instead towards *outcomes*, i.e., *identifying and setting sound educational objectives and measuring the extent to which these objectives have been met*. These objectives and outcomes can be expressed at course or program levels.

More specifically, outcomes-based accreditation requires the setting of clear program *objectives* (i.e. the intended purpose of the program) and program *outcomes* which describe what students should know and be capable of doing upon graduation from the program. Program outcomes can also be expressed as *graduate attributes*, defined as 'a set of individually assessable outcomes that are indicative of a graduate's potential competency'. [1] Institutions will typically set their own specific program objectives, outcomes, and graduate attributes but CIPS-accredited programs are expected to substantially meet one of three ranges of graduate attributes as defined by the Seoul Accord, which has established a set of internationally recognized expectations for students graduating from various types of computer programs.

CIPS accreditation is designed primarily for computer programs as offered through universities, colleges, institutes of technology and other institutes of higher learning. Specific criteria are provided corresponding to the graduate attributes as set by the Seoul Accord[1]. As such, CIPS accreditation is intended to be applied to a wide range of program types and durations, providing institutions with the flexibility to design program outcomes and graduate attributes to meet the needs of their institutional mandates, students, and target industries.

---

[1] "The eight signatories of the Seoul Accord have joined together for the primary purpose of contributing to the improvement of computing education worldwide through the mutual recognition of accredited academic computing programs that prepare graduates for professional practice. By establishing desired attributes for graduates of computing programs that prepare graduates for professional practice..." (Seoul Accord is now 18 signatories)

URL: http://www.seoulaccord.org/about.php

2. Method of Evaluation

2.1 Overall Evaluation

Programs submitted for accreditation will be evaluated based on data submitted by the institution in the form of a self-study report and other supporting documentation, together with the report of an on-site visit by a qualified team representing the Council.

The self-study report should follow a structured outline to be described in Sections 5 though 12 and involves answering a series of questions and completion of tables. During the process of creating the report, the institution should demonstrate to itself and to the Council that it can meet the accreditation criteria or, if not, it should demonstrate that it is aware of the shortcomings and has a concrete plan to rectify them. In particular, the report should demonstrate how all aspects of the program, including students, faculty, resources and curriculum together enable the achievement of a set of defined program objectives, discussed in Section 5. The self-study report will be used as primary input for the analysis of the program by the on-site visiting team.

The purpose of the site visit is three-fold:

First, the site visit should assess factors beyond those described in the questionnaire. The overall educational environment, the morale and calibre of the staff and the student body, and the approach taken to the work performed are examples of intangible qualitative factors that are not always apparent in a written statement.

Second, the visiting team can observe firsthand the strengths, unique characteristics and areas of potential improvements related to the program.

Third, the team will assess and validate the material in the self-study including:

1. Organization structure and administration of the institution.
2. Education programs offered, and credentials conferred by the institution overall.
3. The basis of and requirements for admission of students both in general and to the program(s) undergoing accreditation.
4. Number of students enrolled in the educational programs undergoing accreditation.
5. Teaching staff and teaching loads.
6. Resources:
    a. physical: classrooms, laboratories, equipment, and offices,
    b. reference materials: electronic resources and/or digital libraries,
    c. additional facilities, where they exist and are relevant (e.g., entrepreneurship labs, innovation labs)
7. Curricular content of the program(s).
8. Program delivery and outcomes, including sampling of transcripts, examinations, projects, assignments, etc.
9. Commitment to and support for research.
10. Innovative and special features of the program.
11. Institutional policies and supports.

2.2 Submission Guidelines

Please submit the following documents in an accessible platform, in the identified folder structure as part of your electronic submission package. (e.g., Cloud, Dropbox, SharePoint, LMS)

Folder Structure is as follows:

Program Name (root folder), which will contain the following:

- Self-study report in Word format
- Final visit agenda in Word format
- The following sub-folders:
    - **Curriculum** folder will contain the following:
        - All Course Detailed Syllabus Outlines
        - Curriculum development policies and procedures documents
    - **Facilities** folder will contain the following:
        - Lab report(s) listing hardware and software configurations:
            - Provide a detailed lab room listing with computer hardware specifications.
            - A listing of software used in the credential.
            - Provide any student computer specification documentation, such as Bring Your Own Device or any other required resources.
        - Campus Map or Program Area Map (if available)
    - **Faculty** folder will contain the following:
        - Faculty CVs
        - Faculty Load policy or collective agreement article
    - **Institutional** folder will contain the following:
        - Organization Chart, which includes a diagram of the administration of the school emphasizing the reporting structure of the department offering the program to be accredited.
        - Institutional Calendar, which also includes copies of all external publications which describe the school and the program.
        - Program Handbook (if one exists)
        - Program Entrance Requirements
        - Program KPI Document
    - **Industry** folder will contain the following:
        - Program Advisory Committee Terms of Reference
        - Previous PAC meeting minutes
        - Current membership names and organizations
    - **Other** folder will contain materials additional to the ones listed above.

Please include any other documentation you believe relevant to the program accreditation process within the appropriate folders identified.

3. Glossary

For the purpose of a CIPS accreditation, the following definitions apply:

Graduate Attributes: A high level 'set of individually assessable outcomes that are indicative of a graduate's potential competency' [https://www.seoulaccord.org/document.php?id=79]. The Graduate Attributes used by CIPS Accreditation Council are those established by the Seoul Accord (https://www.seoulaccord.org).

Three ranges of attributes are defined:

Computing Professional

Computing Technologist

Computing Technician

Degree program*: A University program typically 4 years in duration leading to a baccalaureate degree. The expected graduate attributes will be in the **Computing Professional** range and are recognized by the Seoul Accord.

Diploma Program*: A College or Polytechnic program typically 2 or 3 years in duration leading to a Diploma, Diploma of Technology, Applied Degree, or similar credential. The expected graduate attributes will be in the **Computing Technologist** range and are not recognized by the Seoul Accord at this time.

Certificate Program*: A program, offered by a post-secondary institution, typically 1 year in duration leading to a Certificate. The expected graduate attributes will be in the **Computing Technician** range and are not recognized by the Seoul Accord at this time.

* These typically refer to an initial credential in the ICT area. The expected graduate attributes for programs which require advanced prerequisites, especially in the ICT field, may differ depending upon the nature of the program. E.g., the expected graduate attribute for a one-year certificate program where the prerequisite is a degree or diploma in an ICT area might be in the Computing Technologist or Computing Professional range. For further information which graduate attribute range might be appropriate for a specific program, please contact the CIPS Accreditation Secretariat.

Interdisciplinary program: A program in which computer science is taught alongside one or more Other Disciplines to form an interdisciplinary degree program. Such programs may be 'joint programs', tightly integrating sets of courses from different disciplines. They may also involve a set of 'building blocks' (often called 'majors') that can be used to create a variety of interdisciplinary combinations that include computer science. Regardless of how programs are structured, it is essential that computer science is the 'equal' of the other discipline or disciplines; in other words, programs in which computer science is only a 'minor' are not eligible. In this document, unless otherwise specified, all general criteria, as well as computer-science-specific criteria, also apply to interdisciplinary programs.

Business Technology Management program: A program in which computer science or information systems is a minor component of the overall program where business is taught as a major with the program. These programs are interdisciplinary with computing as a minor component and are not recognized by the Seoul Accord.

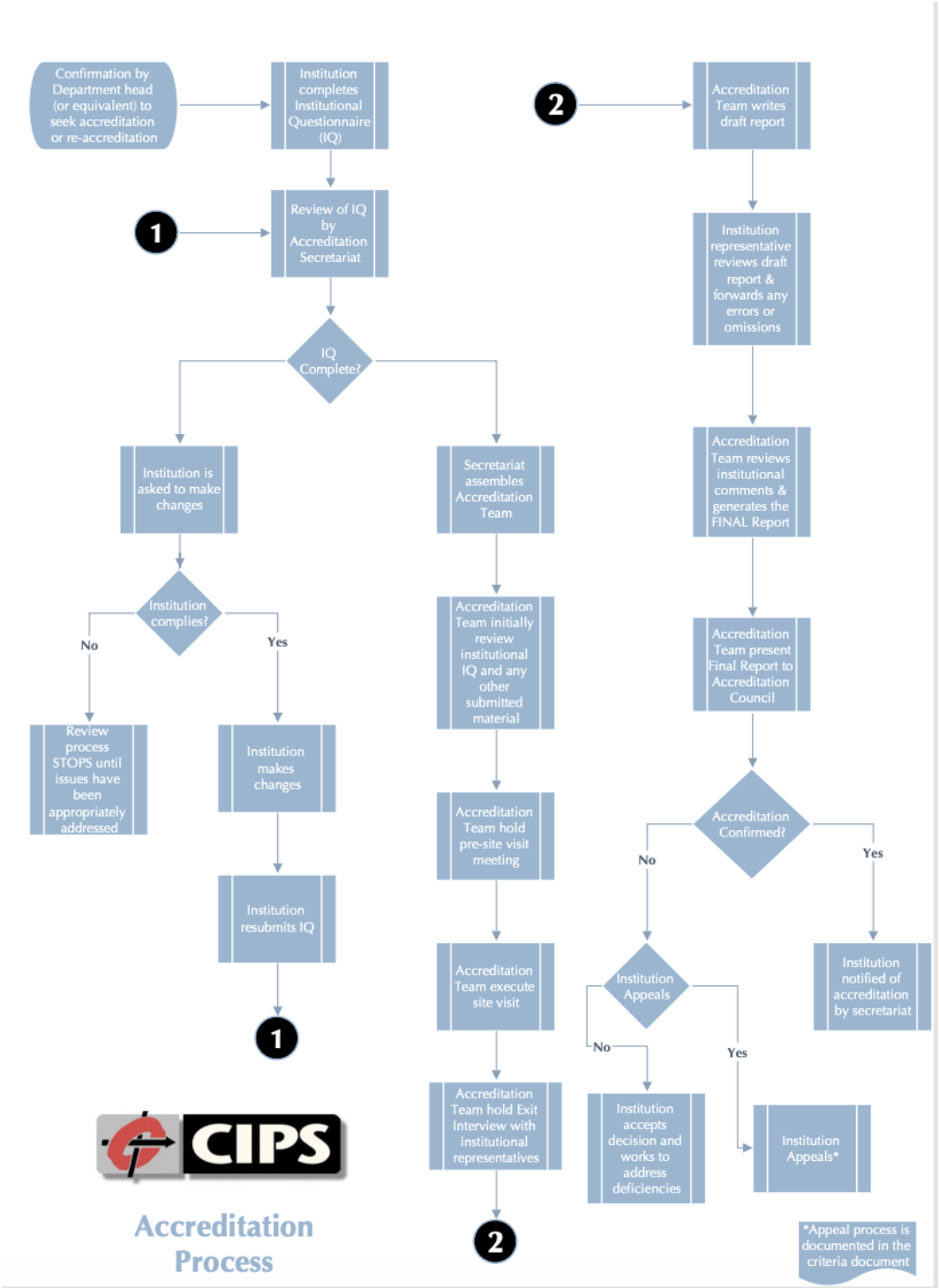
Objective: Planned goals or intent of a program or educational unit (e.g., course). Generally expressed from the perspective of the teacher or faculty. Occasionally the term is used interchangeably with Outcome.

