Exercise 1: Read the Introduction to Part 2 in the notes (pages v to xiii) and reflect on the following questions:

(a) Why should we bother thinking about what computation is?
(b) What’s the difference between a program and a function?
(c) Are there things that computers cannot do? Does that matter?
(d) What does it mean for a program to be correct?
(e) Can two programs do different things (what does that mean) and still satisfy the same correctness requirements?
(f) Can two programs do the same thing (what does that mean) and have different computational complexity (runtime performance)?

You should make some notes but the TAs will mark this question by discussing your thoughts with you, rather than marking what you have written. They will be looking to see that you have engaged with the exercise; rather than having the ‘right’ answer.

Exercise 2: This question asks you to write a while program and then execute it using the semantic relation.

(a) Write a logarithm-base2 while program, i.e. a program which accepts an argument passed in variable \( x \in \mathbb{N} \), and calculates \( z \) and \( r \) such that \( x = 2^z + r \) where \( 0 \leq r < 2^z \) e.g. \( z \) contains \( \lfloor \log_2(x) \rfloor \) and \( r \) contains \( x - z \). Assume that \( x \neq 0 \).

(b) Let \( S \) be your above program. Write out the complete computation \( \langle S, [x \mapsto 5] \rangle \Rightarrow \sigma \). You should include every intermediate step.

(c) Will your program always terminate? Are there programs that might not?