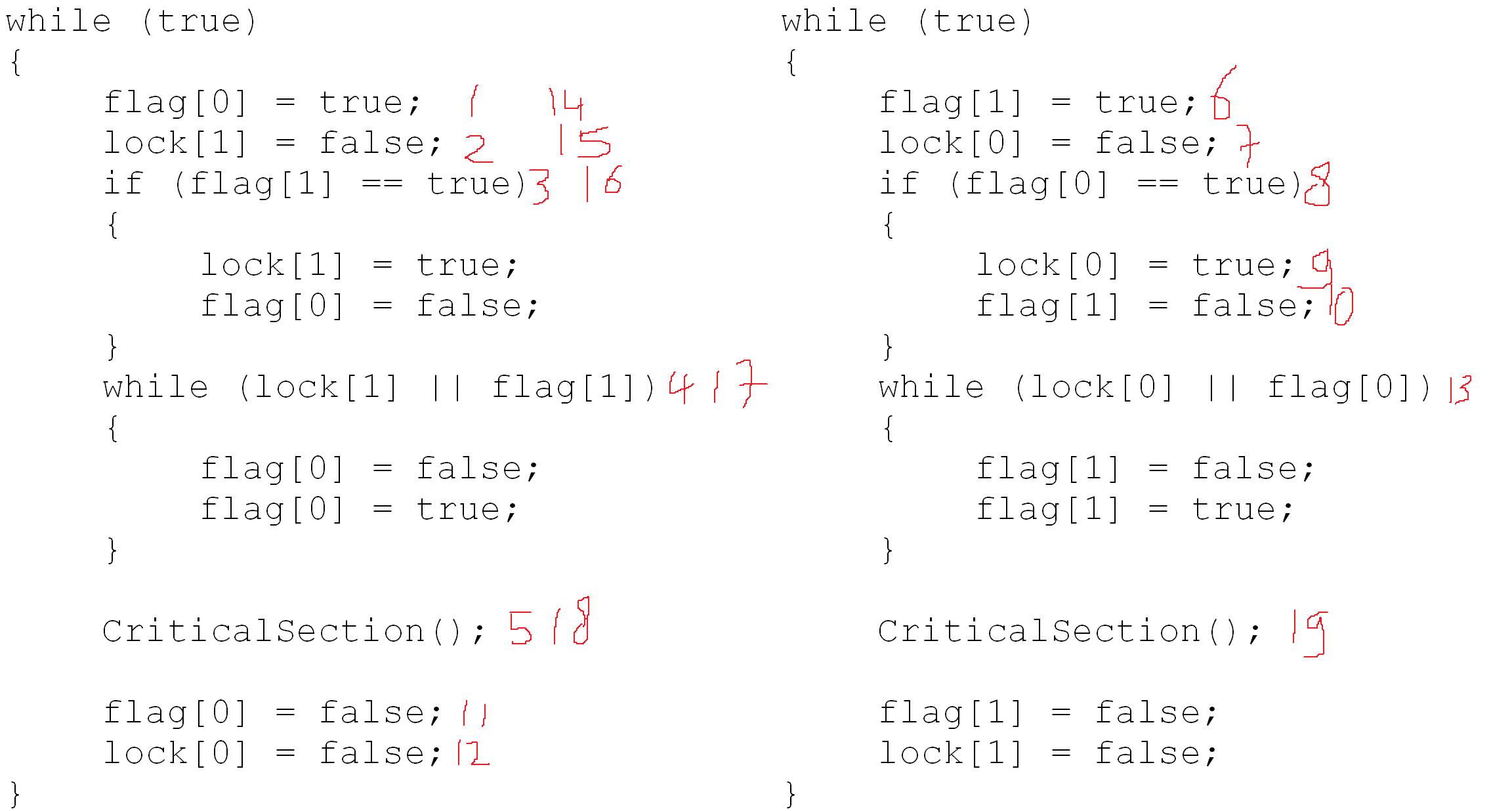
# Opdracht I

Joery van den Hoff

Ik heb hieronder de volgorde van statements aangegeven waar de situatie gecreëerd wordt waar beide processen in hun CriticalSection() zitten.



Vragen:

* Een ‘deadlock’ kan **niet** voorkomen omdat in de check of het proces verder mag, geprobeerd wordt om de andere proces voor te laten gaan. (De processen zijn dus wel in beweging en zijn niet ‘dead’.
* Echter een ‘livelock’ kan **wel** voorkomen omdat de processen in een staat kunnen zijn waardoor beide processen niet verder komen. Als de statements ‘flag[0] = false;’ en ‘flag[0] = true;’ altijd achter elkaar worden uitgevoerd (en hetzelfde voor ‘falg[1]’) voordat naar het andere proces wordt geswitched dan is de vergelijking van de while statement altijd true. En dan blijven de processen elkaar tevergeefs voorrang geven.
* Deze implementatie is **niet** fair. Als een proces naar de CriticalSection gaat dan blokkeert hij daarmee het andere proces (behalve in het eerste screenshot uiteraard) om naar de CriticalSection te gaan. Als het eerste proces terug komt uit de CriticalSection kan hij weer terug gaan naar de CriticalSection en dus het tweede proces blijven blokkeren. En dat is niet eerlijk.