Outcome: Measurable evidence that an objective has been met. Also refers to the expectations of the students' achievements or accomplishments after the educational activity. Generally expressed from the learners' perspective. Occasionally used interchangeably with Objective.

Quality Indicator: Qualitative or quantitative data used to help assess whether an objective has been met.

Rubric: A document describing how an exam, assignment or other student activity should be evaluated, specifically mapping to the learning objectives that should be assessed.

4. Accreditation Process



NOTE: Please refer to appendix B for an accreditation process elaboration.

5. Objectives and Outcomes

Each program must have a set of program objectives and graduate attributes describing what students should know and be capable of doing following graduation.

Institutions and individual programs have flexibility in setting their own program objectives, outcomes and graduate attributes but they must also substantially meet the appropriate graduate attribute range set by the Seoul Accord. Institutions may: adopt these verbatim; add to them; or reword some of them to meet local conditions or conform to internal requirements. In some situations, a documented mapping of the institution's program objectives to the Seoul Accord attributes may be required to demonstrate that the appropriate attribute range has been achieved. For example, a typical situation in which it would be reasonable for an institution to add or refine a graduate attribute would be when the program focuses on preparing graduates to work in a particular domain (e.g., bioinformatics/health informatics) or a particular sub-specialty of computer science (e.g., artificial intelligence).

5.1 The CIPS Graduate Attributes

The table in Appendix A provides profiles of graduates of three types (professional, technologist, technician) of postsecondary educational computing programs, as defined by the Seoul Accord.

The following is the default set of program-level objectives for all accredited programs. Since these refer to what students should know and be able to do following graduation, they are referred to as Graduate Attributes.

A graduate of university, college, polytechnic or software engineering program, at the appropriate credential expectation level, must be able to:

1. 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.

2. 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, to reach substantiated conclusions.

3. 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

4. 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.

5. 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

6. 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

7. 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.

8. 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.

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

Please note: The above attributes, for the purpose of university programs, are derived from the following two sources and are designed to be as consistent as possible with these sources: The Graduate Attributes of the Seoul Accord, which is the international accord affording portability of accreditation among various countries, and the Graduate Attributes of the Canadian Engineering Accreditation Board. The latter alignment allows departments that wish to seek accreditation of their software engineering programs by both CIPS and the Engineers Canada Canadian Engineering Accreditation Board (CEAB) to reuse a substantial portion of CEAB self-study reports for CIPS accreditation purposes.

5.2 Quality Indicators and Outcomes

Evidence should be provided that the graduate attributes defined for each program have been fulfilled. In other words, there must be evidence that what students know and can-do following graduation correspond with the defined program-level objectives. This is achieved by *quality indicators*. These are qualitative and quantitative data gathered by the institution.

The Accreditation will review quality indicators in each of the following areas/sections: Faculty, Students, Curriculum, and Resources. Suggestions for quality indicators are provided in each corresponding section. The self-study and accreditation process largely involve studying and verifying the quality indicators to ensure that the outcomes correspond to the defined objectives.

5.3 Quality Improvement/Enhancement Process

Evidence should be provided that the department has a quality enhancement process to improve the programs under accreditation through regular review and upgrade of program objectives and graduate attributes.

6. Faculty

The capability of any program to provide high-quality education or training is ultimately dependent upon the background and dedication of the faculty members. A competent and dedicated faculty can provide a good program despite deficiencies in other areas such as physical resource availability. However, excellent resources and facilities cannot compensate for insufficient or poor-quality staff. It is therefore important that a school seeking accreditation for a program has allocated the resources necessary to provide enough high-quality faculty and is committed to maintaining the allocations necessary for the continuation of high-quality program delivery.

A CIPS-accredited program will possess a strong, competent faculty with the necessary size and capabilities to deliver the program, establish the appropriate rapport with students and maintain overall program quality.

Factors related to faculty size include the enrolment, objectives of the program(s) being accredited, subject areas required by the curriculum, assigned teaching loads, funded (or applied) research, industry involvement, extension and continuing education activities, and participation in professional and technical societies.

To function effectively as teachers, faculty members require enough time to acquire new knowledge and skills, pursue industrial interaction, explore instructional innovation, as well as consult and engage in other professional development activities. Teaching loads should reflect these requirements.

Suitable quality indicators regarding faculty for the self-study and accreditation report include:

- The proportion of full-time faculty to part-time and sessional.
- The teaching load (number of courses and preps taught per year)
- Distribution of faculty expertise and experience over areas required by the program.
- Knowledge and skills of faculty in areas corresponding to the graduate attributes, including areas related to ethics and professionalism.
- Sustained levels of research and research grants. (University Accreditations only)

Faculty members should be aware of the technical skills and business knowledge found in the local communities to be served by the program as well as those common to the business community in general. Factors to be considered in this assessment include academic background, pedagogical credentials, and work experience [*Defined as professional level experience including working in nontrivial, complex project planning, analysis, development, and implementation areas, including all phases of the system development cycle.*], and the related business experience of the instructional and administrative staff. There should be a broad range of knowledge and experience enough to provide expertise in all areas of the program's curriculum.

It is the expectation of accredited programs that all faculty possess formal qualifications in Computer Science, IS, IT, Computer Engineering, Software Engineering, or a closely related discipline. At the college level, the faculty profile should also reflect a balance of credential levels, including undergraduate and graduate degrees. At the university level, the faculty profile will more closely reflect graduate-level training. CIPS National Guidelines and Expectations for Faculty Teaching undergraduate and graduate courses are:

1. Faculty members have academic credentials and work experience equivalent to or better than the credential and work experience expectations of the students upon graduation.

2. A significant percentage (typically 30% or more) hold graduate degrees in Computing or a related field.
3. IT faculty members have at least 2 years of full-time IT related industry experience (other than teaching)
4. A significant percentage (typically 50% or more) have a minimum of 5 years of senior full-time Computer or IT related experience.
5. Alternative considerations:
    a. Faculty who possesses their CIPS I.S.P. or ITCP designation.
    b. Mature faculty who does not have formal credentials but learned "on-the-job" and usually have a significant number (10 or more) of years of full-time IT related industry experience.
    c. Mature faculty who does not have formal credentials, but who do have an appropriate combination of vendor specific credentials and significant (typically 5 years or more) of senior full-time IT related experience.

To evaluate the quality of the faculty, the visiting team will examine the data presenting the quality indicators in the self-study report, as well as the CVs of the faculty members and any collective agreement. It will also meet with groups of faculty members. The team will gather further insights from discussions and interviews with staff, students, and administrators.

In addition to faculty teaching directly into the program, the visiting team may also interview faculty from other specialty areas related to the program such as mathematics, business, communications, etc. The visiting team will not necessarily need to meet faculty members, nor examine a complete set of CVs, but will examine institutional policies and procedures to satisfy themselves that faculty members in these disciplines are comparable to faculty members within the faculty of the program seeking accreditation. For example, the visiting team will need to ensure that, in general, the faculty members in mathematics and the cognate disciplines have appropriate backgrounds and teaching loads.

7. Facilities and Resources

All the disciplines in an accredited program must have buildings, offices, laboratories, equipment, support staff, and fiscal resources that are appropriate for the characteristics of the program that is being undertaken. The accreditation team will review and tour these facilities as part of the on-site visit.

