

# Projet tutoré 2020 – Langage C et algorithmique

IAM – RT & IG 1<sup>ère</sup> année

Durée : 1 mois

---

**Consigne :** Le projet devra être réalisé par binôme. Il débutera dès le 30 Avril et vous disposerez de 1 mois (à compter de cette date) pour rendre le travail demandé. Ledit projet fera l'objet d'une mini-soutenance de présentation des travaux. Vous serez notés non seulement par la pertinence du travail fourni mais également par la qualité de la présentation

**Objectif :** Ce projet a pour objectif de réaliser une mini application présentant l'analyse statistique de l'état d'avancement de la pandémie du Coronavirus au Burkina Faso en particuliers

Comme vous le savez, les coronavirus forment une vaste famille de virus qui peuvent être pathogènes chez l'homme et chez l'animal. On sait que, chez l'être humain, plusieurs coronavirus peuvent entraîner des infections respiratoires dont les manifestations vont du simple rhume à des maladies plus graves comme le syndrome respiratoire du Moyen-Orient (MERS) et le syndrome respiratoire aigu sévère (SRAS). Le dernier coronavirus qui a été découvert est responsable de la maladie à coronavirus 2019 (COVID-19).

Au Burkina Faso, le COVID19 a débuté le 9 Mars 2020 à Ouagadougou. De nos jours, on totalise 638 cas confirmés dont 476 guérisons et 42 décès.

A partir des données disponibles sur internet (en l'occurrence « lefaso.net » (lien [ici](#)), l'objectif de ce projet est d'une part d'élaborer un tableau récapitulant les différents cas et ceux depuis le 9 Mars jusqu'à nos jours et d'autre part de fournir des éléments statistiques

## Déroulement du projet :

Le projet se déroulera en 2 étapes :

### **Etape 1 :** L'implémentation du cœur de l'application :

Dans cette phase, il s'agira de collecter les informations sur le COVID19. Pour faciliter le projet, nous allons privilégier le site du « [lefaso.net](#) » présentant en direct l'état de la pandémie au burkina. Il s'agira donc d'élaborer un tableau (excel) présentant les données comme suit :

date	Nouveaux Cas confirmés	Nouveaux décès	Nouveaux guérisons	Total cas confirmés	Total décès	Total guérison
------	------------------------	----------------	--------------------	---------------------	-------------	----------------

28/04/2020	3	0	7	638	42	476
...						

### ***Etape 2 : Gestion des données en C:***

Dans cette étape, il s'agira de gérer le fichier des données. Il faudra donc créer 2 fichiers « fichier.h » et « fichier.c ». Ils contiendront principalement un structure et 2 fonctions :

- Une structure « daily\_data » permettant de gérer les données par jour
- « lireDonnees » prenant en paramètre le chemin du fichier de données (« chemin\_fichier ») et un tableau de structure (« tab »). La fonction charge l'ensemble des données contenues dans le fichier dans le tableau « tab ».
- « sauvegarderDonnes » prenant en paramètre le chemin du fichier de données (« chemin\_fichier ») et un tableau de structure (« tab »). Elle sauvegarde les données contenues du tableau dans le fichier

### ***Etape 3 : Analyse statistique en C:***

Cette partie est consacrée à la réalisation de fonction statistique. Il s'agira de créer 2 fichiers « statistique.h » et « statistique.c ». Ensuite d'y écrire les fonctions suivantes :

- ***float moyenne(daily\_data \* data, int n, int cas)*** : elle prend en paramètre un pointeur sur la structure « daily\_data » et sa taille « n » et un paramètre « cas ». Ce paramètre ne prend que 3 valeurs :
  - 0 : nouveaux cas confirmés
  - 1 : nouveau cas de décès
  - 2 : nouveau cas de guérison

Cette fonction calcule la moyenne des cas définit en fonction du paramètre « cas ». Par exemple, si le paramètre cas = 0 alors on calcule la moyenne des nouveaux cas confirmés.

- ***float ecartType(daily\_data \* data, int n, int cas)*** : elle prend en paramètres les mêmes éléments que la fonction précédente puis calcule et retourne l'écart-type
- ***float mediane(daily\_data \* data, int n, int cas)*** : idem à la précédente mais retourne la valeur médiane
- ***afficherStatistiques(daily\_data \* data, int n)*** : cette fonction affiche les informations statistiques (moyenne, écart-type, médiane) des 3 cas : nouveaux cas confirmés, nouveau cas de décès et nouveau cas de guérison
- ***afficherCasSpecifique(daily\_data \* data, int n, int cas, float seuil)*** : cette fonction affiche, en fonction du cas, toutes les données qui sont en dessous du

paramètre « seuil ». Par exemple, si cas = 2 et seuil = 10, on doit afficher toutes les données pour lesquelles les nouveaux cas de guérison sont supérieurs à 10.

- ***afficherInfosJour(daily\_data \* data, int n, char [] date)*** : cette fonction prend en paramètre une date puis afficher les informations liées à ce jour :
  - *Nouveau cas confirmés*
  - *Nouveaux cas de décès*
  - *Nouveaux cas de guérisons*
  - *Total cas confirmés*
  - *Total cas de décès*
  - *Total cas de guérisons*

On souhaiterait prédire le nombre de nouveaux cas confirmés dans de nouvelles villes. L'analyse des données décès ont permis de mettre en exergue une fonction permettant de prédiction (attention, il s'agit d'une fonction factice) :

$$f(t) = \left| \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2} \right|$$

Où t est le nombre de mois

Ecrire une fonction « *predictionNouveauCasConfirmer* » prenant en paramètre le nombre de et retournant le nombre de nouveaux cas

#### ***Etape 4 : Fonction principale***

Dans la fonction principale, créer un pointeur de taille 1000. Charger les données contenues dans le fichier de données. Proposer les options à l'utilisateur :

- 0 : Afficher la moyenne des cas
- 1 : Afficher l'écart type
- 2 : Afficher la valeur médiane
- 3 : Afficher les statistiques
- 4 : Afficher les informations d'une date
- 5 : Afficher les informations spécifiques
- 6 : Prédiction de nouveaux cas dans de nouvelles villes

A la fin du programme n'oubliez pas de libérer la mémoire.

Bon courage !!!