

MD-2045, CHIȘINĂU, STR. STUDENȚILOR, 9/7, TEL: 022 50-99-15,
www.calc.fcim.utm.md

F.O.008 TECHNIQUES AVANCÉES DE PROGRAMMATION

1. Données sur le sujet/module

Faculté	Ordinateurs, Informatique et Microélectronique				
Chaire/département	Informatique et Ingénierie des Systèmes				
Le cycle des études	Études supérieures de premier cycle, cycle I				
Programme d'études	0612.3 Science des Données				
Année d'étude	Semestre	Type d'évaluation	Catégorie formative	Catégorie d'optionnalité	Crédits ECTS
II (éducation à temps plein);	4	E	F – unité de cours fondamentale	O - unité de cours obligatoire	4

2. Temps total estimé

Total des heures dans le plan d'enseignement	Dont				
	Heures dans la salle		Travail individuel		
	Cours	Laboratoire/ Travaux pratiques	Projet annuel	Étude du matériel théorique	Préparation des applications
120	30	30		30	30

3. Prérequis pour accéder à la discipline

Selon le plan d'éducation	Programmation informatique, Programmation orientée objet
Selon les compétences	Acquisition de compétences pratiques en programmation orientée objet et utilisation de diverses procédures d'utilisation des objets.

4. Conditions de réalisation du processus éducatif pour

Cours	Pour la présentation du matériel théorique en salle de cours, un projecteur et un ordinateur sont nécessaires. Les retards des étudiants ainsi que les conversations téléphoniques pendant le cours ne seront pas tolérés.
Laboratoire/séminaire	Les étudiants prépareront des rapports conformément aux conditions imposées par les indications méthodologiques. Le délai de remise du travail de laboratoire – jusqu'au prochain travail de laboratoire. Pour une remise tardive du travail, celui-ci est pénalisé de 1 point/cycle de retard..

5. Compétences spécifiques accumulées

Compétences professionnelles	<p>CP2. Conception et développement d'applications (B.1)</p> <p>K1 Programmes/modules logiciels appropriés. K2 Composants matériels, outils et architectures matérielles. K3 Conception fonctionnelle et technique. K4 Technologies de pointe. K5 Langages de programmation. K6 Bases de données (DBMS). K7 Systèmes d'exploitation et plateformes logicielles. K8 Environnement de développement intégré (IDE - integrated development environment). K9 Développement rapide d'applications. K10 Problèmes liés aux droits de propriété intellectuelle (DPI).</p>
------------------------------	--

	<p>K11 Technologie de modélisation technique et langages. K12 Langages de définition d'interface (IDL). K13 Problèmes de sécurité. CP3. Intégration des composants (B.2) K1 Composants/modules matériels/logiciels, qu'ils soient anciens, existants ou nouveaux. K2 Impact de l'intégration d'un système sur l'organisation ou le système existant.</p>
	<p>K3 Techniques d'interface entre modules, systèmes et composants. K4 Techniques de test d'intégration. K5 Outils de développement (ex. environnement de développement, gestion, contrôle des modifications et accès au code source). K6 Bonnes pratiques de conception CP8. Marketing digital (D.12) K1 Stratégies de marketing. K2 Technologies web. K3 Moteurs de recherche marketing. K4 Optimisation des moteurs de recherche. K5 Marketing lié aux outils mobiles (ex. Pay Per Click). K6 Marketing lié aux médias sociaux K7 Marketing par e-mail. K8 Marketing display. K9 Problèmes/exigences légales</p>

6. Objectifs de la discipline

Objectif général	Acquisition des techniques de programmation structurée et de développement de la programmation concurrente, par raffinement en étapes successives. Apprentissage de techniques de base pour la réalisation de programmes de grande fiabilité et sécurité fonctionnelle.
Objectifs spécifiques	<p>Comprendre les méthodologies de développement de la programmation orientée objet. Former la capacité d'extraction, d'identification et de spécification des exigences. Former la capacité de conception orientée objet des programmes. Acquérir des connaissances concernant l'identification des étapes de programmation. Appliquer correctement les procédés et méthodes d'implémentation des programmes. Acquérir des connaissances et des compétences dans l'utilisation des objets pour différents domaines. Appliquer correctement les procédés de gestion de projet, amélioration du processus de développement.</p>