Software/Hardware provided and used within the program should be representative of commonly encountered business software/hardware/infrastructure in use in the community and should provide for the full spectrum of applications. There should be specialized systems, generalized business application software and technical software, as well as the hardware and infrastructure required, available to the students during the learning process.

There must be adequate access to electronic and other reference resources, such as the ACM and IEEE and other ICT digital libraries. The collections must be maintained and refreshed to remain current, and there must be a breadth of materials included. Electronic networking sufficient to provide students and faculty access to external resources is also important.

Suitable quality indicators for the self-assessment and accreditation report include the following:

- Computers and software in labs or Bring Your Own Device specifications.

To evaluate the quality of resources, the visiting team will inspect them while touring the facilities, and will interview students, staff, and faculty.

Access to technical information and documentation must be readily available. Alternative access modalities (e.g., online, databases, etc.) should be available for use to enrich the program delivery.

8. Administration, Planning, and Internal Process

A capable administration must be in place that understands the special needs of a technical program. Formally documented policies and procedures should be in place and well communicated to faculty, students, and other stakeholders as appropriate. Planning (long and short term, operational and strategic) must take place at all levels, and a monitoring and feedback process must be present.

The process for ongoing curriculum renewal must include the gathering of data from a variety of sources to inform the curriculum renewal process, including feedback from graduates and industry.

9. Students

A CIPS-accredited program is typically characterized by its enthusiastic students. Student selection and retention standards will be appropriate to the program. Well-established protocols will be in place for students transferring from other institutions, programs, or branch campuses.

A student advisory system for both academic and personal support is an important component in any educational program. The advisory system should embrace course selection, graduation advice and resolution of problems of a personal nature. Career guidance and employment support both pre- and post-graduation is a valuable resource.

Faculty members and career counsellors will be familiar with professionalism issues, ethical codes of conduct and related matters within the ICT industry. Exposure to specific professional standards and ethical expectations in the Computer and IT industry (e.g., CIPS Ethics Code) is expected.

Suitable quality indicators regarding students for the self-study and accreditation report include:

- Feedback from employers, assessed through questionnaires or surveys that ask employers the extent to which students they hire possess the graduate attributes defined for the program.
- Students' satisfaction with their program and progress as assessed through questionnaires and interviews.
- Attrition rates
- Admission rates
- Graduation rates

To assess the quality of students, the visiting team will interview groups of students and alumni, study transcripts and samples of student work, as well as analyse the data presented in the self-study report. Where possible, the team will also speak to employers, such as members of the department's program advisory committee.

10. Industry Support

An important feature in the success of an information systems program is the interaction between the school and the local business community. A good interaction means support from industry and advice for the instructional staff. This interaction can take many forms, but the most common are:

- An advisory committee made up from a cross-section of local business representatives; such a committee can keep the school aware of current trends in hardware, software, and skill requirements, as well as to inform the local community of the current activities of the academic world.
- Visits to local businesses where the faculty and/or students are provided with exposure to real life situations not only from the information systems point of view but also involving the various functional areas of business such as marketing, manufacturing, finance, etc.
- Guest lecturers from a variety of business and industry backgrounds can also be used to provide faculty and students with a valuable insight into the realities of the working world,
- Co-op and work-term projects which facilitate student involvement in a program-related activity in a real business setting are highly desirable. Such activities are not only beneficial to the students, but also to all groups involved. Industry representatives can assess the skill level of potential employees and to acquire a better knowledge of the skills embodied in the graduates.

11. Curriculum Overall

The curriculum must serve the needs of the students, employers and the community. Accredited programs should allow all these stakeholders the opportunity to provide an influence on the curriculum and to ensure that graduates are qualified for information systems related employment. It should foster the development of graduates with a diverse set of skills to meet immediate and long-term needs. It is particularly important that graduates be prepared for 'life-long learning' to maintain currency in this rapidly changing and evolving field.

The name of a program seeking accreditation must be descriptive of the program's content and be stated the same way on the graduate's transcript and in the institution's literature. Programs outside Canada where English or French is not the native language, must provide the program's name both in English or French and in the native language(s). An institution may not use the same program name to identify both an CIPS accredited program and a program that is not CIPS accredited.

This section refers to units of instruction comprising the curriculum for a program. It is recognized that institutions can design their instructional units in differing ways (traditional courses, learning modules, self-paced units, etc.). For this section, typically an instructional unit consists of a course outline or contract, learning objectives, teaching materials (notes, textbooks, etc.) and teaching/learning activities (labs, lectures, assignments, etc.) and evaluations (tests, quizzes, exams). Different institutions define courses and programs in different ways. For convenience, the term course is often used in this section to refer to instructional units given that most post-secondary institutions use this terminology.

For purposes of clarity in this document, a course is a single semester course, of roughly 12 weeks in length consisting of approximately 36 lecture hours. Thus, an undergraduate program would consist of roughly ten (10) courses per academic year. College and polytechnic programs may have additional lecture hours within courses, reflecting their unique learning environment. It is recognized that institutions assign titles to courses in differing ways, and it is also recognized that courses are offered by different academic units, potentially within different institutions. In determining the extent to which courses meet the requirement specified herein, it is the contents of the courses, as evinced by the course outlines, learning objectives and teaching materials, which are of significance, and not the course titles or the academic unit offering the courses.

In addition, although a course is defined approximately as 36 hours of lectures, it is understood that pedagogical methods such as Problem Based Learning, studios, and online learning may to a greater or lesser extent be used instead of lectures.

An introductory course has no post-secondary-level course as a required prerequisite. An intermediate or advanced course has at least one course as a required prerequisite. In many post-secondary institution introductory courses are referred to as 'first year courses', while an intermediate or advanced course might be referred to as an 'upper year course'.

The learning objectives of the instructional unit will contribute meaningfully to the objectives of the program and its graduate attributes. The accreditation team will review the curriculum to ensure that this is the case - both as a plan (i.e., the mapping of instructional unit level objectives to the appropriate graduate attributes) and in its execution (i.e., the actual outcomes of the course to the attributes in terms of the capability of the student).

While local industry or community needs will impact the content of the program, accredited programs also address preparing graduates to work as Computer and/or IT professionals nationally and even globally.

It is expected that the school will have several quality control mechanisms in place which will be used on an on-going basis to:

- Conduct environmental scans related to both local and national industry trends.
- Review program objectives and align to industry needs and trends.
- Evaluate the appropriateness of course content.
- Evaluate the appropriateness of grading and assessment procedures.
- Monitor the quality of faculty and teaching.

Note for Quebec Universities Accreditations: In Canadian provinces other than Quebec, a student typically obtains their degree after 16 years of study, including grades 1 through 12, plus four years of university. In Quebec, a student typically obtains their degree after 16 years: 11 years of primary and secondary school, 2 (general curriculum, pre-university) or 3 (technical curriculum leading to university) years of CEGEP, and 3 or 2 years of university. The criteria below (explained in terms of numbers of courses) are specified assuming a university program of four (4) years (40 courses). To satisfy the criteria, a program from Quebec may therefore include up to: 10 CEGEP courses for students who have obtained a (general curriculum) pre-university CEGEP diploma; 15 CEGEP courses for students who have obtained a technical CEGEP diploma (leading to a university program).

11.1 Evidence that Graduate Attributes Have Been Met (Quality Indicators)

Central to outcomes-based self-study and accreditation is demonstrating that the appropriate range of graduate attributes have been met for the program undergoing accreditation. For each of the remaining curriculum subsections measurable evidence (quality indicators) should be provided that each student has fulfilled each of the graduate attributes. Such evidence can include mappings from course-level objectives to graduate attributes, rubrics for assignments and tests indicating which graduate attributes are being assessed, and other quality indicators.

Learning objectives and outcomes should be presented for core courses, as well as topics in other related disciplines where appropriate. Expressed from the learners' perspective, learning outcomes should map to the appropriate graduate attributes and describe what students will have achieved by the end of each course. Each outcome will be typically expressed in sentences with an active verb. Verbs implying performance capabilities, such as ‘calculate’, ‘design’, ‘evaluate’, ‘apply’, ‘solve’, ‘create’, ‘build’, ‘determine’, ‘develop’, ‘assess’, ‘use’, ‘lead’ and ‘present’ are preferable to verbs implying more passive learning, such as ‘know’ and ‘understand’, although the latter would be appropriate for some types of knowledge.

In the same manner that the program-level objectives, outcomes and graduate attributes should be approved and reviewed through an appropriate quality review process, the learning objectives and outcomes at the instructional unit level should be approved by the department and reviewed periodically. Part of this review will be to ensure that the learning objectives, outcomes and graduate attributes remain in alignment. Evidence should be presented that the objectives are being accomplished, applied, within each course. Such evidence might include that they are present in course syllabi distributed to students, and that textbooks, lecture notes, assignments and exams match the objectives. The use of *rubrics* or similar documentation for learning activities is recommended. Rubrics describe what is to be expected of students in each learning

activity and should indicate which learning outcomes students would be demonstrating by successfully completing the activity.

The visiting team will review samples of student work including examinations, tests, projects and assignments to this end. In addition, the team will cross-reference information gathered from interviews with faculty, students and others.

Taken together, the outcomes of the instructional unit taken regardless of the path a student chooses for graduation, should satisfy the graduate attributes. To demonstrate this, the self-study report contains a table for each required instructional unit, course (or course group) mapping how the unit contributes to each graduate attribute.

