

# INFORMATIQUE APPLIQUÉE – RÉGIME COOPÉRATIF

<p><b>STAGE I</b> 2<sup>e</sup> année complétée (quatre sessions d'études)</p> <p>Travail sous supervision et entraînement initial</p>	<ul style="list-style-type: none"> <li>• Traitement de texte, chiffrier électronique.</li> <li>• Langages C++, C#, Java, Assembleur.</li> <li>• Capacité de modéliser un problème d'une manière structurée.</li> <li>• Connaissances approfondies de la programmation orientée objet : classes, objets, héritage, polymorphisme, encapsulation, généricité.</li> <li>• Circuits numériques: séquentiels et combinatoires, conception, microprogrammation.</li> <li>• Organisation de base des ordinateurs, représentation de données et codage d'instruction, registres, mémoires, modes d'adressage, interruption, langage assembleur.</li> <li>• Algorithmique : structures de données, récursivité, complexité d'algorithmes, classes de complexité, décidabilité.</li> <li>• Introduction au développement de logiciels : cycle de vie, analyse, spécifications, stratégies de résolution, mise en œuvre, validation et vérification, interfaces graphiques.</li> <li>• Langages de programmation : types, gestion de l'allocation dynamique de la mémoire, compilateurs, interpréteurs.</li> <li>• Introduction aux fichiers et systèmes de gestion de bases de données (SGBD).</li> </ul>
<p><b>STAGE II</b> Cinq sessions d'études complétées</p> <p>Travail sous supervision périodique</p>	<ul style="list-style-type: none"> <li>• Développement de logiciels : modèles du cycle de développement, gestion de projets, évaluation du risque, types de spécifications, les types de conceptions, validation et vérification, métriques, assurance de qualité, estimation de coûts et d'échéanciers.</li> <li>• Systèmes d'exploitation : structures et fonctions, tâches et processus, synchronisation, ordonnancement, gestion de la mémoire, sécurité et protection.</li> <li>• Réseaux de communication, protocoles de communication.</li> <li>• Architecture d'ordinateur : microprogrammation, microprocesseurs, superpipeline, superscalaire, cache de mémoire, architectures parallèles.</li> <li>• Langages de programmation et algorithmique: grammaires, compilateurs, interpréteurs, paradigmes de programmation, classes de complexité.</li> <li>• Langage Python.</li> <li>• Introduction aux algorithmes parallèles pour architectures parallèles et distribuées.</li> <li>• Introduction à l'intelligence artificielle : heuristiques, apprentissage machine, apprentissage profond.</li> </ul>
<p><b>STAGE III</b> Six sessions d'études complétées habituellement</p> <p>Travail semi-autonome sous supervision ponctuelle</p>	<p>L'étudiante ou l'étudiant aura ajouté de 6 à 8 cours à option en informatique dans l'une ou plusieurs des orientations suivantes :</p> <ul style="list-style-type: none"> <li>• Conception, développement et testage de logiciels.</li> <li>• Résolution efficace de problèmes.</li> <li>• Intelligence artificielle.</li> <li>• Cybersécurité &amp; cryptographie.</li> <li>• Bases de données relationnelles et distribuées.</li> <li>• Administration système.</li> <li>• Réseaux avec et sans fil.</li> <li>• Traitement d'images et vision par ordinateur.</li> </ul>

# BACHELOR OF COMPUTER SCIENCE – CO-OP PROGRAM

## SKILLS ACQUIRED DURING THEIR STUDIES

<p><b>Coop work-term I</b> 2<sup>nd</sup> year completed (Four study sessions completed)</p> <p>Work under supervision and initial training</p>	<ul style="list-style-type: none"> <li>• Word processing, spreadsheet.</li> <li>• Programming languages: C++, C#, Java, Assembly.</li> <li>• Structured approach to problem solving.</li> <li>• In depth knowledge of object-oriented programming: classes, objects, inheritance, polymorphism, encapsulation, genericity.</li> <li>• Digital circuits: sequential and combinatorial, design, micro-programming.</li> <li>• Basic computer organization, data representation and instruction coding, registers, memory, addressing modes, interruption, assembly language.</li> <li>• Algorithms and data structures, recursion, computational complexity, undecidability.</li> <li>• Software development: life cycle, analysis, specifications, resolution strategies, implementation, validation and verification, user graphical interfaces.</li> <li>• Programming: types, dynamic memory allocation, compilers, interpreters.</li> <li>• Introduction to files and database management systems (DBMS).</li> </ul>
<p><b>Coop work-term II</b> (Five study sessions completed)</p> <p>Work under periodic supervision</p>	<ul style="list-style-type: none"> <li>• Software development: development cycle models, project management, risk evaluation, types of specifications, types of conceptions, validation and verification, metrics, quality assurance, cost estimation and timelines.</li> <li>• Operating systems: structures and functions, tasks and processes, synchronization, scheduling, memory management, security and protection.</li> <li>• Computer networks and protocols.</li> <li>• Computer architecture: micro-programming, micro-processors, super-pipeline, super-scalar, memory cache, parallel architectures.</li> <li>• Programming languages and algorithms: grammars, compilers, interpreters, programming paradigms, complexity classes.</li> <li>• Python Language.</li> <li>• Introduction to artificial intelligence: heuristic search, machine learning, deep learning.</li> <li>• Introduction to parallel algorithms for parallel and distributed architectures.</li> </ul>
<p><b>Coop work-term III</b> (Six study sessions completed)</p> <p>Semi-autonomous work under limited supervision</p>	<p>The student will have added 6 to 8 computer science electives in one or more of the following orientations:</p> <ul style="list-style-type: none"> <li>• Software design, development and testing.</li> <li>• Efficient problem solving.</li> <li>• Artificial intelligence.</li> <li>• Cybersecurity &amp; cryptography.</li> <li>• Relational and distributed databases.</li> <li>• System administration.</li> <li>• Wired and wireless networks.</li> <li>• Image processing and computer vision.</li> </ul>