

**Série d'exercices #3 (Variables de condition)****Nom :****Prénom :**

Objectif

- Appliquer adéquatement la gestion des variables de condition.

**À chaque fois que vous terminez un exercice, vous m'appeler pour le faire vérifier.**

---

**Exercice #1 (avec Windows et variable de condition)**

Un procédé de production utilise deux machines pour réaliser un mélange de produit et le brassage du produit.

Vous devez, à l'aide de deux threads, simuler le procédé de production suivant:

Thread Mélange

- Ajoute 1ml à toute les secondes au mélange. (simuler la seconde avec la méthode `this_thread::sleep_for(seconds(1))` de l'entête `chrono`).
- Lorsqu'on atteint 15ml, le brassage doit commencer.
- Le thread doit continuer d'ajouter 1 ml à toutes les secondes jusqu'à atteindre 20 ml.

Thread Brassage

- Doit brasser dès qu'il y a 15 ml de solution.
- Le brassage dure 5 secondes.

Implanter ce procédé à l'aide des variables de condition.

## Exercice #2 (avec Linux, Mini-Usine et sans variable de condition)

À l'aide de la mini-usine, réalisez le procédé suivant :

Thread	Action
Monter_Ascenseur	<ul style="list-style-type: none"><li>• Monter l'ascenseur.</li><li>• Attendre 5 secondes en haut.</li><li>• Descendre l'ascenseur.</li></ul>

## Exercice #3 (avec Linux, Mini-Usine et variable de condition)

Créer une application par l'intermédiaire de thread qui permet de réaliser le procédé suivant :

- Un thread pour vérifier le type de pièce.
- Un thread pour le convoyeur.

Type de pièce	Action
Orange	Rejet immédiat et allume la lumière gauche pendant 3 secondes.
Noire	Lumière centrale s'allume, l'ascenseur monte, reste en haut pendant 3 secondes et redescend. La lumière s'éteint.
Métallique	Lumière droite s'allume. L'ascenseur monte, reste en haut 3 secondes puis pousse la pièce sur le convoyeur. Le convoyeur doit partir et s'arrêter dès que la pièce est dans le bac. L'ascenseur redescend et la lumière s'éteint.

- **Utiliser les variables de condition pour signaler au convoyeur de partir.**

## **Exercice #4 (Sous Linux, Mini-usine et variable de condition)**

Réaliser le procédé suivant :

Un thread surveille l'arrivée d'une pièce dans le bac de présence (celui après la descente du convoyeur). Dès qu'une pièce arrive dans le bac de présence, le thread signale la présence de la pièce au thread bras\_aspirateur qui devra prendre cette pièce et l'entreposer dans le cylindre avant droit. Le bras\_aspirateur doit ensuite revenir à sa position de base(en haut du bac) pour être prêt à recevoir une autre pièce. Vous devez utiliser les variables de conditions pour signaler la présence de la pièce dans le bac.

### **Modalités de remise**

- **Date de remise: Mardi 26 septembre 2017 avant 23:59**
- **Le code source ET l'exécutable de TOUS LES EXERCICES sous format compressé dans la boîte de remise Col.Net prévue à cet effet.**

Sous Linux:

Compilation:

```
g++ --std=c++11 -o nom_exécutable nom_source.cpp -pthread
```

--std=c++11            Pour indiquer à votre compilateur de prendre en charge le standard c++11

-pthread                Pour indiquer de lier de votre exécutable avec la librairie "pthread"

N'oubliez pas de faire `iopl(3)` au début de vos programmes !