# NTNU
Kunnskap for en bedre verden

TDT4165 - PROGRAMMING LANGUAGES

# Project Part II

*Authors:*
Sondre Pedersen
Emiel E.M.H Eij

November 13, 2024

# Task 3: Explaining the code

**1.)**

This is a bank system that manages accounts and processes transactions. The system must be able to handle a lot of incomming requests and process transactions in a thread safe way.

The bank keeps track of the accounts using a map datastructure, making it easy to find the account instance given the account code. The bank also has two transaction pools, one for pending and one for completed transactions. When a new transfer is requested, the bank puts it in the pending pool. As long as there are transactions to process, a dedicated thread is trying to do just that.

When done, they will be added to the pool of completed transactions.

Of course, it is important that multiple threads make changes to the same accounts, because that would cause race conditions to occur.

**2. and 3.**

It was easy to implement the Account and Transaction class, because these are essentially just dataclasses. The overall outline of the transactions was not too difficult either, because we are just adding items to a list, and updating some values.

It was mch more challenging to implement this in a thread safe way however. I think the choice of datastructure was a bad choice. It would have been much better to make a queue, and have a thread pool that could poll the queue instead. But at least it worked in the end.

**4.**

All tests passed, but I had to make a questionable decision. Test 11 failed until I made the main thread sleep for a second before processing. Without this, it would not update properly when adding transactions to the pool.

```
[info] loading settings for project bankingsystem from build.sbt ...
[info] set current project to bankingsystem (in build file:/home/sondre/projects/
[info] AccountTests:
[info] - Test 01: Account should be immutable
[info] - Test 02: Valid account withdrawal
[info] - Test 03: Invalid account withdrawal should return error message
[info] - Test 04: Withdrawal of negative amount should return error message
[info] - Test 05: Valid account deposit
[info] - Test 06: Deposit of negative amount should return an error
[info] - Test 07: Correct balance amount after several withdrawals and deposits
[info] AccountTransferTests:
[info] - Test 08: Valid transfer between accounts
[info] - Test 09: Transfer of negative amount between accounts should fail
[info] - Test 10: Invalid transfer between accounts due to insufficient funds sho
[info] - Test 11: Correct balance amounts after several transfers
[info] - Test 12: All the submitted transactions are processed
[info] Run completed in 5 seconds, 371 milliseconds.
[info] Total number of tests run: 12
```

Figure 1: Tests