

# PCO: Rapport labo 7

---

**Auteur:** Anthony Pfister, Santiago Sugranes

**Date:** 25.01.2026

**Cours:** PCO2025 lab07, HEIG

---

## Description de l'implémentation

Le salon utilise un **moniteur de Hoare**. Le moniteur de Hoare utilise des **sémaphores forts** pour implémenter les conditions variables, garantissant que les threads sont réveillés dans l'ordre où ils ont appelé `wait()`, assurant ainsi l'ordre FIFO requis.

Quatre conditions variables gèrent la synchronisation:

- `_barberWaitingForClient`: barbier endormi
  - `_clientWaitingForTurn`: ordre FIFO des clients
  - `_barberWaitingForClientAtChair`: client sur la chaise de travail
  - `_clientWaitingForHaircut`: fin de la coupe
- 

## Étapes principales

**Barbier:** Vérifie s'il y a des clients -> prend le prochain (`pickNextClient()` signale `_clientWaitingForTurn`) ou dort -> attend le client sur la chaise -> coiffe (`beautifyClient()` signale `_clientWaitingForHaircut`).

**Client:** Vérifie l'accès au salon -> trouve un siège libre -> réveille le barbier si premier client -> attend son tour sur `_clientWaitingForTurn` -> va sur la chaise de travail -> attend la fin de la coupe.

**FIFO:** Les clients attendent sur `_clientWaitingForTurn` dans l'ordre d'arrivée. Les sémaphores forts garantissent que `signal()` réveille le thread qui a attendu en premier.

**Terminaison:** `endService()` met `_isInService = false` et signale toutes les conditions pour réveiller les threads bloqués. Les clients vérifient `isInService()` et appellent `goHome()`.

---

## Méthode de vérification

Tests manuels avec différentes configurations (1-20 clients, 1-8 sièges):

- Ordre FIFO respecté: les clients sont servis dans l'ordre d'arrivée
- Le barbier dort/se réveille correctement
- Terminaison propre: tous les clients sont servis avant la fermeture