UNIVERSITY
OF SKÖVDE

School of Informatics

WRITTEN EXAMINATION

Course Concurrent Programming G1F, 7.5hp

Sub-course

Course code IT404G

Credits for written examination 5hp

Date 2026-01-08

Examination time 14:15-19:30

Examination responsible: Andras Marki

Teachers concerned

Instructions

- ☐ Take a new sheet of paper for each teacher.
- ☒ Take a new sheet of paper when starting a new question.
- ☒ Write only on one side of the paper.
- ☒ Write your name and personal ID No. on all pages you hand in.
- ☒ Use page numbering.
- ☒ Don´t use a red pen.
- ☒ Mark answered questions with a cross on the cover sheet.

Examination results should be made public within 18 working days

Good luck!

Total number of pages: 5

Parallella processer G1F, IT404G. 2026-01-08

Grading

The five main questions on the written exam correspond to the course objectives. Each main question comprises three sub-questions, which are graded pass or fail. To pass the exam, you need to ***pass at least one sub-question for each of the main questions****. The more sub-questions you pass, the higher your grade will be. The detailed grading scheme is published in Canvas. For your convenience, each section lists the relevant examination criterion.*

Main question 1

Examination criterion: *Redogöra för olika frågor som måste hanteras i program med samtidigt exekverande processer, inklusive tävlan om resurser och ömsesidig uteslutning*

Sub-question 1a

Does the following code fulfil all three safety properties: mutual exclusion, absence of deadlock and absence of unnecessary delay? You should assume that the semaphore is a blocking FIFO-semaphore.

Sem s(1) Int counter=0		
Process 1	Process 2	Process 3

Int x=0;
While true{
P(s)
counter++
x=counter
Print(x)
V(s)
}
	Int x=0;
	While true{
	P(s)
	x=counter+2
	Counter=x
	Print(x)
	V(s)
	}
		Int x=0;
		While true{
		P(s)
		counter--
		counter/=2
		x=counter
		V(s)
		}

Sub-question 1b

Define the two properties, liveness and absence of unnecessary delay, in your own words. Ensure that your answer is sufficiently detailed to distinguish between the two properties.

Sub-question 1c

Many synchronisation mechanisms make use of queues to manage waiting threads. FIFO queues are the most common, but implementations may use another queue type, such as a priority queue (in which each thread specifies its priority). Discuss the advantages and risks of priority queues.

Parallella processer G1F, IT404G. 2026-01-08

Main question 2

Examination criterion: *Identifiera, beskriva och diskutera klassiska synkroniseringsproblem mellan parallella processer såsom synkronisering av läsare och skrivare eller av producenter och konsumenter*

Sub-question 2a

Please identify and describe the classic problem seen in the following description. Please identify the different processes, what each process does, and the synchronisation and communication mechanisms involved. For the mechanism you choose, please explain how it will work and why you chose it.

At the university pub students turn up and order drinks. The bartender(s) take(s) the order and produces the drink (or cocktail from appropriate ingredients) before giving it back to the student. The bartender might have to tell the student that the drink cannot be given, because the ingredients are not available.

Sub-question 2b

Please identify and describe the **classic problem** seen in the following code. Please identify what each process is doing. Please identify any issues you observe. Message passing is asynchronous.

channel c1, c2;				
Process 1	Process 2	Process 3	Process 4	Process 5

While True{
m=make_next()
c1.send(m)
}
	While True{
	c1.receive(n)
	transform(n)
	c2.send(n)
	}
		While True{
		c1.receive(o)
		transform(o)
		c2.send(o)
		}
			While True{
			c2.receive(p)
			use(p)
			}
				While True{
				c2.receive(p)
				use(p)
				}

Sub-question 2c

Describe the rules of the critical section problem. Describe what can happen if the rules are not obeyed, and give an example of a real-world problem where critical sections are necessary.

Parallella processer G1F, IT404G. 2026-01-08

Main question 3

Examination criterion: *Beskriva för- och nackdelar med olika tekniker för att lösa synkroniseringsproblem, inklusive semaforer, monitorer och tekniker för meddelandeöverföring*

Sub-question 3a

Describe the advantages and disadvantages of stateful vs stateless communication in the client-server architecture. Please give examples to illustrate your answer.

Sub-question 3b

RPC and Rendezvous are two communication mechanisms that are well-suited to client-server architectures. Describe and discuss the advantages and disadvantages of these two approaches in the context of an airline booking service.

Sub-question 3c

Monitors provide implicit mutual exclusion when processes access them. Semaphores can also provide mutual exclusion. Describe these two and discuss the advantages and disadvantages of each.