12. University Curriculum Accreditation Specific *Criteria*

12.1 Overall Distribution of Courses

To meet the graduate attributes, it is expected that an accredited program will include the equivalent of forty courses of study (not necessarily all at the same institution) and lead to a Baccalaureate degree or equivalent. Furthermore, the program must ensure that all students must meet the graduate attributes to be eligible for the degree. The next two subsections describe the distribution of courses that would generally be expected for different types of programs. Prior to the era of outcomes-based accreditation, the numbers of courses were considered minimums. As of these current criteria, priority is given to demonstration that the graduate attributes described in Section 5.1 have been met. The numbers of courses listed can now be seen as guidelines only and may be dropped in a future version of these criteria.

12.1.1 Expected Minimum Number of Courses for Computer Science and Software Engineering Programs

A. Fifteen (15) courses in computer science, software engineering or computer engineering (Graduate Attribute 1a through 1e; see Sections 12.3.1 to 12.3.3 for details)
B. Five (5) courses in mathematics or statistics (Graduate Attribute 1f and 1g; see Section 12.3), and
C. Ten (10) courses in subjects other than computer science, software engineering, computer engineering, mathematics and statistics (Graduate Attribute 9; see Section 12.4).

The set of courses in each area should exhibit some breadth and some of the courses in each area should be at an intermediate or advanced level. This thirty-course requirement leaves each student with the equivalent of one year of study to satisfy additional institutional requirements and to accomplish personal objectives.

12.1.2 Expected Minimum Number of Courses for Interdisciplinary Programs

Ten (10) courses in computer science, computer engineering or software engineering (Graduate Attribute 1a through 1e; see Sections 12.3 for details).

A. Ten (10) courses from the Other Discipline or disciplines of the interdisciplinary program, with no discipline having fewer than five (5) courses (See section 12.5.2).
B. Three (3) courses in mathematics (Graduate Attribute 1f and 1g; see Section 12.3). In the case where mathematics is one of the other disciplines, the total number of math courses should be 10, not 13; in other words, courses in B and C can be double counted.
C. Three (3) courses in non-technical disciplines, defined as disciplines other than computer science, natural sciences, mathematics and engineering (Graduate Attribute 9; see Section 12.4.3 for details). If one of the Other Disciplines is non-technical, then this requirement is already met; in other words, courses in B and D can be double counted.

The above requirements leave each institution and/or student with between 14 and 20 courses to satisfy additional program requirements (which may include more CS, more mathematics, and/or more of the Other Disciplines) and to accomplish personal objectives. However, to ensure that there is a balance between the Other Disciplines and computer science, the total number of courses required for the Other Disciplines combined should be similar to the number of computer science courses required.

12.2 Breadth of Coverage in Computer Science Topics

Similarly to the program, the computer science component of the program must offer a breadth of exposure to different topics, along with a depth of understanding.

12.2.1 Key Areas of Computer Science

All students in all accredited programs, including computer science, software engineering and interdisciplinary programs should be required to take courses which satisfy graduate attributes 1 (a), (b), (c) (d) and (e) described in Section 5.1. The following paragraphs expand on the range of knowledge and skills that students might acquire in the respective areas. Introductory courses in computer programming and computer usage would not be expected to meet these requirements. See sections11.3.2 to 11.3.4 regarding specifics of how to apply this section to computer science, software engineering and interdisciplinary programs respectively.

This classification of five subtopics is not intended to limit the scope of computing nor the areas to be included or excluded at institutions, but rather it is illustrative of a range of topics that an accredited program would be expected to provide.

Graduate Attribute 1a: Software engineering, including requirements specification, software design and software architecture, software development, software testing, software maintenance, and other topics related to software process (For Software Engineering programs this list is expanded in Section 11.3.3).

Graduate Attribute 1b: Algorithms and data structures, including data structures such as stacks, trees, lists, queues, etc.; abstract data types, established solutions to classical problems (e.g., sorting and searching), and analysis of algorithms.

Graduate Attribute 1c: Systems software, including operating systems concepts, virtual memory management, management of distributed, parallel, and concurrent processes, transaction processing, logging, security, and computer networking.

Graduate Attribute 1d. Computer elements and architectures, including computer organization, digital device and communications technology, logical and physical hardware design.

Graduate Attribute 1e. Theoretical foundations of computing, including models of computation, analysis of algorithms, fundamentals of program specification and verification, computational complexity, grammars and automata.

12.2.2 Key Areas of Computer Science to be Learned in Computer Science Programs

The curriculum of a computer science program must require every student to have acquired knowledge and skills in each of the five areas in Section 12.2.1. An advanced skill level is not required in all five areas, but all students should acquire a reasonably broad knowledge of each area at the level typically provided by a first course in the area.

Advanced knowledge: In addition, each computer science student should acquire knowledge and skills at an advanced level, demonstrating both breadth and depth, in at least two key areas of computer science, and in one third of the suggested minimum number of computing courses described in Section 12.2.1a (i.e., 5 advanced computing courses). Advanced level knowledge and skills refer to those that would normally be attained from a second or subsequent course in the area.

12.2.3 Key Areas of Computer Science to be Learned in Software Engineering Programs

Software engineering programs must require every student to have attained broad knowledge and

skills in each of the last four areas of Section 12.3.1 (Graduate Attributes 1b, 1c, 1d and 1e), and deeper knowledge in software engineering (Graduate Attribute 1a).

The following describes knowledge and skills in software engineering (Graduate Attribute 1a) that students in software engineering programs would be expected to have learned:

a) Software requirements (elicitation, analysis, specification, validation)
b) Software design and architecture (architectural structures, styles and patterns, structural vs. behavioural descriptions, strategies and methods)
c) Software construction and maintenance (abstraction mechanisms, assertions, debugging, coding style, programming environment and tools, including configuration management)
d) Software testing and quality assurance (unit, integration and system testing, black/white box testing techniques, validation and verification, reliability)
e) Software engineering management and process (metrics, estimating, planning, scheduling)
f) Applications of software engineering to various areas (embedded, real-time or distributed systems, database-centric systems, etc.)
g) Human computer interaction
h) Standards of Practice (Practice Guides, Recommended Practices, Practice Standards), including those endorsed by CIPS (e.g., ISO/IEC 12207)

It would normally be the case that the proper acquisition of software engineering knowledge and skills in these areas would imply the completion of at least three (3) courses in software engineering.

Breadth in key areas of computer science: In addition to knowledge of software engineering as defined above, students should acquire a reasonably broad knowledge in each of the other key areas of computer science defined in Graduate Attributes 1b, 1c, 1d and 1e. It is suggested that this be at a level typically provided by a first course in each area.

Advanced knowledge: Each software engineering student should acquire knowledge and skills at an advanced level, demonstrating both breadth and depth, in at least two key areas of computer science, one of which would be software engineering. Also, each student should acquire advanced knowledge and skills in one third of the suggested minimum number of computing courses described in Section 12.2.1a (i.e. five (5) advanced computing courses). Advanced level knowledge and skills refer to those that would normally be attained from a second or subsequent course in the area.

12.2.4 Coverage of the Key Areas for Interdisciplinary Programs

All students in an accredited interdisciplinary program must have attained knowledge and skills in at least three of the five areas listed in Section 12.3.1. This knowledge and these skills must include Graduate Attribute 1b (Algorithms and data structures) and specifically include coverage of data structures, abstract data types and established solutions to classical problems (e.g., searching and sorting). A high level of skill is not required in all three of these topics, but all students should acquire a reasonably broad knowledge of each topic at the level typically provided in a first course.

Advanced knowledge: In addition, students in interdisciplinary programs should acquire knowledge and skills at an advanced level, demonstrating both breadth and depth, in at least two key areas of computer science (Graduate attributes 1a, 1b, 1c, 1d and 1e). Advanced level knowledge and skills refer to those that would normally be attained from a second or subsequent course in the area.

12.2.5 Exposure to Multiple Programming Languages and Paradigms

Students graduating from an accredited program should have proficiency in at least one programming language and exposure to a variety of programming languages. Exposure to a variety of programming paradigms – procedural, object-oriented, logical or functional, sequential and concurrent – is also important.

12.2.6 Significant Design Experience [Graduate Attribute 3]

Students graduating from an accredited program should have had the chance to develop a complete significant system, or make a major modification to an existing system, at some point in their studies, whether it be in course projects, a final fourth-year project, or an internship or in some other manner. This design experience should be open-ended in the sense that there is 'no right answer', and should enable the student to integrate their knowledge from most, if not all, of the areas of computer science listed in Section 12.3.1, as well as knowledge of mathematics, domain knowledge and, where appropriate, with consideration for economics, societal issues, safety, etc.

12.2.7 State of the Art Tools and Practices [Graduate Attribute 4]

Accredited programs must prepare students to meet the computing challenges they will face after graduation, whether they embark on careers immediately or continue their education. Thus, as part of their undergraduate education, students must be well-grounded in state-of-the-art computing practices. An accredited program should expose students to several computing configurations, including varied hardware, operating systems, and programming environments.

12.2.8 New Areas of Computing

Computer Science is a rapidly developing and growing subject. At the current time, these newer developments include such areas as mobile devices and networks, health and medical informatics, cyber security, bioinformatics, data mining, quantum computation, AI, and so forth. While it is not to be expected that an accredited program will include material in all of these areas, accredited programs should nonetheless demonstrate that they recognize the rapidly evolving nature of the subject and should include some of the newer areas of the subject, particularly within the intermediate and advanced courses which they offer to their students.

