

## PROGRAMMATION PILOTÉE DES ÉVÉNEMENTS

### 1. Informations sur l'unité de cours / module

<b>Faculté</b>	Ordinateurs, Informatique et Microélectronique				
<b>Chaire/département</b>	Filière Francophone Informatique, dép. Génie Logiciel et Automatique				
<b>Cycle d'études</b>	Études supérieures, Licence - cycle I				
<b>Programme d'études</b>	526.2 Technologies de l'information				
<b>Année d'étude</b>	<b>Semestre</b>	<b>Type d'évaluation</b>	<b>Catégorie formative</b>	<b>Catégorie d'option</b>	<b>Crédites ECTS</b>
II (enseignement à plein temps);	4	E	S – Unité de cours de spécialité	O - unité de cours obligatoire	4

### 2. Estimation du temps total

Nombre total d'heures dans le programme	Din care				
	Heures dans la salle de cours		Travail individuel		
	Cours	Travaux pratique/dirigés	Projet d'année	Étude du matériel théorique	Préparation de l'application
120	30	30	0	30	30

### 3. Prérequis pour l'accès à l'unité de cours/module

Selon le programme d'études	Afin d'atteindre les objectifs du cours, les étudiants doivent avoir les compétences développées en mathématiques discrètes, théorie de la probabilité et de l'information, de la programmation informatique, de la programmation en C ++, des structures de données et des algorithmes, des langages formelles et des compilateurs.
Selon les compétences	Connaissance générale de la structure de l'ordinateur, des structures de données et de la structure algorithmique des programmes, des modes de fonctionnement des dispositifs numériques.

### 4. Conditions de déploiement le processus éducatif pour

Cours	Pour présenter le matériel théorique en classe, sont nécessaires un projecteur et un ordinateur. Les retards des étudiants et les appels téléphoniques pendant le cours ne seront pas tolérés.
Travaux pratique/dirigés	Les étudiants rédigeront des rapports selon les conditions formulées dans les indications méthodiques. La durée de la présentation du travail pratique est de 2 semaines après l'achèvement.

### 5. Compétences spécifiques accumulées

Compétences professionnelles	<b>C3 Compétences concernant les technologies d'applications</b> C3.1 Identifier et définir les concepts, les processus et les méthodes de traitement de l'information utilisés dans la réalisation des applications découlant des besoins de l'activité humaine; C3.2 Explication des bonnes technologies pour développer les applications nécessaires
------------------------------	---

	<p>dans les activités des organisations;</p> <p>C3.3 Utilisation de technologies modernes dans la définition d'applications logicielles;</p> <p>C3.4 Utilisation des critères et méthodes de la technologie d'application pour évaluer le respect des normes d'interopérabilité;</p> <p>C3.5 Développement d'applications logicielles utilisant des technologies modernes pour transmettre, stocker et traiter des données en fonction des besoins d'une organisation.</p> <p><b>C4 Compétences concernant les méthodes et les technologies de développement de logiciels</b></p> <p>C4.1 Identifier et définir des concepts et des méthodes axés sur le développement, la mise en œuvre et l'utilisation de logiciels;</p> <p>C4.2 Expliquer les concepts et les méthodes utilisés pour développer, mettre en œuvre et utiliser le logiciel;</p> <p>C4.3 Application des langages de programmation, des environnements de modélisation et de développement, des méthodologies pour la création de logiciels;</p> <p>C4.4 Utilisation de critères et de méthodes pour évaluer le processus de conception des systèmes en termes de qualité et de performance;</p> <p>C4.5 Développement et mise en œuvre de logiciels pour des problèmes concrets dans divers domaines de l'activité humaine.</p>
--	---