7. Contenu de la discipline

Thématique des activités didactiques	Nombre d'heures
T1 Technologies Java : Java, présentation. Interfaces Java. Éléments de base du langage Java.	2
T2. Objets en Java. Relations entre objets. Initialisation des objets. Composition. Héritage. Héritage multi-niveaux.	6
T3. Polymorphisme en Java. Redéfinition et surcharge des méthodes.	2
T4. Classes incluses. Masquage et encapsulation des données. Classes abstraites. Interfaces.	4

T5. Exceptions et leur manipulation. Traitement des exceptions, Lancement des exceptions. Hiérarchie des classes décrivant les exceptions, Exceptions spéciales. Avantages concernant le traitement des exceptions.	2
T6. Interfaces graphiques en Java. Interfaces graphiques utilisateur. Étapes de conception des interfaces utilisateur. Composants et packages de la bibliothèque Swing et JavaFX.	8
T7. Flux de données et opérations E/S. Collections d'objets. Tableaux. Conteneurs. Itérateurs. Collections et listes. Ensembles. Livres de données.	6
Total cours:	30

Thématique des activités didactiques	Nombre d'heures
LL1. Classes, constructeurs, objets.	4
LL2. Héritage et composition.	4
LL3. Surcharge et redéfinition des méthodes en Java.	4
LL4. Formes de polymorphisme.	4
LL5. Utilisation et création d'exceptions.	4
LL6. Interfaces graphiques	4
LL7. Collections d'objets.	6
Total travaux de laboratoire:	30

8. Références bibliographiques

Principales	1. Cristian Frasinescu Curs practic de Java. Curs electronic. 2011.
Supplémentaires	<ol style="list-style-type: none"> 1. David Flanagan <i>Java in a nutt-shell</i>. O'reilly 1997. 2. Doug Lea <i>Concurrent programming in Java</i>. Addison-Weslez, 1998. 3. Paul Hzde <i>Java Threading Programming</i>. SAMS, 2001. 4. Bruce Eckel <i>Thinking in Java and Enterprise Java</i> (Free web: www.BruceEckel.com) 5. Sun-Microsystems (java.sun.com).

9. Utilisation de l'IA générative

Permission d'utilisation	<p>L'utilisation de l'IA générative dans le cadre des thèmes et projets est autorisée, à condition que les étudiants respectent les règles suivantes:</p> <ul style="list-style-type: none"> • L'IA générative peut être utilisée pour générer des idées, des structures de texte ou de code, mais tous les matériaux générés doivent être révisés et ajustés par l'étudiant pour s'assurer qu'ils correspondent aux exigences académiques. • Toute utilisation de l'IA générative doit être déclarée dans la section appendice de chaque travail, en utilisant la phrase : "Durant la préparation de ce travail, l'auteur a utilisé [NOM DE L'OUTIL / SERVICE] dans le but de [RAISON]. Après l'utilisation de cet outil/service, l'auteur a révisé et édité le contenu selon les besoins et assume l'entière responsabilité du contenu du travail."
Restrictions d'utilisation	<p>Les étudiants ne doivent pas considérer l'IA générative comme une source fiable d'informations, car elle n'offre pas de références claires ou de sources documentées.</p> <ul style="list-style-type: none"> • Il n'est pas permis de citer directement le contenu généré par l'IA dans les travaux académiques comme s'il s'agissait d'une source primaire. • Les activités dans lesquelles l'utilisation de l'IA générative est interdite sont spécifiées par le professeur et sont généralement des évaluations intermédiaires et finales ou qui ne supposent pas d'activités de développement des compétences professionnelles.

10. Évaluation

Forme d'enseignement	Périodique		Courante	Travail individuel	Examen final
	Attestation 1	Attestation 2			
Éducation à temps plein	15%	15%	15%	15%	40%
Norme de performance minimale					
Prezența și activitatea la prelegeri și lucrări de laborator Obtention de la note minimale de "5" à chacune des évaluations et travaux de laboratoire					