12.3 Mathematics [Graduate Attribute 1f to 1g]

12.3.1 Mathematics for Computer Science and Software Engineering Programs

Students in accredited computer science and software engineering programs must have attained knowledge and skills in the following three areas: (i) discrete mathematics, (ii) differential and integral calculus, and (iii) probability and statistics. Students are expected to receive a solid grounding in logic, Boolean algebra and graph theory. Other topics of particular relevance to computing should also be offered, including a selection from linear algebra, set theory and modern algebra, numerical analysis, differential equations, optimization, and queuing theory.

12.3.2 Mathematics for Interdisciplinary Programs

Students in accredited interdisciplinary programs must have attained knowledge and skills in the following two areas (i) discrete mathematics, and (ii) probability and statistics. In addition, students are expected to learn the basics of logic, Boolean algebra and the basics of graph theory.

12.4 Breadth and Depth in Topics Outside Computing and Mathematics [Graduate Attribute 9]

Computing applications can be found in all human endeavours, so education in any discipline outside computer science has the potential to prepare students in a field of direct relevance to their future livelihood.

The diversity of backgrounds needed by various computing professionals necessitates flexibility within these accreditation requirements. Innovation in establishing institutional requirements or in promoting everyone's ability to reach personal goals is encouraged. Nevertheless, this section identifies topics that enhance a program's ability to meet the needs of computing students. As before, the goal of breadth in exposure must be balanced by the goal of depth in understanding, which can best be achieved through the selection of complementary courses having a common focus.

12.4.1 Details of Breadth Requirements for Computer Science and Software Engineering Programs

Students in computer science and software engineering should be encouraged to choose courses such that their programs include sufficient breadth in other domains, so as to be able to communicate with people working in those domains, to develop software for those domains, and to in general be well-rounded graduates. It is suggested that to meet this requirement, students be required to take:

- Five (5) courses in science, engineering (but not computer or software engineering), or business
- Five (5) courses in humanities or social science

Topics in physics and electrical engineering are basic to many aspects of computing, and courses in these areas are particularly encouraged. Challenges such as the endeavour to map the human genome underline the value of education in other fields of natural science as well, including chemistry, the earth sciences, and the life sciences, especially when integrated with computer studies.

A thorough grounding in business fundamentals is important to prepare students of computer science to contribute to Canadian industry. Relevant knowledge and skills in this area include particularly accounting, business organization, economics, and auditing, but also include marketing, personnel management, and production management.

Students graduating in computing must also exhibit skills in non-technical disciplines. Through knowledge and skills in humanities or social science, students will gain understanding of political theories and processes, knowledge of individual and group social interactions, appreciation of cultures and history, sensitivity to the literary and fine arts, and fluency in languages. Not only is this important background for future self-study, but several aspects have direct bearing on particular areas of computing: linguistics is central to natural language processing, cognitive science provides mechanisms to evaluate human-computer interaction, law can be applied to assessing liability of computer professionals, and ethics helps to evaluate social implications of computing.

12.4.2 Details of the Other Discipline(s) for Interdisciplinary Programs

The Other Disciplines of interdisciplinary programs (B in Section 12.1.2) combined must be the 'equal' of Computer Science and must be designed to provide a coherent education. In order to achieve this, it is suggested that:

A. The courses in the Other Disciplines may include courses supporting those disciplines, and each Other Discipline may include courses from several related departments.
B. There be at least ten (10) courses in the Other Disciplines, with no single discipline having less than five (5) courses.
C. At least two (2) of the courses in each Other Discipline should be advanced courses, defined as courses that would normally be taught in the latter two years of study and build upon the introductory and intermediate courses.
D. There must be a structure to the set of courses required in the Other Disciplines; in other words, allowing students to choose any random set of 10 courses is not appropriate, although allowing students to select from several groups of electives would be fine. Allowing students to select a custom program would also be fine provided this process is carefully guided by an advisor, such that the resulting program is coherent and meets the program objectives.
E. It is highly desirable, although not essential, for courses in the Other Disciplines to build upon computer science and mathematics courses, and vice versa. For example, a course in another discipline could involve computing, and a course in computer science could present application programs related to the other discipline when illustrating computing concepts.
F. There must be an approval process in the institution for the sets of courses comprising the Other Disciplines, and there must be a process in place within the institution to regularly evaluate the curriculum in the Other Disciplines as well as the delivery of the courses.

12.4.3 Details of the Breadth Requirement for Interdisciplinary Programs

The breadth requirement of interdisciplinary programs (D in Section 12.1.2) ensures that students attain knowledge and skills beyond the purely technical. Through knowledge and skills in humanities or social science, students will gain understanding of political theories and processes, knowledge of individual and group social interactions, appreciation of cultures and history, sensitivity to the literary and fine arts, and fluency in languages. Not only is this important background for future self-study, but several aspects have direct bearing on particular areas of computing: linguistics is central to natural language processing, cognitive science provides mechanisms to evaluate human-computer interaction, law can be applied to assessing liability of computer professionals, and ethics helps to evaluate social implications of computing.

A grounding in business fundamentals is useful to prepare students of computer science to contribute to Canadian industry. Relevant knowledge and skills in this area include accounting, business organization, economics, and auditing, but also include marketing, personnel management, and production management.

12.5 Non-Trivial Problem Solving in Teams [Graduate Attributes 2 and 5]

A significant component of an accredited program must be practical in nature; students must have direct experience with non-trivial problem-solving. This component helps to develop students' creativity in solving open-ended problems through practice in formulating problem statements and specifications, considering alternative solutions, determining feasibility and cost, and communicating the results including detailed systems descriptions. Projects and courses that require teamwork are strongly encouraged.

In the case of software engineering programs, students must have direct experience with the major phases of the software development life cycle.

Whereas practical aspects should be included at all levels of the program, the major portion of the practical component is to be satisfied in advanced courses. Students, individually and as

members of teams, must be required to design and implement a system, program, process, or device to achieve stated objectives.

The problem solution should include the establishment of objectives and criteria, analysis, synthesis, documentation, implementation, testing, and evaluation. It is desirable that problems include a variety of realistic constraints such as economic factors, performance thresholds, ergonomics, compliance with standards, interoperability with other systems, and conformance to ethical, professional and legal restrictions.

12.6 Written and Oral Communication Skills [Graduate Attribute 6]

As an adjunct to specific areas of instruction, the program must also include training to develop students' written and oral communications skills. Not only should students be taught to present information both verbally and in writing, but they should practice collecting information through reading and listening and assembling the information for various audiences. The realization of this requirement in the program can be tailored to suit other institutional goals, but the involvement of the computer science faculty must be evident.

12.7 Professionalism [Graduate Attribute 7]

Aspects of professionalism are to be emphasized throughout the curriculum. A specific course or courses in social implications of computing may be offered, but ethical and legal issues surrounding computing, including the social responsibility of programmers and computer users, must be emphasized in courses throughout the program so that students learn that these aspects are part of computing, not merely tangential disciplines.

12.8 Coordination of Program Schedules in Interdisciplinary Programs

A particular concern in interdisciplinary programs is that students are able to take the courses they need and want from the collaborating disciplines, without conflicting schedules unduly restricting their choice, or requiring the program to take excessive time to complete.

13. College Curriculum Accreditation Specific *Criteria*

13.1 Technical Curriculum - Overall Distribution of Courses

For each range of graduate attributes, it is expected that an accredited program will have the necessary duration and deliver the required instructional unit or course to fulfill the graduate attribute requirements. Furthermore, the program must ensure that *all* students must meet the graduate attributes to be eligible for the credential. This can be especially important where students are permitted electives or alternate courses for graduation.

The next two subsections describe the distribution of instructional units or courses that would generally be expected for different types of programs to fulfill the graduate attributes of the three ranges. The intent here is **NOT** to prescribe specific courses or topics but to illustrate how typical courses can map to the corresponding range of graduate attributes in terms of duration, breadth and depth. Institutions may design their programs to meet the graduate attribute requirements in other ways. The priority is to demonstrate that the proper range of graduate attributes has been met for the program undergoing accreditation.

13.1.1 Typical Instructional Durations to Fulfill Graduate Attributes

A. 70% of instructional activity in computer science, information systems, or information technology (Graduate Attributes 1a thru 1e).
B. 10% in mathematics or statistics (Graduate Attributes 1f and 1g), and
C. 20% of courses in topics other than ICT or mathematics/statistics (Graduate Attributes 7 thru 9).

ICT programs and courses are considered as an integrated whole and often will meet graduate attributes through combinations of courses and objectives, rather than being restricted to individual courses. In the following section, we offer guidance, arranged by Graduate Attribute, as to how ICT programs often meet graduate attribute requirements.

13.2 Breadth of Knowledge Graduate Attribute

Graduate Attribute 1a: Software Engineering including requirements specification, software design, software development, software testing, software maintenance, and other topics related to software development life cycle (SDLC) and adaptive methodologies.

Graduate Attribute 1b: Algorithms and Data Structures including data structures such as stacks, trees, lists, queues, database management, etc.

Graduate Attribute 1c: Systems Software including operating systems concepts, transaction processing, logging, security, mobile, cloud, IoT, and computer networking.