## 6. Objectifs de l'unité de cours / module

<b>Objectif général</b>	Introduction au paradigme de programmation pilotée par des événements.
<b>Objectifs spécifiques</b>	<ol style="list-style-type: none"> <li>1. Apprendre les principes, termes, outils et principales technologies pour la création de logiciels dans l'environnement de programmation événementielle (fondé sur les événements);</li> <li>2. Acquérir des compétences pour la mise en œuvre d'outils de programmation visuelle dans l'environnement intégré de développement et donc élargir les connaissances et les compétences des étudiants pour la communauté informatique;</li> <li>3. Développer les compétences de réflexion algorithmique, construites avec des compétences pour créer des applications pilotées d'événements avec une interface graphique simplifiée;</li> <li>4. Encourager les étudiants à rechercher et à appliquer la créativité à partir des compétences acquises pour développer des projets et créer des interfaces graphiques basées sur fenêtres pour résoudre des problèmes pratiques réels ;</li> <li>5. Développer des connaissances et des compétences pour analyser et comparer différents points de vue et résoudre des problèmes dans différentes variantes;</li> <li>6. Développement de compétences de travail en équipe et collaboration dans le développement de projets.</li> </ol>

## 7. Contenu de l'unité de cours / module

Thématique des activités didactiques	Nombre d'heures	
	enseignement à temps plein	enseignement à temps partiel
<b>Thème des cours</b>		
<b>T1.</b> Introduction. But et objectifs du cours. Exigences et besoins. Plate-forme MS.Net. Environnement interactifs visuels pour développer des logiciels. Principaux concepts de la programmation événementielle. Brève histoire. L'interface utilisateur graphique (GUI). Concepts et base de GUI. Consistance concernant l'interface utilisateur. L'avantage du multitâche. Gestion de la mémoire. Interface graphique indépendante du périphérique. Appels de fonction.	4	

<p>Programmation orientée objet. Architecture basée sur les messages. Le point d'entrée dans le programme. Procédure de fenêtre.</p>		
<p><b>T2.</b> Programme "Hello, world!" et programme HelloWin. Fonctions WinMain et WndProc. Appels de fonctions. Identifiants avec Caps. Nouveaux types de données. Variables handle. Notation hongroise. Point d'entrée en programme. Enregistrement de la classe fenêtre. Création d'une fenêtre. Affichage de fenêtre. Repenser la zone client. Cycle de messages. Structure de type MSG. Type de données POINT. Fonctions GetMessage, TranslateMessage, DispatchMessage. Procédure de fenêtre - fonction WndProc. Traitement de messages. Messages WM_CREATE, WM_PAINT et WM_DESTROY. Lecture d'un fichier sonore.</p>	2	
<p><b>T3.</b> Affichage de texte. Zone client. Dessiner et redessiner. Le message WM_PAINT. Rectangles valides et non valides. L'interface GDI. Fonction TextOut. Le contexte de l'appareil. Obtenir une variable handle du contexte de l'appareil. Type PAINTSTRUT. Fonction TextOut. Méthodes de cartographie. MM_TEXT. Régions de découpage. Système de police SYSTEM_FONT. Dimensions d'un caractère. Fonction GetTextMetrics . Type TEXTMETRIC. Formatage du texte. Fonction GetSystemMetrics. Paramètre PitchAndFamily. Fonction SetTextAlign. La taille de la zone client. Macroinstructions LOWORD et HIWORD. Barres de défilement. La portée et la position d'une barre de défilement. SetScrollRange (hwnd, iBar, iMin, iMax, bRedraw), SetScrollPos (hwnd, iBar, iPos, bRedraw). Messages des barres de défilement. Structurer le programme pour dessiner.</p>	2	
<p><b>T4.</b> Éléments de base des graphiques. Interface pour dispositifs graphique (GDI - Graphics Device Interface). La philosophie GDI. La structure GDI. Types d'appels de fonction. Primitives GDI. Contexte du périphérique. Obtention de la variable handle du DC. Obtenir des informations DC. Fonction GetDeviceCaps. Obtenir des informations sur les dimensions de l'appareil. Obtenir des informations sur couleurs. Type COLORREF RGB (r, g, b). Attributs de contexte de périphérique. Sauvegarde du contexte de l'appareil. Dessin de lignes. Rectangles de cadrage. Courbe Bézier. Fonctions SetBkColor, SetBkMode. Modes de dessin. Dessiner des surfaces pleines. Modes de mappage. Coordonnées de l'appareil et coordonnées logiques. Viseur et fenêtre. Modes de mappage "métrique".</p>	4	
<p><b>T5.</b> Le clavier - éléments de base. Ignorer le clavier. Curseur d'entrée. Actions de clé et de caractère. Messages concernant le clavier: actions du clavier et "caractères". Touches communs et touches du système. WM_KEYDOWN, WM_SYSKEYDOWN, WM_KEYUP, WM_SYSKEYUP. Variable IParam. Codes des touches virtuelles. Etat des touches de modification. Gestion des messages d'action des touches. Ajouter la logique de traitement du message WM_KEYDOWN. Renvoi des messages. Messages de type caractère. Message WM_CHAR. Messages pour "caractères morts". Examen des messages du clavier. Curseur d'édition (CARET). Fonctions pour le curseur l'édition. Set de caractères ANSI. OEM, ANSI et les polices.</p>	4	
<p><b>T6.</b> La souris - principes de base. Fonction GetSystemMetrics. Paramètre SM_SWAPBUTTON. Zone sensible du pointeur de la souris. Messages dans la zone client BUTTONDOWN, UP et DBLCLK. Messages non-clients. Les masques MK_LBUTTON, MK_MBUTTON, MK_RBUTTON, MK_SHIFT, MK_CONTROL. Opérations de traitement des messages de la souris. Messages générés en dehors de la zone client. Le message de test de position. Les messages génèrent des messages. Vérifiez la position. Programme CHECKER. Ajout d'une interface avec clavier au programme CHECKER. Utilisez les fenêtres descendantes pour vérifier la position. Fenêtres descendantes dans le programme CHECKER. Captage de la souris.</p>	2	

