# INF214 / HØST 2021 — Exam 6.12.2021

Mikhail Barash

Institutt for Informatikk, Universitetet i Bergen

## Task 1.1 – 5 points

**Describe the Critical Section Problem. What requirements do we have for the entry and exit protocols?**

## Task 1.2 – 10 points

**What is the purpose of 'synchronized' in Java? Why would one use the method 'tryLock()' of class 'ReentrantLock' in Java? What are the advantages (or disadvantages) of using atomic variables (e.g., 'AtomicInteger')?**

## Task 1.3 – 5 points

**Describe the difference between synchronous and asynchronous message passing.**

## Task 1.4 – 10 points

**Explain the main ideas, and write about similarities/differences, between actors and channels.**

## Task 1.5 – 5 points

**Explain briefly the main ideas of RPC (Remote Procedure Calls).**

## Task 2.1. "Savings Account" – 15 points

**A savings account is shared by several people (processes). Each person may deposit or withdraw funds from the account. The current balance in the account is the sum of all deposits to date minus the sum of all withdrawals to date. The balance must never become negative. A deposit never has to delay (except for mutual exclusion), but a withdrawal has to wait until there are sufficient funds. A junior software developer was asked to implement a monitor to solve this problem, using Signal-and-Continue discipline. Here is the code the junior developer has written so far:**

```
monitor Account {
   int balance = 0;
   cond cv;

   procedure deposit(int amount) {
      balance = balance + amount;
   }

   procedure withdraw(int amount) {
      balance = balance - amount;
   }
}
```

**This solution is incorrect. Help the junior developer implement the monitor correctly.**

## Task 2.2. "The Airport & Vaccination Certificates" – 20 points

**Background:**

**Consider Multilandet, which is a country, and Multibyen, which is its capital city. The airport of Multibyen has introduced strict entry requirements for arriving passengers because of a global pandemic. Some of the passengers are vaccinated, while others are not vaccinated. All passengers arriving to Multibyen must go through the document/passport control, where their vaccination certificates are checked by border guard officers.**

**Upon arriving to Multibyen Airport, passengers, both vaccinated and unvaccinated, are mingling in the Mingling Zone, and they are walking towards to the Documents Checking Zone. There, the border guard checks their vaccination certificates, imposes quarantine on the unvaccinated ones, and in any case lets all the passengers into the city.**

**Task:**

**Assume that the passengers enter the Documents Checking Zone in a random order. The only requirement is that there must be never unvaccinated and vaccinated passengers in the Documents Checking Zone at the same time. However, people with the same vaccination status are allowed in the Documents Checking Zone at the same time (that is, at any given moment of time, either all passengers in the Documents Checking Zone are vaccinated, or all passengers in the Documents Checking Zone are unvaccinated).**

**Assume that there are N vaccinated passengers, and M unvaccinated passengers who have just landed at Multibyen Airport.**

**"Simulate" the described situation in the AWAIT language.**

**Represent passengers as processes.**

**Use semaphores for synchronization.**

**Make sure that your solution avoids deadlock.**
**Your solution need NOT to be fair.**

## Task 2.3. "The Gløgg Problem" – 20 points

**Three persons, who like gløgg very much, have gathered to play the following game in a bar. To drink a portion of gløgg, each of them obviously needs three "ingredients": the gløgg (liquid in the decanter – see image), a mug, and almonds. One player has the gløgg (liquid in the decanter), the second player has mugs, and the third has almonds. Assume that each of the players has an unlimited supply of these ingredients (i.e., gløgg in the decanter, mugs, almonds), respectively.**

**The barista, who also has an unlimited supply of the ingredients, puts two random ingredients on the table. The player who has the third ingredient picks up the other two, makes the drink (i.e., pours the gløgg liquid from the decanter into a mug and adds almonds), and then drinks it. The barista waits for the player to finish. This "cycle" is then repeated.**

**"Simulate" this behaviour in the AWAIT language.**
**Represent the players and the barista as processes.**
**Use semaphores for synchronization.**
**Make sure that your solution avoids deadlock.**

## Task 3.1. – 10 points

**Using JavaScript, give an example of how ‘`yield`‘ can receive data when a coroutine is resumed.**

**Hint 1:**
**Start by declaring a generator.**
**Hint 2:**
**Think how many calls of ‘`next`‘ you need.**