

Domaine : **Sciences et Technologies**

Parcours : **Génie Logiciel (GL), Maintenance et Réseaux Informatiques (MRI)**

Etablissement : **Centre Informatique et de Calcul (CIC)**

**Code et Intitulé de l'UE : INF351 Programmation en langage C++**

Crédits : **trois (3)**

Public cible : **Tous les étudiants qui désirent connaître les techniques de la POO (Programmation Orientée Objet).**

Semestre pédagogique : **Quatrième (4<sup>e</sup>)**

Semestre académique : **Mousson**

Prérequis : **Programmation en langage C++ et Algorithmique Objet.**

Enseignant responsable de l'UE : **AKAKPO Folly, Assistant délégué,  
Ingénieur Informaticien en GL, fakakpo@univ-lome.tg**

Disponibilité : **Mercredi de 7h30 à 10h30**

## 2. DESCRIPTION DE L'UNITE D'ENSEIGNEMENT

### 2.1 OBJECTIFS DE L'UNTE D'ENSEIGNEMENT

#### Objectif général :

Apprendre et pratiquer la programmation orientée objet (POO) en s'appuyant sur le langage **C++**.

#### Objectifs spécifiques :

À l'issue des séances de cette UE, les participants assidus devront être capables de :

- Expliquer le paradigme de la programmation Orientée Objet ;
- Expliquer les avantages de la POO par rapport à la **programmation** procédurale ;
- Employer le vocabulaire de la POO ;
- Expliquer les propriétés de base de l'OO ;
- Écrire des programmes au moyen du langage C++ ;
- S'adapter à tout autre langage de programmation fondé sur l'OO
- Écrire des programmes répondant à des cahiers des charges ;
- Etc.

### 2.2 CONTENU DE L'UNITE D'ENSEIGNEMENT

Bref descriptif de l'UE : (max. 10 lignes)

Cette unité d'enseignement est importante dans les parcours de formation en informatique comme le génie logiciel et la maintenance & réseaux informatiques.

En effet, la POO est l'un des paradigmes sur lequel sont basés plusieurs langages de programmation modernes. Dès lors, tout informaticien véritable est encouragé à se familiariser avec les bases de la POO.

Le choix du langage C++ est stratégique en ce sens qu'il est prisé pour la rapidité de ses applications. C'est le cas dans le domaine des jeux vidéo.

**Plan du contenu d'enseignement** (parties, chapitres et sous-chapitres)

<b>Séance n°</b>	<b>Rappel des objectifs spécifiques</b>	<b>Titres des parties/ chapitres / sous-chapitres</b>
<b>1</b>	Découvrir les généralités de la programmation en C++	<b>Les généralités en C++</b>
<b>2</b>	Approfondir et découvrir les spécificités des tableaux en C++	<b>Les tableaux</b>
<b>3</b>	Approfondir et découvrir les spécificités des pointeurs en C++	<b>Les pointeurs</b>
<b>4</b>	Approfondir et découvrir les spécificités des fonctions en C++	<b>Les fonctions</b>
<b>5</b>	Approfondir et découvrir les spécificités du type enregistrement et des fichiers en C++	<b>Le type enregistrement (struct) Les fichiers</b>
<b>6</b>	Découvrir les bases de l'OO en C++	<b>Le vocabulaire et les principes de l'OO</b>
<b>7</b>	Appréhender et savoir pratiquer l'héritage et le polymorphisme	<b>L'héritage Le polymorphisme</b>
<b>8</b>	Savoir programmer des éléments graphiques (widgets)	<b>Les généralités du GUI</b>
<b>9</b>	Se familiariser avec la programmation graphique	<b>La pratique du GUI</b>
<b>10</b>	Savoir combiner les différents layouts dans une fenêtre	<b>Les différents layouts et leur combinaison dans une fenêtre</b>
<b>11</b>	Savoir concevoir vos fenêtres visuellement	<b>Modéliser ses fenêtres avec Qt Designer</b>
<b>12</b>	Réaliser un petit projet Qt attaquant une base de données MySQL	<b>Mini-projet de synthèse en Qt</b>