<b>Tema 7.</b> Chronomètre. Bases de base sur les minuteriers. Système et minuterie. Utilisation du chronomètre: trois méthodes. Première méthode. Que faire si nous n'avons pas de chronomètre disponible? La deuxième méthode. La troisième méthode. Utilisation de la minuterie pour mettre en œuvre une horloge. Positionner et dimensionner des fenêtres contextuelles. Obtenir l'heure et la date. Internationalisation. Implémenter une horloge analogique. Animations.	2	
<b>Tema 8.</b> Contrôles de type de fenêtre descendante. Classe <i>bouton</i> . Créer des fenêtres descentes. Boîtes de validation. Boutons <i>radio</i> . Boîtes de groupe. Changer le texte d'un bouton. Boutons visibles et boutons activés. Boutons et le curseur d'entrée. Contrôles et couleurs. Couleurs du système. Couleurs des boutons. Message CTLCOLORBTN. Boutons créés par le propriétaire. La classe statique. Classe de barre de défilement. Interface automatisée du clavier. Sous-classification des fenêtres. Colorier l'arrière-plan. Coloriage des barres de défilement et du texte statique. La classe <i>édit</i> . Styles de classe <i>édit</i> . Utilisation des contrôles d'édition. Messages à un contrôle d'édition. Classe <i>listbox</i> . Styles de cassettes liste. Entrez les chaînes de caractères dans les zones de liste. Sélection et récupération d'entrées. Réception de messages à partir des zones de liste. Liste des fichiers. Utilisation des codes pour les attributs de fichier. Ordonner la liste des fichiers.	4	
<b>Tema 9.</b> Ressources. Icônes, curseurs, images bitmap et chaînes de caractères. Compilation des ressources. Pictogrammes et curseurs. Éditeur d'image. Utilisation d'icônes dans les programmes. Utilisation de curseurs alternatifs. Images bitmap: peintures pixelisés. Utilisation d'image bitmap et du pinceau. Personnages. Utiliser des ressources pour les chaînes de caractères. Utilisez des chaînes de caractères avec la fonction MessageBox. Ressources définies par l'utilisateur.	2	
<b>Tema 10.</b> Menus et touches d'accélération. Structure des menus. Modèle des menus. Référencer un menu en programme. Menus et messages. Comment utiliser les menus. Menus déroulants "flottants". Utilisation du menu système. Modification du menu. Autres commandes de menu. Utilisation des images bitmap dans les menus. Création d'images bitmap pour les menus. Contexte du périphérique en mémoire. Création d'une image bitmap avec du texte. Mise à l'échelle des images bitmap. Ajout d'une interface avec le clavier. Touches d'accélération. Règles concernant l'affectation des touches d'accélération. Table des accélérateurs. Téléchargement des tables d'accélérateurs. Conversion des messages du clavier. Réception des messages d'accélération. Le programme POPPAD avec le menu et des accélérateurs. Activation des éléments du menu. Traitement des options de menu.	4	
Total des cours:	<b>30</b>	
<b>Thèmes des travaux pratiques</b>		
<b>TP1.</b> Présentation de la plate-forme MS.Net et le choix d'un environnement visuel pour le développement de logiciels.	4	
<b>TP2.</b> La fenêtre principale d'une application.	4	
<b>TP3.</b> Interface des périphériques graphiques.	4	
<b>TP4.</b> Lignes, surfaces pleines et courbes de Bézier.	4	
<b>TP5.</b> Clavier, souris et minuterie.	2	
<b>TP6.</b> Fenêtres descendants: boutons, boîtes de validation, boîtes de dialogue.	2	
<b>TP7.</b> Programmation des TP de 1 à 6 en langage C #.	10	
<b>Total des travaux pratiques:</b>	<b>30</b>	