Graduate Attribute 1d: Computer Elements and Architecture including computer organization, digital device and communications technology, logical and physical hardware design.

Graduate Attribute 1e: Theoretical Foundations of Computing including models of computation, fundamentals of program specification and verification.

Graduate Attribute 1f: Discrete Mathematics where students are expected to receive a solid grounding in logic and Boolean algebra.

Graduate Attribute 1g: Probability and Statistics where students must have attained sufficient mathematics to meet their program objectives.

13.3 Depth of Knowledge [Graduate Attribute 2]

Basic knowledge of each area (1a – 1e) at the level typically provided by a first course, instructional unit or equivalent in the area. Advanced knowledge in at least one key area typically from a second or third course or instructional unit in the area.

13.4 Non-Trivial Problem-Solving in Teams [Graduate Attribute 2, 3, 4 and 6]

A significant component of an accredited program is direct experience with non-trivial problem-solving. This component helps to develop students' creativity in solving open-ended problems through practice in formulating problem statements and specifications, considering alternative solutions, determining feasibility and cost, and communicating the results including detailed systems descriptions. Projects and courses that require teamwork are strongly encouraged. Students, individually and as members of teams, must be required to design and implement a system, program, process, or device to achieve stated objectives. Team-based learning does not imply face-to-face modalities for teamwork; in fact, it may be appropriate for teams to complete at least part of their work using electronically mediated and/or asynchronous means.

The problem solution should include the establishment of objectives and criteria, analysis, synthesis, documentation, implementation, testing, and evaluation. It is desirable that problems include a variety of realistic constraints such as economic factors, performance thresholds, ergonomics, compliance with standards, interoperability with other systems, and conformance to ethical, professional and legal restrictions.

Must complete a team project or equivalent.

13.5 Significant Design / Project Experience Teams [Graduate Attribute 3, 4 and 6]

Students graduating from an accredited program should have had the chance to develop or modify a complete significant system, or make a major modification to an existing system, at some point in their studies, whether it be in course projects, a final project, and an internship or in some other manner. This design experience should be sufficient to enable the student to integrate their knowledge from most, if not all, of the areas of ICT, as well as knowledge of mathematics, domain knowledge and, where appropriate, with consideration for economics, societal issues, safety, etc.

13.6 State-of-the-Art Tools and Practices [Graduate Attribute 4 and 5]

Accredited programs must prepare students to meet the computing challenges they will face after graduation, whether they embark on careers immediately or continue their education. Thus, as part of their education, students must be sufficiently well-grounded in state-of-the-art computing practices to meet the needs of their program. An accredited program should expose students to several computing configurations, including varied hardware, operating systems, and programming environments.

An accredited program should expose students to several computing configurations, including varied hardware, operating systems, and programming environments.

13.7 Exposure to Multiple Programming Languages and Paradigms [Graduate Attribute 5]

Students graduating from an accredited program should have proficiency in at least one programming language and exposure to a variety of programming languages. Exposure to a

variety of programming paradigms – procedural, object-oriented, logical or functional, sequential and concurrent – is also important.

13.8 New Areas of Computing [Graduate Attribute 5]

Computer Science is a rapidly developing and growing subject. At the current time, these newer developments include such areas as mobile devices and networks, health and medical informatics, cyber security, bioinformatics, data mining, quantum computation, augmented and virtual reality, machine learning, artificial neural networks, robotics, and so forth. While it is not to be expected that an accredited program will include material in all of these areas, accredited programs should nonetheless demonstrate that they recognize the rapidly evolving nature of the subject and should include some of the newer areas of the subject, particularly within the intermediate and advanced courses which they offer to their students.

Accredited programs should expose students to the rapidly evolving nature of the subject and should include some of the newer areas of the subject, particularly within the intermediate and advanced courses which they offer to their students.

13.8 Written and Oral Communication [Graduate Attribute 7, 8 and 9]

The expanding and demanding role of modern information systems dictates strong interpersonal and teamwork skills requirements of the IT practitioner. The ability of the graduate to communicate has a direct bearing on the quality of "on-the-job" performance. To facilitate the participation of the graduate in this environment, students are expected to develop the following proficiencies:

- ✓ oral presentation and the use of presentation tools: projectors, graphics, multi-media
- ✓ writing skills: letters, memos, business reports, electronic communications media
- ✓ use of productivity software
- ✓ interpersonal skills: listening skills, interviewing, analyzing problem situations, teamwork, leadership, organizational skills
- ✓ workplace and employment skills
- ✓ interpersonal, customer relations and user support skills
- ✓ time management skills
- ✓ face-to-face and electronic communications
- ✓ remaining current (technical and non-technical)
- ✓ training and knowledge transfer
- ✓ ethical and legal impact of information systems and emphasize on professionalism

The program should include curriculum components designed to encourage students to apply these proficiencies throughout their studies. These components may be delivered in courses specifically designed to teach certain of these skills and taught by specialists in the area (e.g., communication or technical writing courses taught by a Communication or English department) or be embedded as a formal component in other courses or learning activities throughout the program.

13.9 Professionalism [Graduate Attribute 8 and 9]

Aspects of professionalism are to be emphasized throughout the curriculum. A specific course or courses in social implications of computing may be offered, but ethical and legal issues surrounding computing, including the social responsibility of programmers and computer users, must be emphasized in courses throughout the program so that students learn that these aspects are part of computing, not merely tangential disciplines. Topics can be covered in a dedicated course or embedded throughout.

13.10 Breadth and Depth in Topics Outside Computing and Mathematics [Graduate Attribute 10]

Computing applications can be found in all human endeavours, so education in any discipline outside computer science has the potential to prepare students in a field of direct relevance to their future livelihood.

The diversity of backgrounds needed by various computing professionals necessitates flexibility within these accreditation requirements. Innovation in establishing institutional requirements or in promoting everyone's ability to reach personal goals is encouraged. Nevertheless, this section identifies topics that enhance a program's ability to meet the needs of ICT students. As before, the goal of breadth in exposure must be balanced by the goal of depth in understanding, which can best be achieved through the selection of complementary courses or equivalent learning experiences having a common focus.

Students should be encouraged to choose courses such that their programs include enough breadth in other domains, to be able to communicate with people working in those domains, to develop software for those domains, and to in general be well-rounded graduates. It is suggested that to meet this requirement, students be required to take:

Topics in physics and electrical engineering are basic to many aspects of computing, and knowledge in these areas are particularly encouraged. Challenges such as the endeavour to map the human genome underline the value of education in other fields of natural science as well, including chemistry, the earth sciences, and the life sciences, especially when integrated with computer studies.

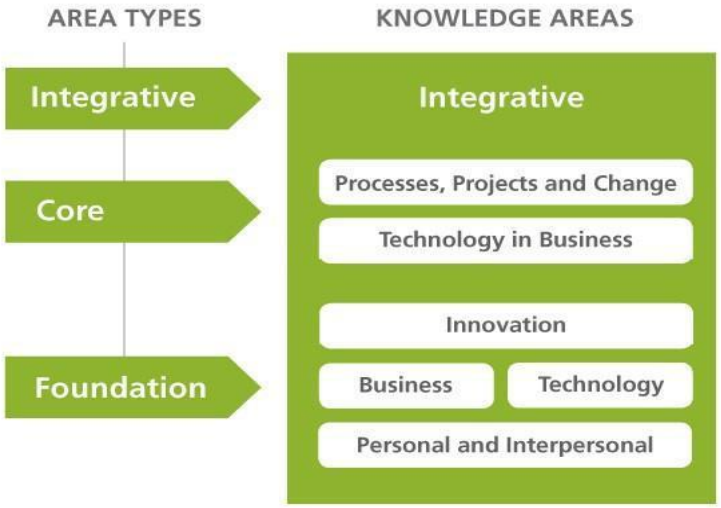
A thorough grounding in business fundamentals is important to prepare students of computer science to contribute to Canadian industry in domains such as: eHealth, Public/Private sectors, Education, Manufacturing to name but a few. Relevant knowledge and skills in this area include particularly accounting, business organization, economics, and auditing, but also include marketing, personnel management, and production management.

Students graduating in computing must also exhibit skills in non-technical disciplines. Through knowledge and skills in humanities or social science, students will gain understanding of political theories and processes, knowledge of individual and group social interactions, appreciation of cultures and history, sensitivity to the literary and fine arts, and fluency in languages. Not only is this important background for future self-study, but several aspects have direct bearing on particular areas of computing: linguistics is central to natural language processing, cognitive science provides mechanisms to evaluate human-computer interaction, law can be applied to assessing liability of computer professionals, and ethics helps to evaluate social implications of computing.

It is also recognized that educational institutions will often impose specific breadth requirements on programs which must be accommodated.

14. Business Technology Management Curriculum Accreditation Specific

14.1 Hierarchical Structure of the BTM 2.0



14.2 BTM Competency Standard Framework

The BTM 7 broad competency areas, are:

1. **Integrative (I1)**: This knowledge area contains learning outcomes that integrate the competencies developed in the following six knowledge areas. It produces a "deliverable" of direct relevance to employers.

2. **Personal and Interpersonal (F1)**: The ability to make a meaningful contribution depends upon one's self-knowledge and ability to have constructive, long term, interactions with others. Successful leaders have strong personal and interpersonal competencies.

3. **Business (F2)**: To be effective in the workplace one must have both the broad context of business – its role and place in society – and a working knowledge of how business operates.

