

RAPPORT PROJET C++: SIMULATEUR DE FONCTIONNEMENT D'UN SYSTÈME DE CARBURANT D'UN AVION

ALI AHMEDI Mycipssa – **DOUBI** Dylan

Introduction

Le but de l'application est de réaliser un système de gestion de carburant d'un avion. Elle permet aux pilotes de s'entraîner à résoudre d'éventuelles pannes pouvant intervenir durant d'un vol (vidange d'un réservoir, panne d'une pompe). Le système de notation permet à chaque pilote authentifié de consulter quand il souhaite la sauvegarde de l'historique de sa progression. L'application est réalisée en C++ et utilise la bibliothèque graphique Qt.

1) Mécanisme d'installation

Il est nécessaire d'installer la bibliothèque graphique Qt.
Pour ce faire veuillez exécuter les lignes suivantes dans un terminal:

```
-sudo apt-get install libqt4-dev  
-sudo apt-get install g++
```

Si par jamais vous rencontrez un problème (au niveau du chemin d'accès) vous devez sortir le contenu du dossier qt4 comme suit:

```
- cd/usr/include/qt4  
- mv * ..
```

2) Description du système

Le système est composé de plusieurs éléments regroupés en classes:

-Tank:

Ce sont les réservoirs qui contiennent le carburant servant à alimenter les moteurs. Ils sont au nombre de 3 à savoir Tank1, Tank2 et Tank3 et sont respectivement liés aux moteurs (voir plus bas) M1, M2 et M3.

-Pompe:

Les pompes servent à acheminer le carburant d'un tank à un moteur. Il y en a 6 (2 pour chaque Tank), p11 et p12 pour Tank1, p21 et p22 pour Tank2 et p31 et p32 pour Tank3.

-Vanne:

Il existe 2 types de vannes: les VanneMoteur (V12,V23,V13) qui relient un moteur et un tank et les VanneTank (VT12 et VT23) qui relient 2 tanks.

-Moteur:

Il s'agit d'un élément assurant le fonctionnement du système. Il doit être constamment alimenté. Il y a 3 moteurs : M1, M2 et M3.

Si un Tank se vide, le pilote a la possibilité d'ouvrir une VanneTank pour pouvoir le remplir grâce au carburant du Tank d'à côté.

Si une pompe principale tombe en panne, il faut ouvrir la pompe secondaire analogue pour maintenir l'alimentation.

Si les 2 pompes d'un même Tank tombent en panne, il y a possibilité d'ouvrir la pompe secondaire d'un autre Tank pour alimenter le moteur qui était relié au Tank précédent, il faut de ce fait ouvrir la vanne qui relie le nouveau Tank au moteur qui doit être alimenté. Le flux du carburant change donc.

3) Description de l'interface

L'interface graphique repose sur les différentes bibliothèques de Qt notamment QTGui qui contient les différents éléments graphiques et QObject qui permet de créer des objets. Ces objets réagissent à des signaux comme le clic sur un bouton. L'interface affiche les différents éléments du système et des boutons permettant d'interagir avec celui-ci pour régler les différentes pannes qui peuvent se produire durant les exercices.

Il existe 2 modes, «entraînement libre» et «exercice».

Le mode «entraînement libre» permet de créer des pannes soi-même et de les résoudre librement.

Le mode «exercice» permet de passer une série d'exercices et d'être évalué à la fin. L'utilisateur doit rentrer son pseudo avant de commencer la série.

Un fichier du même nom que le pseudo du pilote va être créé.

Ce fichier va contenir l'historique des notes obtenues par le pilote et est consultable à tout moment.

Il y a au total 5 exercices chacun valant 2 points.

Les quatre premiers exercices sont prédéfinis. Le 5ème est un exercice à pannes aléatoires. À noter que le Tank 2 est plus petit que les Tanks 1 et 3 et a donc plus de chances de se vider que les 2 autres.

4) Manuel d'utilisation de l'application

Pour compiler l'application, il faut effectuer les lignes suivantes en ligne de commande:

```
-qmake -project  
-qmake  
-make
```

Et pour enfin lancer le simulateur:

```
./Simulateur
```

Une fois exécuté, le simulateur demande le pseudo du pilote pour l'enregistrer. Ce pseudo doit être entré dans le terminal.

Par convention, les pompes:

- en marche sont en vert
- en panne sont en rouge
- en arrêt sont en noir.

Les tanks sont verts quand ils sont pleins et oranges quand ils sont vides. L'état de chaque élément du système est dynamiquement affiché dans le terminal à chaque action du pilote.

Le pilote peut uniquement ouvrir ou fermer les vannes et les pompes en mode «exercice».

En «entraînement libre», le pilote a la possibilité de créer ses propres pannes avec un bouton pour chaque panne.

Après avoir fini tous les exercices, un bouton «Afficher le résultat» apparaît et permet au pilote de connaître son résultat et de sauvegarder sa note dans son fichier personnel, ce fichier contenant l'historique des notes qu'il a obtenu.

Il peut ensuite recommencer les exercices, passer au mode entraînement libre ou quitter l'application.

Veuillez consulter le fichier README.md qui contient un listing commenté des méthodes du programme.