

Opdracht 3

Een botanische tuin is opengesteld voor het publiek en heeft twee ingangen (kassa's) en twee uitgangen (hekjes). Om veiligheidsredenen mogen maximaal 100 personen zich tegelijkertijd in de tuin bevinden. Bij de kassa's kunnen enkelingen, maar ook groepen toegelaten worden. De kassamedewerkers reserveren steeds het aantal benodigde plaatsen (1 of meer) bij het geautomatiseerde systeem. Als het licht op groen gaat kan de groep doorlopen. Als er te weinig ruimte is voor de hele groep blijft het licht op rood en wacht uit solidariteit de hele groep. Het is wel zo dat wie het eerst komt het eerst maalt, dat wil zeggen als bij de andere kassa zich een kleinere groep aandient waar nog wel ruimte voor is, dan moet die kleinere groep toch wachten. Bezoekers verlaten de tuin via een van de twee draaihekjes. Ga ervan uit dat de uitstroom van bezoekers op enig moment ongeveer evenredig is met het aantal bezoekers dat op dat moment zich in de tuin bevindt.

Maak een simulatieprogramma voor deze situatie waarbij de applicatiegebruiker de functie van beide kassamedewerkers vervult. De gebruikersinterface is hieronder gegeven. Kies een geschikte tijdschaal.

Maak gebruik van threads, gesynchroniseerde methoden, de BusyFlag klasse (of in Java 5 de klasse ReentrantLock) en andere gegeven bestanden in de map Opdr3GegevenFiles. Zie voor de opzet van het programma de laatste dia's van college 2. Met name de klassendiagrammen geven inzicht in de structuur van het programma. Er is een voorbeeldapplet beschikbaar in de map TuinAppletClassFiles.

Richt je programma zo in dat het eenvoudig uitgebreid kan worden met meer kassa's en draaihekjes. Laat zien dat je de code begrijpt door ook de gegeven code van commentaar te voorzien !