4. **Technology (F3)**: BTM graduates must understand information and communications technologies, their current capabilities, and future trends.

5. **Innovation (F4):** BTM graduates are expected to be innovative in the workplace. Innovators should be able to identify new opportunities, validate and resource them.

6. **Technology in Business (C1)**: This knowledge area is designed to synthesize the knowledge and competencies gained in the foundational knowledge areas and create an additional competency in understanding: the potential (economic, personal, societal), the risks of, and the governance, acquisition, and management of ICTs in and for business.

7. **Processes, Project and Change (C2)**: BTM graduates will gain the foundations that enable them to help create well-designed business processes, well-managed projects, and support for the individuals and groups undergoing change.

14.3 BTM Program Characteristics Typically Associated with Meeting Graduate Attribute

ICT programs and courses are considered as an integrated whole and often will meet graduate attributes through combinations of courses and objectives, rather than being restricted to individual courses. In this section, we offer guidance, arranged by Graduate Attribute, as to how ICT programs often meet graduate attribute requirements.

Graduate Attribute	BTM Curriculum Characteristics
1. Academic Education	As a four-year Bachelor of Commerce, the BTM program provides advanced academic education in business and technology. Programs should cover broad fundamentals in business (economics, finance, accounting, management, marketing) plus specialized ICT-related courses.
2. Knowledge for Solving Computing Problems	This attribute expects graduates to apply knowledge of computing fundamentals… and domain knowledge… to the abstraction and conceptualization of computing models from defined problems. BTM curriculum should include several technical courses that build computing fundamentals, as well as teach object-oriented programming basics (objects, classes, inheritance, polymorphism) for business problem-solving, database models and query design, web and Internet programming for business systems, and applies object-oriented design (UML, use-case models, class diagrams) to build system prototypes. Together these should ensure students learn computing fundamentals (programming, data management, systems design) in a business context.
3. Problem Analysis	This attribute expects graduates to identify, formulate, research literature, and solve complex computing problems… using fundamental principles of mathematics, computing sciences, and relevant domain disciplines. In BTM programs, students practice analysis by decomposing business/IT problems. Using technics such as: the first phase of the system development life cycle: surveying existing systems, identifying deficiencies, developing functional requirements and feasibility/cost analyses, model problems with programming logic and data schemas. While the problems are mostly well-bounded business cases (not research-level complexity), the curriculum should require analytical rigor (use of UML, feasibility studies, etc.).
4. Design/Development of Solutions	This attribute expects graduates to design solutions to meet specified needs (with consideration of societal and safety issues). In BTM programs, students should be able to transform analysis requirements into design and solutions incorporating current and/or emerging information systems to address specific organizational challenges.
5. Modern Tool Usage	This attribute expects graduates to "create, select, adapt and apply appropriate modern computing tools… to computing activities, with an understanding of limitations". Overall, graduates should be able to apply current IT tools and platforms in their projects, such as: IDEs, DBMS, SQL, ERP systems,

Graduate Attribute	BTM Curriculum Characteristics
	Project Management tools, to name a few.
6. Individual and Teamwork	This attribute expects graduates to function effectively as an individual and as leader/member of diverse teams. Teamwork and leadership should be strongly embedded in the curriculum via: group work, interpersonal skills, conflict management, leadership roles, critical thinking in teams.
7. Communication	This attribute requires graduates to effectively communication with the computing community and society at large by writing reports, documentation, presentations, instructions on technical topics to diverse audiences.
8. Professionalism and Society	This attribute involves understanding societal, legal, safety, cultural issues of professional practice. BTM programs graduates should have touched on the following aspects: ethics, sustainability and social responsibility, ethical reasoning and stakeholder theory in business, IT governance and security emphasises protecting data and business continuity, which shows awareness of the broader context of computing. However, the most focus is on business ethics and governance rather than, say, public health or cultural diversity in tech.
9. Ethics	Closely related, CIPS Ethics attribute expects a commitment to professional ethics and norms. BTM programs curriculum should explicitly include ethics topics, such as: ethical reasoning in sustainability contexts, legal and ethical workplace issues (whistleblowing, corporate codes, liabilities). While these are general business ethics, they apply equally to IT managers (e.g. data privacy, compliance). graduates are taught professional ethics, but the emphasis is on business ethics rather than, say, an IT-specific code of ethics.
10. Lifelong Learning	This attribute expects graduates to recognize the need and engage in independent learning for continued professional growth. While they are students they are expected to be curious and adaptable in lifelong learning.

15. Innovation and Research

Although research may or may not, depending on the program and/or institution, be considered to be mandatory for a school to fulfill its mandate, such an involvement can be considered indicative of an ongoing commitment to innovation and excellence. This innovation could take the form of the devising of new pedagogical approaches, or the development of new course notes and manuals. Research work could be evidenced through the personal involvement of faculty members in the use of computers, or in their collaboration with local businesses, university or government research centres.

Appendix A - Graduate Attributes (Seoul Accord 2008). (New 2026 GA to be inserted)

	Attributes	Differentiating Characteristic	… for Computing Professional Graduate	… for Computing Technologist Graduate	… for Computing Technician Graduate
1	Academic Education	Educational depth and breadth	Completion of an accredited program of study designed to prepare graduates as computing professionals.	Completion of a program of study typically of shorter duration than for professional preparation.	Completion of a program of study typically of shorter duration than for technologist preparation.
2	Knowledge for Solving Computing Problems	Breadth and depth of education and type of knowledge, both theoretical and practical	Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.	Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to defined and applied computing procedures, processes, systems, or methodologies.	Apply knowledge of computing fundamentals, knowledge of a computing specialization, and mathematics, science, and domain knowledge appropriate for the computing specialization to a wide variety of practical procedures and practices.
3	Problem Analysis	Complexity of analysis	Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.	Identify, formulate, research literature, and solve broadly-defined computing problems reaching substantiated conclusions using analytical tools appropriate to the discipline or area of specialization.	Identify and solve well-defined computing problems reaching substantiated conclusions using codified methods of analysis specific to the field of activity.

	Attributes	Differentiating Characteristic	… for Computing Professional Graduate	… for Computing Technologist Graduate	… for Computing Technician Graduate
4	Design/ Development of Solutions	Breadth and uniqueness of computing problems, i.e., the extent to which problems are original and to which solutions have previously been identified or codified.	Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.	Design solutions for broadly defined computing technology problems, and contribute to the design of systems, components, or processes to meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.	Design solutions for well-defined computing problems, and assist with the design of systems, components, or processes to meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
5	Modern Tool Usage	Level and appropriateness of the tool to the type of activities performed.	Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.	Select and apply appropriate techniques, resources, and modern computing tools to broadly- defined computing activities, with an understanding of the limitations.	Apply appropriate techniques, resources, and modern computing tools to well-defined computing activities, with an awareness of the limitations.
6	Individual and Teamwork	Role in, and diversity of, the team	Function effectively as an individual and as a member or leader in diverse teams and in multi- disciplinary settings.	Function effectively as an individual and as a member or leader in diverse technical teams.	Function effectively as an individual and as a member in diverse technical teams.

	Attributes	Differentiating Characteristic	… for Computing Professional Graduate	… for Computing Technologist Graduate	… for Computing Technician Graduate
7	Communication	Level of communication according to the type of activities performed.	Communicate effectively 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.	Communicate effectively with the computing community and with society at large about broadly defined computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.	Communicate effectively with the computing community and with society at large about well-defined computing activities by being able to comprehend the work of others, document one's own work, and give and understand clear instructions.
8	Computing Professionalism and Society	No differentiation in this characteristic except level of practice	Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.	Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to computing technologist practice.	Understand and assess societal, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to computing technician practice.
9	Ethics	No differentiation in this characteristic except level of practice.	Understand and commit to professional ethics, responsibilities, and norms of professional computing practice.	Understand and commit to professional ethics, responsibilities, and norms of computing technologist practice.	Understand and commit to professional ethics, responsibilities, and norms of computing technician practice.

	Attributes	Differentiating Characteristic	… for Computing Professional Graduate	… for Computing Technologist Graduate	… for Computing Technician Graduate
10	Life-long Learning	No differentiation in this characteristic except level of practice.	Recognize the need and can engage in independent learning for continual development as a computing professional.	Recognize the need and can engage in independent learning for continual development as a computing technologist.	Recognize the need and can engage in independent learning for continual development as a computing technician.

Appendix B - CIPS Accreditation Process

B.1 CIPS Accreditation Process

A standard level of professional knowledge among all I.T. professionals rely on well understood education approaches and curricula. CIPS has developed a rigorous process to examine and accredit postsecondary computer programs. The history of the accreditation of Computing and I.T. programs originally stemmed from a need to promote and advance all phases of this type of education with a broader aim to promote public welfare through the development of computing professionals who were better educated and able to deal with the demands of industry. With the introduction of the Information Systems Professional of Canada (I.S.P.) program in 1989, the accreditation program was aligned with the requirements for professional membership. Applicants for registration as an I.S.P. are assessed for evidence that their academic background and professional work experience meet the standards for professional membership in the association. CIPS accreditation offers a process for evaluating educational programs at a level of performance, integrity and quality to inform and protect the public. The accreditation process is ongoing. The five key features of accreditation are:

- Self Study: institutions prepare a written summary of performance based on the accreditation criteria. The self-study provides a foundation for the team.
- Peer Review: the accreditation review is conducted by primarily faculty, college administrators and members of industry.
- Site Visit: the accreditation review is conducted by a team of experienced accreditors. This offers an opportunity to the team to meet with a broad spectrum of individuals such as faculty, advisory committee, students, graduates and senior administration.
- Action (judgment) of accreditation agency: The CIPS Accreditation Council have commission that makes decisions about the accredited status of the program.
- Monitoring and oversight: programs are reviewed over time in cycles of three years. Any major program changes need to be presented to the accreditation council to determine if and how they affect the accreditation status.

