

# Apprentissage de la programmation C++

## Évaluation

### Sujet du projet

Le projet proposé est de réaliser en C++ un logiciel de "todo-list". Le programme devra donc être à même de gérer un ensemble de tâches.

Une tâche se définit par :

1. Un identifiant unique
2. Un titre
3. Une description
4. Une date de création
5. Une date de clôture
6. Un status : ouverte, fermée, en progression, ...
7. Un pourcentage d'avancement
8. Une priorité : normale, urgente, ...
9. Un ensemble de commentaires, que l'utilisateur pourra ajouter au fur et à mesure de son avancement sur une tâche
10. Un ensemble de sous-tâches nécessaire à l'exécution de la tâche courante.

Bien évidemment d'une exécution à l'autre du programme il est nécessaire que les tâches définies soient sauvegardées, vous êtes libres sur le choix de la sauvegarde, écriture dans un fichier, base de donnée, ...

Pour le fonctionnement du programme deux modes sont demandés :

1. Le mode "Command Line Interface"

Dans ce mode tout se fait via la ligne de commande (création d'une nouvelle tâche, modification d'une tâche, affichage de l'ensemble des tâches, clôture d'une tâche, ....)

Par exemple, pour créer une nouvelle tâche il faudrait quelque chose dans l'esprit suivant :

```
$$ cpp_todo_app create --title "Le titre" --description "Une description de la tâche" \
    --priority high
```

ou par exemple pour afficher l'ensemble des tâches prioritaires

```
$$ cpp_todo_app list --priority high
```

2. Le mode "Interface Graphique"

Ce second mode sera considéré comme "question bonus" pour la notation.

Il s'agit de réaliser une interface graphique simple permettant de lister, créer et modifier les tâches. Pour la réalisation de cette interface graphique vous êtes libres de la librairie graphique à utiliser, si vous n'avez pas de préférences vous êtes encouragés à utiliser FLTK ([www.fltk.org](http://www.fltk.org)).

Pour l'installation de FLTK :

- (i) sur Linux ou Windows avec WSL il vous suffit de taper `sudo apt-get install libfltk libfltk-devel` ;
- (ii) Pour les utilisateurs de Windows sans WSL il vous suffit d'installer WSL si vous avez Windows 10 et retourner au point (i), si vous avez un Windows 7 ou avant allez voir Basile à la rentrée ;
- (iii) Pour les utilisateurs Mac OS allez voir Basile à la rentrée.

## Modalité d'évaluation

Le projet est à réaliser **individuellement**.

La date finale (et ferme) de rendu du projet est le mercredi 29 janvier mais vous pouvez le rendre avant.

Le rendu se fera sous la forme d'un dépôt Git hébergé sur Github. Vous nous enverrez pour le 29/01/2020 l'url de votre dépôt.

Les critères d'évaluation du projet sont les suivants :

- Fonctionnalités : vérification que l'ensemble des fonctionnalités demandées soient opérantes (vous pouvez en ajouter de nouvelles mais pas en enlever) ;
- Qualité du code : lisibilité (code bien indenté et bien aéré, des identificateurs avec des noms explicites, ...), commentaires (utiles et bien à propos), modularité, architecture, ...
- Documentation utilisateur : votre logiciel doit comporter une documentation synthétique (avec des exemples d'utilisation) permettant à n'importe qui de l'utiliser sans avoir besoin de fouiller dans les sources.
- Jeu de tests : votre repo git doit comporter un jeu de tests couvrant l'ensemble des fonctionnalités de votre logiciel. Bien sûr ces tests doivent passer avec succès leur exécution par votre logiciel.