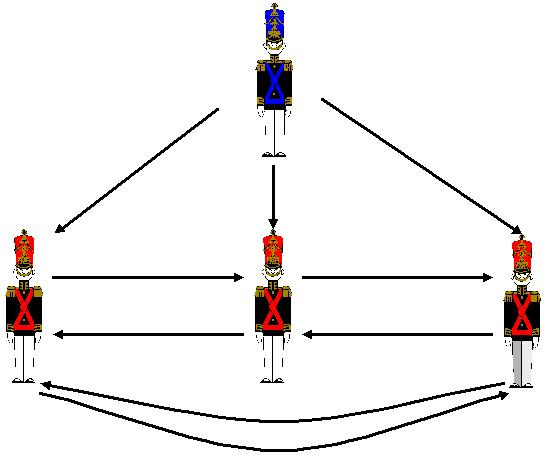
|  |  |  |
| --- | --- | --- |
|  | **Distributed Algorithms - Exercise**  **Winter term 2018/19** | **Danh Le Phuoc**  **Qian Liu**  **ODS** |
| *Exercise sheet 3* | | |

**Exercise 3.1: Distributed Consensus**

**Questions**

1. Implement the agreement of the byzantine generals for m faulty generals using pseudo-codes. Do your implementation in a way that it is possible to distinguish (1) between faulty and non-faulty lieutenant generals, (2) between the commanding general and the lieutenant generals. Explain your implementations with graphs and texts, Messages shall visualize their message path and the actual value, as well as integrity (message has been corrupted). Demonstrate your implementation given the following scenarios:
   1. n = 4 (4 generals), m = 1 (one randomly chosen general is faulty).
   2. n = 7 (7 generals), m = 2 (two randomly chosen generals are faulty).
   3. n = 10 (10 generals), m = 3 (three randomly chosen generals are faulty).



**Exercise 3.2: Fault Tolerance**

**Questions**

1. Buffer overflows (also known as pointer or heap or stack smashing) are a well-known source of programme failure. What kind of fault models would capture their effects on the rest of the system.
2. Consider the following real-life failure scenarios and examine if these can be mapped to any of the known fault classes introduced in the lecture:
3. On January 15 1990, 114 switching nodes of the long-distance system of AT&T went down. A bug in the failure recovery code of the switches was responsible for this. Ordinarily, when a node crashed, it sent out-of-service message to the neighboring nodes, prompting the neighbors to reroute traffic around it. However, the bug (a misplaced break statement in C code) caused the neighboring nodes to crash themselves upon receiving the out-of-service message. This further propagated the fault by sending an out-of-service message to nodes further out in the network. The crash affected the service of an estimated 60,000 people for 9h, causing AT&T to lose $60 million revenue.
4. A programme module of the Arianne space shuttle received a numerical value that it was not equipped to handle. The resulting variable overflow caused the shuttle’s on board computer to fail. The rocket went out of control and subsequently crashed.