**Modalités d'évaluation :** (période, type d'activité, organisation, etc.)

- Devoirs de maison
- Travaux dirigés
- Travaux pratiques
- Examen final

**Bibliographie :** (les ouvrages et sites internet que les étudiants doivent utiliser/consulter pour maîtriser les objectifs de cette UE)

Le grand livre du langage C++, Micro application, Edition 1998

Page web : <http://www.siteduzero.com/tutoriel-3-11406-programmez-avec-le-langage-c.html>

Page web : [www.commentcamarche.net/contents/cpp/index.html](http://www.commentcamarche.net/contents/cpp/index.html)

---

### 3. DEVELOPPEMENT DU CONTENU ET ACTIVITES D'APPRENTISSAGE

#### Séance n° 1 : LES GÉNÉRALITÉS EN C++

Objectif : Découvrir les généralités de la programmation en C++

Consignes / activités d'introduction (éventuellement) :

- Lire le support de cours de la page **1** à la page **47** ;
- Faire des recherches complémentaires.

Contenu :

- Historique et importance du langage C++
- Les identificateurs et leurs règles de nommage
- Les types simples et les opérateurs
- Les commandes d'entrée
- Les commandes de sortie
- Les structures de contrôle alternatives (if, switch)
- Les structures de contrôle répétitives (boucles for, do..while, while)

Activités :

Lire et faire des recherches pour être capable d'expliquer et pratiquer les éléments du contenu ci-dessus.

## Séance n° 2 : Les tableaux

Objectif : **Approfondir et découvrir les spécificités des tableaux en C++**

Consignes / activités d'introduction (éventuellement) :

- Lire le support de cours de la page **114** à la page **129** ;
- Faire des recherches complémentaires.

Contenu :

- ✓ Définition p.114-115
- ✓ Types de tableaux
- ✓ Caractéristiques
- Déclaration de tableau statique (constantes pour la taille p. 115)
  - Accès aux éléments d'un tableau p.115
  - Parcourir un tableau p.116
- Déclaration tableau dynamique p.120
  - Ajout élément (push\_back) p.121
  - Retirer élément (pop\_back) p. 122
- Les tableaux multidimensionnels p.125
  - Syntaxe
  - Initialisation
  - Accès aux éléments
- Les strings comme tableau p.129

Activités :

- Lire et faire des recherches pour être capable d'expliquer la notion de tableau
- Pratiquer et posséder l'utilisation des tableaux en C++

## Séance n° 3 : Les pointeurs

Objectif : **Approfondir et découvrir les spécificités des pointeurs en C++**

Consignes / activités d'introduction (éventuellement) :

- Lire le support de la page **149** à **163**
- Pratiquer et posséder l'utilisation des pointeurs en C++

Contenu :

- ✓ Adressage direct et adressage indirect p.149
- ✓ Définition p.148
- ✓ Déclaration et Initialisation p.151-
- ✓ Déréférencement ou indirection p.152
- Allocation dynamique p.153
  - L'opérateur **new** p.153

- Types scalaires (Individuellement)
- Types composés (Globalement)
  - L'opérateur **delete** (initialisation obligatoire) p.155-
  - Types scalaires (Individuellement)
  - Types composés (Globalement)
- ✓ Quand utiliser des pointeurs p.160
- ✓ Pointeurs et tableaux
- ✓ Tableaux de pointeurs
- ✓ Pointeurs vers pointeurs
- ✓ Pointeurs constants

Activités :

- Lire et faire des recherches pour être capable d'expliquer la notion de pointeur
- Pratiquer et posséder l'utilisation des pointeurs

**Séance n° 4 : Les fonctions**

Objectif : **Approfondir et découvrir les spécificités des fonctions en C++**

Consignes / activités d'introduction (éventuellement) :

- Lire le support de la page **88 à 113**
- Pratiquer et posséder l'utilisation des tableaux en C++

Contenu :