B.2 Accreditation Decisions and Term

Accreditation of a program is granted for a specific period, usually five years for universities and three years for colleges. Accreditation is granted if current conditions are found to be meeting the minimum requirements. If, for any reason, the future of a program appears precarious or definite weaknesses exist, the accreditation will be granted for a shorter period. Factors that may limit the period of accreditation include an inability to meet the Council's curriculum criteria, a need for significant additions or improvements in staff or equipment, an uncertainty as to Department or the educational institution's financial status, the nature of the administrative organization, etc.

Shorter Accreditation Period

If an evaluation indicates that the future of a program appears precarious or that there is/are definite weakness(es) or deficiencies exist, a shorter accreditation period may be granted followed

by an "interim review". The accreditation decision will be specific in terms of what criteria need to be satisfied before full accreditation is granted. The interim review of the shorter accreditation can take on two forms:

- *Interim Report*: The nature of the weakness is such that an on-site visit will not be required to evaluate the remedial actions taken by the institution. A report focusing on the remedial actions taken by the institution will be required.
- *Interim Visit*: The nature of the weakness is such that an on-site visit will be required to evaluate the remedial actions taken by the institution.

The Interim Report and the Interim Visit may result in an extension of the accreditation period.

Not-to-Accredit

A "Not-To-Accredit" decision indicates that a program has deficiencies such that the program is currently or continues to be (in the case of an Interim Visit or Report) in non-compliance with the applicable criteria.

Renewal of Accreditation

An accredited educational institution must apply for renewal of the accreditation after completing the accreditation term. The next accreditation application must be submitted to the CIPS Secretariat at least 12 months prior to the current accreditation termination date.

Revocation of Accreditation

If, during the period of accreditation, the Council has reason to believe that a program is no longer in compliance with the criteria, the institution will be notified of such reason and will be requested to provide a response. If the response is not adequate, Council may institute a revocation of the accreditation decision.

Revocation begins with the notification of the institution as to the reasons why the procedures are being implemented. An on-site one-day evaluation may be scheduled to determine the facts. The one-day review will include two team members. A comprehensive document showing the reasons for revocation will be prepared and provided to the institution for a response. If the institution's response is not adequate, accreditation will be revoked. The institution is notified of such action together with a supporting statement showing the cause. Revocation constitutes a Not-To-Accredit action and may be appealed.

B.3 The Appeal Process

Appeals, requests for reconsideration, and requests for a revisit may be done only in response to a Not-To-Accredit decision. Also, appeals or requests for reconsideration can be based only upon the grounds that the Not-To-Accredit decision of the Council was inappropriate because of errors of fact or failure to conform to the Council's published criteria.

In lieu of an immediate appeal, an institution may first request reconsideration. If such a request is denied, the institution may appeal the original Not-To-Accredit decision. Requests for reconsideration must be made in writing to the Chair of the Council within 30 days of receiving notification of the decision.

Appeals must also be made in writing to the Chair of the Council within 30 days of receiving notification of the decision or the notification of the denial of a request for reconsideration.

Reconsideration

A program that has received a Not-To-Accredit action may be a candidate for reconsideration if it can demonstrate that there were major, documented errors of fact in the information used by the Council in arriving at the Not-To-Accredit decision.

In such cases, the institution must submit a written request for reconsideration to the Chair of the Council within 30 days of receiving notification of the Not-To-Accredit action. This request must be accompanied by a report specifying the major, documented errors of fact and how such errors contributed to the decision, along with substantive documentation.

The Chair shall review the request for reconsideration and either accept or deny the request within 15 days of receipt of the request. If a revisit is not warranted, a statement of reasons will accompany the letter from the Chair.

If an immediate revisit is warranted, the institution is deemed to have waived its right to an appeal either the original decision or the action that will result from the revisit. If the revisit is granted, the institution will be charged the regular visitation fee for the revisit.

If, following the immediate revisit, the Council, upon unanimous vote, judges that the institution is correct in its claim of substantive improvements; the Council may overturn the Not-To-Accredit decision and grant whatever accreditation action is deemed appropriate.

Appeal

A notice of appeal must be submitted in writing to the Chair of the Council within 30 days of receiving a notification of Not-To-Accredit. The submission must include the reasons why the decision is inappropriate because of either error of facts or failure of the Council to confirm to the published criteria.

The Chair of the Council will select three or more members or past members of the Council to serve as an appeal committee. The Chair will designate one of the committee members as Head of the committee.

The appeal committee will be provided with copies of all documentation that has been made available to the institution during the various phases of the accreditation cycle. The appeal committee will consider only the written materials submitted by the institution in determining its recommendation. Representatives

from the institution may not attend the meetings with the appeal committee. Improvements made to a program subsequent to the completion of the initial team visit will not be considered by the appeal committee.

The appeal committee's decision will be reported to the Council in writing by the appeal committee Head. The decision rendered by the appeal committee is advice it will give to the Council. The Council will review the appeal committee's decision and then make the final decision on the appeal.

The institutions will be notified in writing of the decision, and the basis for the decision, within 15 days of the final decision.

B.4 Changes During Period of Accreditation

It is the obligation of the institution to notify the Council of any significant changes that might affect the accreditation status of an accredited program including significant changes in:

- Program title
- Faculty
- Curricular Objectives
- Curricular Content
- Student Body
- Administration
- Institutional Facilities
- Institutional Commitment
- Institutional Financial Status

A review process will be initiated as a result of the notification by the institutions. The institution will be asked to provide information to the Council. The information does not need to be extensive but must provide enough detail about the change so the effect can be assessed. The Council will review the information within 15 days of receipt to determine if modification of the current accreditation is needed. The Council's decision will be based on the degree of certainty about whether the affected program continues to meet the appropriate accreditation decision. The institution will be notified of the Council determination in a timely manner.

If it is deemed that a revisit is required, and if an institution declines an immediate revisit, this shall be cause for revocation of accreditation of the program.

If an accredited program is terminated by the institution, accreditation by the Council is also automatically terminated.

B.5 Publication of Accreditation and Public Release Policy

A list of programs, which have been accredited by the Council, is published on the CIPS National website at www.cips.ca. Information as to whether a program or institution not on the accredited list has been under consideration by the Accreditation Council will NOT be made available except to the appropriate officials of the institution in question. The educational institution may use the accreditation term only while it is valid and only for the programs that are accredited. The following suggested wording will be provided to the educational institution:

> *The following programs have been accredited by the Canadian Information Processing Society's (CIPS) through its Accreditation Council. CIPS works with academic institutions to ensure that educational programs prepare students for the demands of the Computing and Information Systems profession. Completion of accredited computer information systems programs assist graduates in pursuing the Information Systems Professional of Canada (I.S.P.) designation. More information about CIPS and the professional certification program can be obtained at http://www.cips.ca or (905) 602 1370.*

Direct quotation in whole or part from any statement by the Council made to the institution is unauthorized. Correspondence and reports between the Council and the educational institution are confidential documents and should be released only to authorized personnel at the institution.

The educational institution must avoid any implication that programs offered are accredited under program criteria against which they have not been evaluated. Where sub-designations such as "option", "concentration", or similar are used for programs, the institution must clearly identify the program criteria under which accreditation has been obtained.

B.6 Definition of Report Terms

Strengths are those positive elements of the program that go beyond the normal expectations of the criteria, are valuable and should be continued. Strengths may influence a decision to accredit but do not guarantee it.

Concerns are related to issues that have been determined to fall below the normal expectations but still meet the minimum requirements of the criteria. Positive action should be considered.

Deficiencies indicate that a criterion is not satisfied. Specific action is required to bring the program into compliance with the criteria, policies or procedures prior to the next visit. Deficiencies may lead to an accreditation with conditions. Where an accreditation with conditions is not granted, deficiencies may lead to a mutual suspension of the accreditation process until such time as the deficiency is corrected. It should be noted that the draft report, prepared by the site visit team and presented to the Institution for initial comment, refers only to Deficiencies and does not specify which, if any, are considered Major. The decision as to which deficiencies, if any, are deemed Major, and the consequent decision on accreditation, is taken by Council once the final report has been prepared. Positive action is required to strengthen compliance with the criteria.

Major Deficiencies are issues that lead to a Not-To-Accredit decision. It should be noted that the draft report, prepared by the site visit team and presented to the institution for initial comment, refers only to Deficiencies and does not specify which, if any, are considered Major. The decision as to which deficiencies, if any, are deemed Major, and the consequent decision on accreditation, is taken by Council once the final report has been prepared.

Recommendations are linked to concerns and deficiencies and outline suggested actions that might be undertaken to strengthen the program. Recommendations are not intended to be prescriptive, and programs may choose from alternative actions to correct the deficiencies in question. Recommendations are expected to be addressed before the next accreditation review.