Parallella processer G1F, IT404G. 2026-01-08

Main question 4

Examination criterion: *Använda grundläggande tekniker såsom semaforer och meddelandeöverföring för att lösa synkronisering och kommunikation i program med parallella processer*

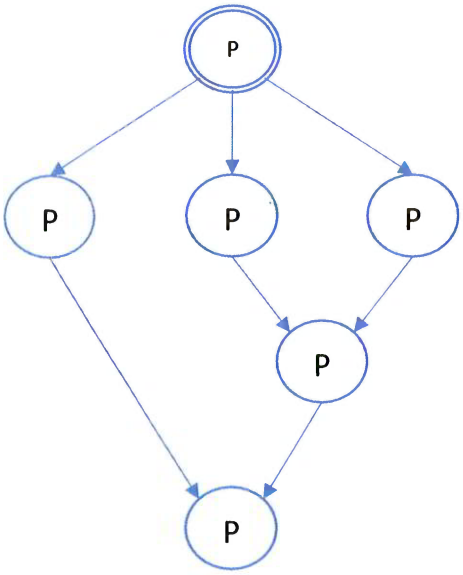
Sub-question 4a

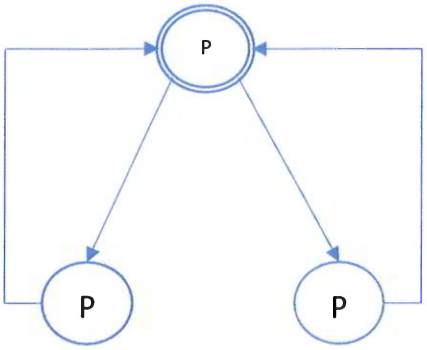
There is a system with 3 processes and 2 printers. There is also a print queue process. All the processes communicate over asynchronous channels.

Each process sends jobs to the printers via the print queue. It does not matter to the process which printer does the actual job. Each process wants to be told when its job starts and when it is complete. Write pseudo code, or a flow chart for the print queue process and state how many channels are needed. yes

Sub-question 4b

What follows is a diagram of a set of processing nodes. Synchronisation will be achieved using semaphores. Please state the number of semaphores required, their initial states and write pseudocode for each process. Please note that the arrows indicate that the later process should not begin until all earlier connected processes have completed. Nodes that can be assumed to be started positions are double ringed.





Sub-question 4c

Implement in pseudo code a controller for the dining philosopher's problem using a monitor. There are 5 philosophers. The Philosophers all follow the following 2 example patterns;

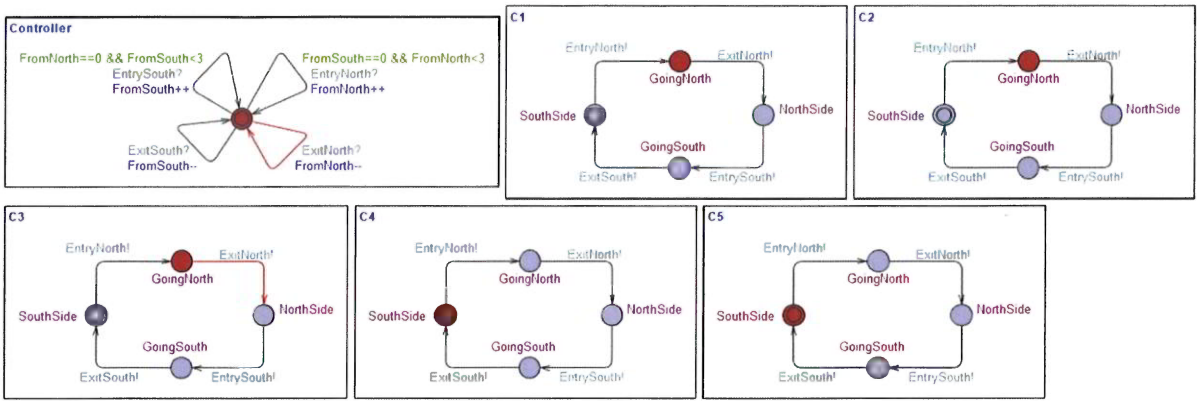
Process 1	Process 2

While True{
  <*your Monitor*>.start_eating(1)
  // do something
  <*your Monitor*>.stop_eating(1)
}
	While True{
	  <*your Monitor*>.start_eating(2)
	  // do something
	  <*your Monitor*>.stop_eating(2)
	}

Parallella processer G1F, IT404G. 2026-01-08

Main question 5

Examination criterion: Modellera och verifiera egenskaper hos program med parallella processer, inklusive progression, frånvaro av låsning och ömsesidig uteslutning



Consider the above model, which model a solution to the old bridge problem. It has a controller process with two local variables, FromNorth and FromSouth. Both of these are initialised to 0. The model also has five instances of car processes. Cars in the state, GoingNorth or GoingSouth, are on the old bridge. **Note** that the system uses the built-in mechanisms ! and ? for synchronisation (i.e., not the emulated semaphore or asynchronous message passing mechanisms from assignment). Hence, two processes will synchronise directly with each other on, e.g., channel ExitNorth in the above figure.

Sub-question 5a

Explain the role of the controller and describe what it does in detail.

Sub-question 5b

Formulate an uppaal query to verify that at most three cars can be on the bridge simultaneously.

Sub-question 5c

Formulate an uppaal query to verify that if there are cars on the bridge going south, there cannot be cars on the bridge going in the opposite direction.