- ✓ Quoi p.88
- ✓ Pourquoi (but) : organisation, travail en équipe (partage et réutilisation) p.88
- ✓ Commentaire indicateur de fin p.88
- ✓ Les moments d'une fonction (déclaration, définition, appel) p.89-91
- ✓ En-tête ou signature ou prototype (type void) p.89
- ✓ Surcharge p.90
- Paramètres
  - sans paramètre p.92
  - passage par valeur p.96
  - passage par référence p.97
  - passage par référence constante p.99
  - par adresse (pointeur)
  - les valeurs par défaut p.108
- Déclaration et définition
  - ~~Définition top-down avec déclarations locales~~
  - ~~Définition top-down avec déclarations globales~~
  - ~~Définition bottom-up sans déclaration~~

- Déclarations et définitions séparées dans plusieurs fichiers p.100-
- ✓ Cartouche p.106, 107)
- ✓ Les fonctions et les tableaux p.118
- ✓ Les fonctions et les vectors p.124
- ✓ Variables locales et variables globales

Activités :

- Lire et faire des recherches pour être capable d'expliquer la notion de tableau
- Pratiquer et posséder l'utilisation des fonctions

### Séance n° 5 : Les fichiers

Objectif : **Approfondir et découvrir les spécificités des fichiers en C++**

Consignes / activités d'introduction (éventuellement) :

- Lire le support de la page **131 à 139**
- Pratiquer et posséder l'utilisation des fichiers en C++

Contenu :

- ✓ Définition
- ✓ Utilisation
- ✓ Types (texte/séquentiel et binaire/direct)
- ✓ L'entête fstream p.131
- Ouvrir un fichier en écriture p.131
  - ofstream p.131
  - Chemin (absolu ou relatif) p.130
  - La fonction c\_str() p.130
  - Test résultat ouverture .130
- ✓ Ecriture dans un fichier (flux<<...)
- ✓ Les différents modes d'ouverture p.131
- Ouverture en lecture p.132
  - ifstream
- Lecture d'un fichier
  - ofstream
  - Les trois méthodes de lecture (ligne, mot, caractère)
    - Par ligne : getline(monFlux,varLigne)
    - Par mot : monFlux>>varMot
    - Par caractère : monFlux.get(varChar)
  - Tester fin de fichier p.133
- ✓ Fermeture volontaire (forcée) : monFlux.close()
- ✓ Ouverture avec open
- ✓ Quelques fonctions utiles

Opération	Ifstream	ofstream	Paramètres
Connaître position	tellg()	tellp()	-
Se déplacer	seekg()	seekp()	ios::beg, end, cur
Connaître taille	seekg + tellg	seekp + tellp	

Activités :

- Lire et faire des recherches pour être capable d'expliquer le concept
- Pratiquer et posséder l'utilisation de ce concept

### Séance n° 6 : Le type enregistrement (struct)

Objectif : **Approfondir et découvrir les spécificités du type enregistrement en C++**

Consignes / activités d'introduction (éventuellement) :

- Lire le support spécifique
- Pratiquer et posséder l'utilisation du type struct en C++

Contenu :

- ✓ Définition
- ✓ Déclaration
- ✓ Nécessité de **typedef**
- ✓ Exemple
- ✓ Pointeurs vers structure
- ✓ Tableaux de structure

Activités :

- Lire et faire des recherches pour être capable d'expliquer le concept
- Pratiquer et posséder l'utilisation de ce concept

### Séance n° 7 : L'HÉRITAGE ET LE POLYMORPHISME

Objectif : **Appréhender et savoir pratiquer l'héritage et le polymorphisme**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports de cours donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le support principal de cours de la page **241** à la page **287** ;
- Lire et faire des recherches complémentaires pour être capable de défendre l'héritage et le polymorphisme ;
- Pratiquer et posséder l'utilisation de l'héritage et le polymorphisme en C++.

