

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 : INF113 Programmation en langage C

Crédits : trois (3)

Public cible : **Tous les étudiants qui désirent apprendre les bases de la programmation informatique et qui ont satisfait les prérequis.**

Semestre pédagogique : **Deuxième (2^e)**

Semestre académique : **Mousson**

Prérequis : **Architecture et Technologie des Ordinateurs (ATO), Système d'exploitation, Algorithmique.**

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

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

2. DESCRIPTION DE L'UNITE D'ENSEIGNEMENT

2.1 OBJECTIFS DE L'UNTE D'ENSEIGNEMENT

Objectif général :

Apprendre la base de la programmation informatique 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 la notion de programme dans les systèmes informatiques ;
- Écrire des programmes au moyen du langage C ;
- 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 programmation occupe une place centrale en informatique. Ainsi, il est nécessaire à tout apprenant en informatique de comprendre et maîtriser la base de la programmation.

Le choix du langage C est stratégique en ce sens que les syntaxes du langage C se retrouvent dans beaucoup d'autres langages de programmation dont la plupart sont dérivés du C.

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

Séance n°	Rappel des objectifs spécifiques	Titres des parties/ chapitres / sous-chapitres
1	Connaître la définition de certains termes de base, les notions de base, les types de base, les opérateurs et les expressions.	Les généralités
2	Savoir programmer un ordinateur pour lire des données et afficher des résultats	Les commandes d'entrée/lecture/saisie et de sortie/écriture/affichage
3	Savoir programmer des traitements conditionnels (if et switch)	Les structures de contrôle alternatives
4	Savoir programmer des traitements répétitifs (boucles for, while, do while)	Les structures de contrôle répétitives
5	Savoir programmer des traitements utilisant des séries de données (vecteur, matrice, grille, etc.)	Les tableaux
6	Savoir programmer des traitements sur des textes	Les chaînes de caractères
7	Savoir adresser indirectement la mémoire de l'ordinateur en plus de l'adressage direct.	Les pointeurs
8	Savoir programmer avec l'allocation dynamique de la mémoire	Les pointeurs (suite et fin)
9	Apprendre la notion et l'intérêt des sous-programmes ainsi savoir structurer un programme en sous-programmes	Les fonctions
10	Maîtriser le passage des paramètres des fonctions	Les fonctions (suite et fin)
11	Savoir programmer des traitements à base de collection de rubriques d'information	Les enregistrements
12	Savoir programmer des traitements utilisant des fichiers	Les fichiers séquentiels

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)

- Introduction to C programming, Tutoriel de DevC++
 - Introduction au langage de programmation C, Bernard Mans, Automne 1993
 - Page web : <http://www.commentcamarche.net/contents/c/>
 - Page web : <http://www.siteduzero.com/tutoriel-3-14189-apprenez-a-programmer-en-c.html>
-

3. DEVELOPPEMENT DU CONTENU ET ACTIVITES D'APPRENTISSAGE

SEANCE N° 1 : LES GÉNÉRALITÉS

Objectif : **Connaître la définition de certains termes de base, les notions de base, les types de base, les opérateurs et les expressions.**

Consignes / activités d'introduction (éventuellement) :

- Lire le support n°1 de cours ;
- Lire le support n°2 de cours de la page 6 à la page 18 ;
- Lire le support n°3 de cours de la page 3 à la page 47.

Contenu :

- Installation de l'environnement de développement (code::blocks)
- Prise en main de l'IDE
- Notion d'éditeur ASCII
- Notion de compilateur
- Notion de linker
- Les types simples
- La déclaration des variables simples
- Les opérateurs standard
- Les expressions et les instructions
- Les priorités des opérateurs
- Les fonctions arithmétiques standard
- Les conversions de type
- Exercices d'application

Activités :

- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées ;
- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées aux délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 2 :

LES COMMANDES D'ENTREE/LECTURE/SAISIE ET DE SORTIE/ECRITURE/AFFICHAGE

Objectif : **Savoir programmer un ordinateur pour lire des données et afficher des résultats**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire les sections 1.2.3 et 1.2.4 à la page 10 du support n°2 ;
- Lire les chapitres 3 et 4 du support principal (n°3) du cours, de la page **26** à la page **59** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Écriture formatée de données
- Lecture formatée de données
- Écriture d'un caractère
- Lecture d'un caractère
- Indentation
- Les commentaires
- Le cartouche d'un fichier programme
- Exercices d'application

Activités :

- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;

- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 3 : LES STRUCTURES DE CONTROLE ALTERNATIVES

Objectif : **Savoir programmer des traitements conditionnels (if et switch)**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 5 du support principal (n°3) du cours, de la page **60** à la page **67** ;
- Lire le second support (n°2) du cours, de la page **6** à la page **25** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- if – else
- if sans else
- if - else if - ... – else
- Les opérateurs conditionnels
- switch
- Exercices d'application

Activités :

- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 4 : LES STRUCTURES DE CONTROLE RÉPÉTITIVES

Objectif : **Savoir programmer des traitements répétitifs (boucles for, while, do while)**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;

- Lire le chapitre 6 du support principal (n°3) du cours, de la page **68** à la page **76** ;
- Lire le second support (n°2) du cours, de la page **26** à la page **29** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Boucle **for**
- Boucle **while**
- Boucle **do ... while**
- Comment choisir la boucle la plus appropriée lors de la résolution d'un problème ?