## 8. Références bibliographiques

Principales	<ol style="list-style-type: none"> <li>1. Charles Petzold. Programing Windows. Fifth Edition. The definitive guide to the Win32 API. Microsoft Press, 2002, 1105 pp.</li> <li>2. Charles Petzold. Programarea în Windows 95, - Editura Teora, București, 1996, 1062 pp.</li> </ol>
-------------	--

	<ol style="list-style-type: none"> <li>3. Ciclu de prelegeri la disciplina “Programarea în Windows”, Varianta de calculator. UTM, bloc 3, 506</li> <li>4. V.Beșliu ș.a. Îndrumar metodic pentru lucrări de laborator la disciplina “Programarea în Windows”, UTM, 2006</li> <li>5. Doru Turturea. Programarea aplicațiilor Windows în limbajul C. Editura Tehnică, București, 1995, 540 pp.</li> <li>6. Peter Norton, Paul Yao. Windows 3.1 Advanced Programming Techniques. – Sybex, 1992</li> <li>7. Florica Moldoveanu, Gabriel Hera. Programarea aplicațiilor Windows, - Editura Teora, București, 1994</li> </ol>
Supplémentaires	<ol style="list-style-type: none"> <li>1. Stephen Ferg. Event-Driven Programming: Introduction, Tutorial, History. <a href="http://Tutorial_EventDrivenProgramming.sourceforge.net">http://Tutorial_EventDrivenProgramming.sourceforge.net</a> (08.07.2017)</li> <li>2. Bertrand Meyer. TOUCH OF CLASS. Learning to Program Well with Objects and Contracts. Springer, <a href="http://touch.ethz.ch/">http://touch.ethz.ch/</a> (08.07.2017)</li> </ol>

### 9. Evaluation

Actuelle		Projet d’année	Examen final
Attestation 1	Attestation 1		
30%	30%	-	40%
Normes de rendement minimum			
Présence et activité aux cours et travaux pratiques; Obtenez le score minimal "5" pour chacune des attestations et des travaux pratiques; Démonstration a l'examen final des connaissances théoriques et pratiques qui ont été enseignées pendant le semestre.			