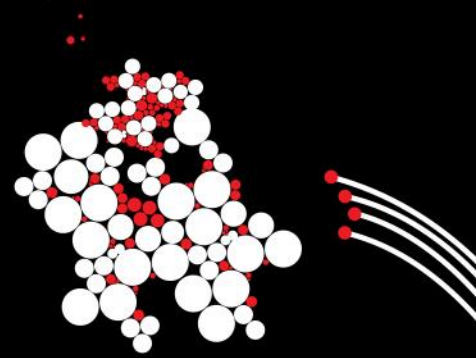


UNIVERSITY OF TWENTE.



# Synchronisation

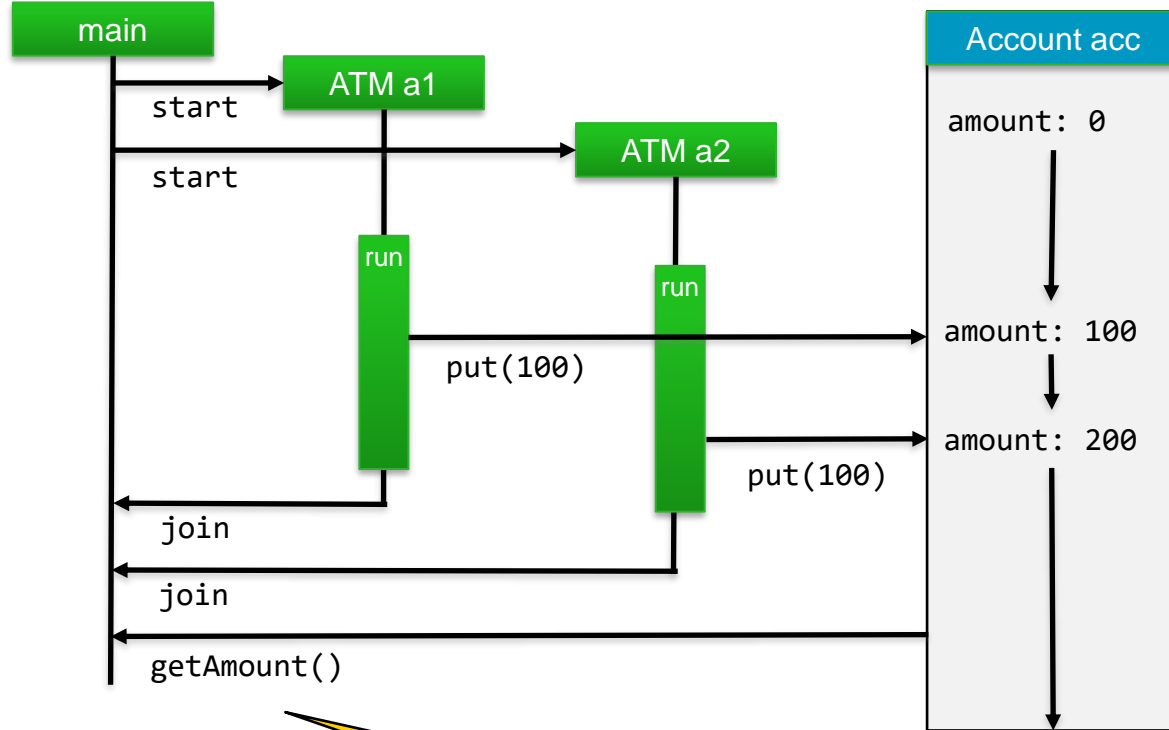
Topic of Software Systems (TCS module 2)