Activités :

- Faire le(s) devoir(s) de maison,
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 5 : LES TABLEAUX

Objectif : **Savoir programmer des traitements utilisant des séries de données (vecteur, matrice, grille, etc.)**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 6 du support principal (n°3) du cours, de la page **77** à la page **98** ;
- Lire le second support (n°2) du cours, de la page **29** à la page **32** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Définition
- Déclaration et mémorisation

- Initialisation et réservation automatique
- Accès aux éléments
- Affichage et affectation
- Tri des éléments dans un tableau

Activités :

- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 6 : LES CHAÎNES DE CARACTÈRES

Objectif : **Savoir programmer des traitements sur des textes**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 6 du support principal (n°3) du cours, de la page **97** à la page **115** ;
- Lire le second support (n°2) du cours, de la page **33** à la page **37** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Déclaration et mémorisation
- Les chaînes de caractères constantes
- Initialisation de chaînes de caractères
- Accès aux éléments d'une chaîne
- Précédence alphabétique et lexicographique
- Travailler avec des chaînes de caractères
- Tableaux de chaînes de caractères
- Problèmes liés à la saisie bufférisée [Support n°2, p.36 / voir aussi l'emploi de **fflush(stdin)**]
- Exercices d'application

Activités :

- Faire le(s) devoir(s) de maison

- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 7 : LES POINTEURS

Objectif : **Savoir adresser indirectement la mémoire de l'ordinateur en plus de l'adressage direct.**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 9 du support principal (n°3) du cours, de la page **116** à la page **138** ;
- Lire le second support (n°2) du cours, de la page **51** à la page **56** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- L'importance des pointeurs en C
- Adressage de variables (direct et indirect)
- Les pointeurs
 - o Les opérateurs de base
 - o Les opérations élémentaires sur pointeurs
- Pointeurs et tableaux
 - o Adressage des composantes d'un tableau
 - o Arithmétique des pointeurs
 - o Pointeurs et chaînes de caractères
 - Pointeurs sur char et chaînes de caractères constantes
 - o Pointeurs et tableaux à deux dimensions
- Tableaux de pointeurs
- Exercices d'application

Activités :

- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;

- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 8 : LES POINTEURS (SUITE ET FIN)

Objectif : **Savoir programmer avec l'allocation dynamique de la mémoire**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 9 du support principal (n°3) du cours, de la page **139** à la page **146** ;
- Lire le second support (n°2) du cours, de la page **56** à la page **59** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Tableaux de pointeurs
- L'allocation dynamique de mémoire
- Exercices d'application

Activités :

- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 9 : LES FONCTIONS

Objectif : **Apprendre la notion et l'intérêt des sous-programmes**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 10 du support principal (n°3) du cours, de la page **147** à la page **162** ;

- Lire le second support (n°2) du cours, de la page **38** à la page **45** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Modularisation de programmes
- La notion de blocs et la portée des identificateurs
- Déclaration et définition de fonctions
- Exercices d'application

Activités :

- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 10 : LES FONCTIONS (SUITE ET FIN)

Objectif : Maîtriser le passage des paramètres des fonctions

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 10 du support principal (n°3) du cours, de la page **163** à la page **194** ;
- Lire le second support (n°2) du cours, de la page **38** à la page **45** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Renvoyer un résultat
- Paramètres d'une fonction
 - Généralités

- Passage des paramètres par valeur
- Passage de l'adresse d'une variable
- Passage de l'adresse d'un tableau à une dimension
- Passage de l'adresse d'un tableau à deux dimensions
- Les modules en langage algorithmique, en Pascal et en C
- Discussion de deux problèmes
- Exercices d'application

Activités :

- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 11 : LES FICHIERS SÉQUENTIELS

Objectif : **Savoir programmer des traitements utilisant des fichiers**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le chapitre 11 du support principal (n°3) du cours, de la page **195** à la page **219** ;
- Lire le second support (n°2) du cours, de la page **62** à la page **65** ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer et pratiquer les concepts du contenu ci-dessous ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Définitions et propriétés
- La mémoire tampon
- Accès aux fichiers séquentiels
- Ouvrir et fermer des fichiers séquentiels
- Lire et écrire dans des fichiers séquentiels
- Résumé sur les fichiers
- Mise à jour d'un fichier séquentiel en C

- Exercices d'application
-

Activités :

- Faire le(s) devoir(s) de maison
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;
- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

SEANCE N° 12 : LES ENREGISTREMENTS (LE TYPE STRUCT)

Objectif : **Savoir programmer des traitements à base de collection de rubriques d'information**

Consignes / activités d'introduction (éventuellement) :

- Considérer prioritairement les supports donnés par l'enseignant ;
- Approfondir les explications précédentes de l'enseignant ;
- Lire le second support (n°2) du cours, de la page **46** à la page **49** ;
- Lire les deux (2) supports spécifiques sur le type enregistrement (struct) ;
- Lire le support spécifique sur la programmation modulaire ;
- Lire et faire des recherches complémentaires pour être capable d'expliquer la notion de **structure** en programmation ;
- Pratiquer en considérant tous les exemples et exercices des supports numéros 2 et 3 dans les plages de pages indiquées.

Contenu :

- Différence entre une structure et un tableau
- Déclaration d'une structure
- Définition d'une variable structurée
- Accès aux champs d'une variable structurée
- Utilité de **typedef**
- Tableaux de structures
- Structures et fonctions

Activités :

- Pratiquer et posséder l'utilisation des **structures** en langage C ;
- Faire le(s) devoir(s) de maison ;
- Partager les difficultés en groupe restreint avec des camarades proches ou plus éclairés ;

- Signaler les difficultés non réglées au délégués pour une redirection à l'enseignant dans de meilleurs délais.

4. ACTIVITES COMPLEMENTAIRES (éventuellement)

Autres activités d'apprentissage (éventuellement)

5. DOCUMENTS COMPLEMENTAIRES

Documentation complémentaire d'approfondissement (éventuellement), liens utiles