Overall the average mark was 62% (lower than previous years but a reasonable average for a third-year exam). However, this was made up of some very good performances and some very poor performances. It was noticeable that attendance at lectures was low and some of the candidates I saw rarely if at all in the lectures. This is likely to be the key factor - those who engaged with the course delivery likely to have high marks.

Q1: The description of concurrency concepts was not well handled in general, with definitions not precise enough. The FSP modelling was handled well, but the multithreaded Java programming varied quite considerably.

Q2: Students did very well on using inference rules to construct derivations, with marks lost mainly in small mistakes.

Q3: The algorithm for deciding bisimilarity was described by most students, but marks often lost through imprecision. The application of the algorithm to processes was well carried out, in general, though some started with the wrong labelled transition systems for some examples.

Q4: This required students to describe various behaviours of processes, define properties and develop FSP models. Again descriptions lost marks through imprecision and incompleteness. Most defined properties correctly, but the FSP modelling was not well carried out in general.

Q5: Students either knew the material on graphs and its relationship to types of fairness in concurrency, and gained very good marks or, in quite a few cases, clearly didn’t know and therefore got poor marks.