Lab 11 Simon Razniewski

Conceptual Exercises on Coordination

In this lab we discuss some conceptual exercises like they could be on the exam.

Exam Instructions: You are allowed to bring one handwritten A4 cheat sheet with you (both sides may be used). For your answers, it is ok to make assumptions not mentioned in the questions, just tell which assumptions you make.

1. Lamports Logical Clocks

Consider three distributed processes P1 , P2 , and P3 . The processes are involved in the events a, b, . . . , k listed below, which happen at specific points in time, specified as "wall-clock time" (WCT) and measured in ms:

At 0ms WCT:

a: P2 sends message m1 to P1

b: P3 reads a startup file

At 100ms WCT:

c: P1 opens a file containing a user profile

d: P3 sends message m2 to P2

At 200ms WCT:

e: P1 receives message m1

f : P2 receives message m2

At 300ms WCT:

g: P1 sends message m3 to P2 and P3

At 400ms WCT:

h: P2 receives message m3

At 500ms WCT:

i: P2 sends message m4 to P1

At 600ms WCT:

j : P1 receives message m4

At 600ms WCT:

k: P3 receives message m3

1. Which of these events are related by Lamport's "happened before" relation (in the lecture denoted as "E1 → E2 " if event E1 happened before event E2)? Draw a directed graph where the events are vertices and where there is an edge from E1 to E2 if E1 happened before E2.

2. Associate to each event a logical timestamp according to the logical clock algorithm. Use process IDs to break possible ties. What is the linear order of events induced by these timestamps?

2. Concurrent Events

Consider two processes P1, P2 where sending and receiving messages are the only events.

Can you construct an example of two events that are concurrent? That is, none "happened before" the other?

Is it possible that two events are concurrent and have different logical timestamps?

3. Bully Algorithm

1. Sketch the Bully Algorithm. Remember there are 3 types of messages:

– election, vote, coordinator.

2. Execute the bully algorithm in a network with 6 peers, numbered from 1 to 6, where Peer 2 is the first to notice the crash of the leader number 6.

3. What happens if two processes notice at the same time that the leader has crashed?

4. In the Bully algorithm, a recovering process starts an election and will become the new coordinator if it has a higher identifier than the current leader.

– Is this a feature that is necessary for the algorithm to work correctly?

– Under which circumstances can it be dropped?