Lecturer: Marieke Huisman

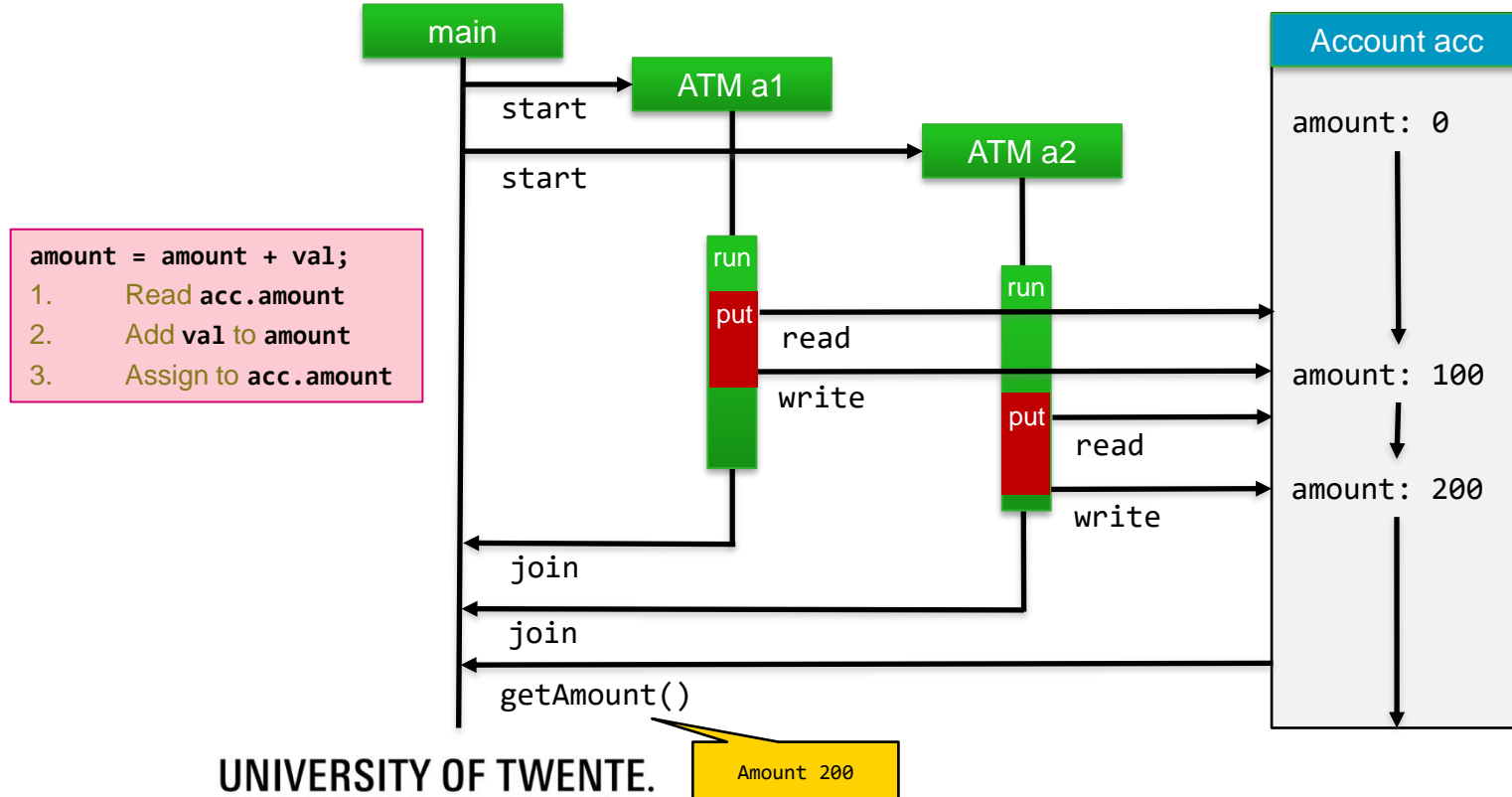


# A POSSIBLE EXECUTION

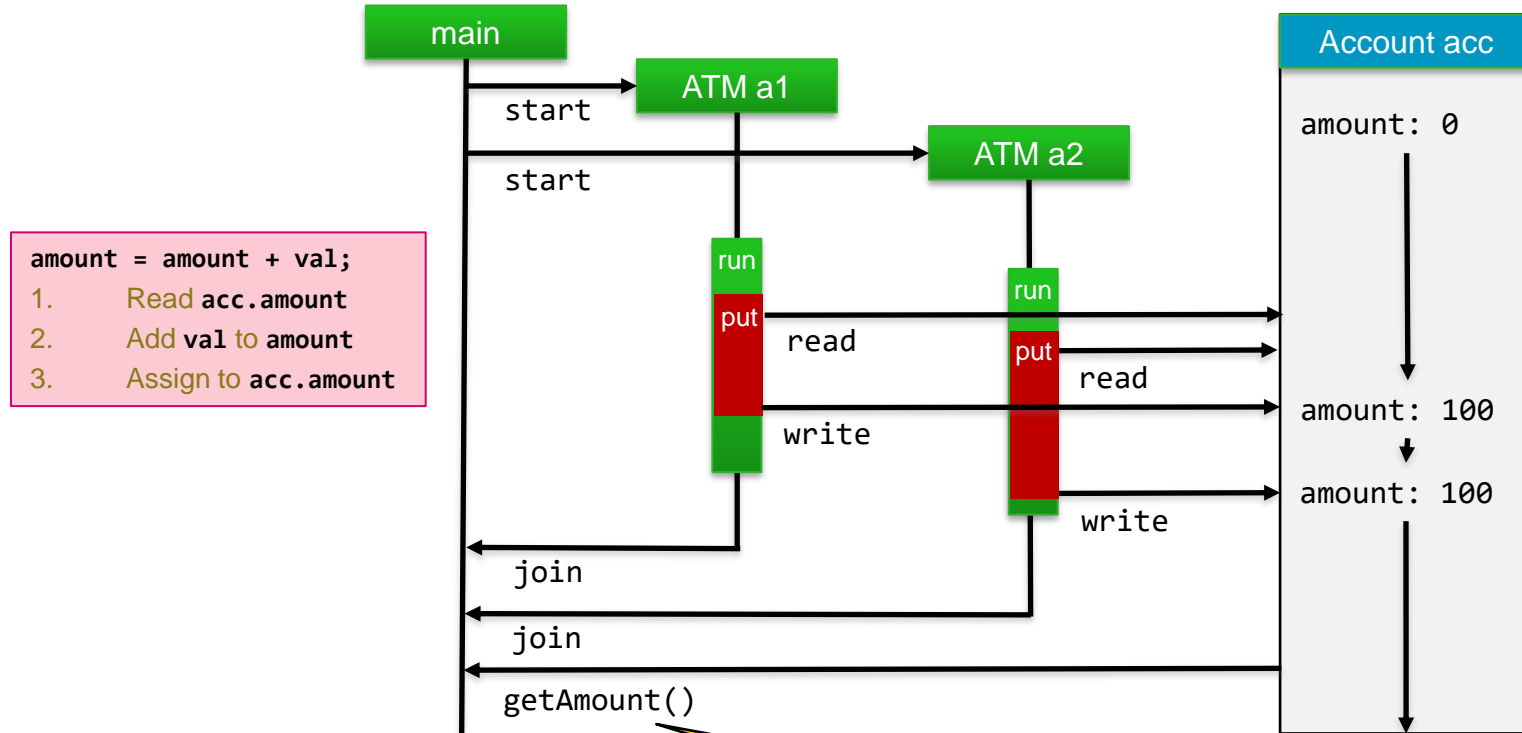
Always?  
No, not always!



# A POSSIBLE EXECUTION



# A POSSIBLE EXECUTION



# MUTUAL EXCLUSION FOR CRITICAL SECTION

---

- **Critical section** = code that **must not** be executed concurrently
  - Access to fields of shared objects
  - Use of external resources
- Example: the methods deposit and withdraw of BankAccount
- Solution: **locks** (also called **mutexes**, for **mutual exclusion**)
  - At most one thread at the time can **hold** the lock
  - **Acquire** the lock before the critical section
  - **Release** the lock after the critical section
  - Usually a thread is **blocked** until the lock becomes **available**

# A POSSIBLE EXECUTION