Contenu :



- L'héritage
  - Les classes et les pointeurs
  - Gestion de l'allocation dynamique (new et delete)
  - Le constructeur de copie
  - L'opérateur de copie
  - Définition de l'héritage (spécialisation et généralisation)
  - Comment reconnaître l'héritage
  - Héritage et constructeurs
  - Droits d'accès dans l'héritage : la portée de **protected**
- Le polymorphisme
  - Définition (paramétrique avec surcharge et transtypage)
  - La résolution des liens statiques (non reconnaissance des types spécifiques)
  - La résolution dynamique des liens (pointeurs/références et méthodes virtuelles)
  - Gestion des collections hétérogènes
  - Les méthodes virtuelles pures (=0)
  - Les classes abstraites
  - Attributs et méthodes statiques

Activités :

- Faire l'exercice pratique ci-dessous ;
- Partager les difficultés en groupe restreint ;
- Noter les difficultés non résolues et les signaler aux délégués en vue de leurs redirections.

## Séance n° 8 : LES GÉNÉRALITÉS DU GUI

Objectif : **Savoir programmer des éléments graphiques (widgets)**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports de cours donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le support principal de cours de la page **288** à la page **332** ;
- Lire et faire des recherches complémentaires pour être capable de défendre la programmation graphique ;
- Pratiquer et posséder l'utilisation de l'héritage et le polymorphisme en C++.

Contenu :

- Préalable : classe et héritage
- GUI (quoi, pourquoi, comment) ?

- Installation de Qt Créateur
- Création d'un projet vide (p.301)
- Ajout d'un fichier main.cpp (p.305)
- Codons la première fenêtre (p.307)
- QT += widgets dans le fichier .pro (p.310)
- Compilation + Exécution
- Exploration documentation
- Personnalisation des widgets
  - o setText()
  - o setTooltip()
  - o setFont(QFont)
  - o setCursor()
  - o setIcon()
  - o setGeometry()
  - o setFixedSize()
  - o etc.
- Qt et l'héritage en folie
- QObject : une classe de base incontournable
  - o Les classes abstraites
  - o Conteneurs et contenus

Activités :

- Faire l'exercice pratique ci-dessous ;
- Partager les difficultés en groupe restreint ;
- Noter les difficultés non résolues et les signaler aux délégués en vue de leurs redirections.

## **Séance n° 9 : LA PRATIQUE DU GUI**

Objectif : **Se familiariser avec la programmation graphique**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports de cours donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le support principal de cours de la page **333** à la page **387** ;
- Lire et faire des recherches complémentaires pour être capable de défendre la programmation graphique ;

Contenu :

- Les slots et les signaux (p. 333)
- Les boîtes de dialogues usuelles (p. 345)

Activités :

- Faire l'exercice pratique ci-dessous ;
- Partager les difficultés en groupe restreint ;
- Noter les difficultés non résolues et les signaler aux délégués en vue de leurs redirections.

### Séance n° 10 : **Les différents layouts et leur combinaison dans une fenêtre**

Objectif : **Savoir combiner les différents layouts dans une fenêtre**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports de cours donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le support principal de cours de la page **371** à la page **403** ;
- Lire et faire des recherches complémentaires pour être capable de défendre la programmation graphique.

Contenu :

- Positionner ses widgets avec les layouts verticaux et/ou horizontaux (p. 371)
- Les principaux widgets (p. 388)

Activités :

- Faire l'exercice pratique ci-dessous ;
- Partager les difficultés en groupe restreint ;
- Noter les difficultés non résolues et les signaler aux délégués en vue de leurs redirections.

### Séance n° 11 : **Modélisez ses fenêtres avec Qt Designer**

Objectif : **Savoir concevoir vos fenêtres visuellement**

Consignes / activités d'introduction (éventuellement)

- Lire le support spécifique
- Lire et faire des recherches complémentaires pour être capable de défendre la programmation graphique.

Contenu :

Activités :

Séance n° 12 : **Mini-projet de synthèse en Qt**

Objectif : **Réaliser un petit projet Qt attaquant une base de données MySQL**

Consignes / activités d'introduction (éventuellement)

Contenu :

Activités :

#### **4. ACTIVITES COMPLEMENTAIRES (éventuellement)**

Autres activités d'apprentissage (éventuellement)

#### **5. DOCUMENTS COMPLEMENTAIRES**

Documentation complémentaire d'approfondissement (éventuellement), liens utiles