

### IC1022 - GOP - Toelichting Opdracht 3

Op voorhand was (op basis van de theorie) te verwachten dat de methoden `kook()` van Kok en `serveer()` van Ober een synchronisatieprobleem zouden opleveren dus die heb ik de optie `synchronized` gegeven.

Ook zijn de methoden `plaatsMaaltijd` & `pakMaaltijd` gesynchroniseerd gemaakt en voorzien van een lock omdat deze niet gelijktijdig mogen worden gebruikt door verschillende objecten omdat beide methodes een arrayist bewerken (die van zichzelf niet `synchronized` is).

Het grote probleem met deze opdracht is dat de verschillende objecten allemaal dezelfde verwerkingstijd in hun threads hebben. Daardoor krijg je soms vreemde verschijnselen in de output (foute volgorde van het paatsen op de uitgifte balie door de koks en het serveren door de obers).

Dit heb ik opgelost door een kleine pauze tussen het aanmaken van de drie kok objectenb en de twee ober objecten.

Verder was het nodig om in het kok object de volgorde van de melding en het plaatsen op de uitgifte balie om te wisselen: het zou logischer zijn om eerst de maaltijd op de uitgiftebalei te plaatsen en dan pas de melding te tonen maar dan krijg je het verschijnsel dat bij het uitserveren lijkt dat de eerste ober die iets van de uitgifte balie haalt een maaltijd krijgt die later op de uitgiftebalie is gezet dan de maaltijd van de volgende ober terwijl dat in werkelijkheid niet het geval is (dit heb ik gecontroleerd door tijdelijk een melding op het scherm te laten zetten bij het plaatsen op de uitgifte balie). Nu laten alle runs op mijn machine de juiste info in de juiste volgorde zien.