Homework 10

Recall the "coffee with a friend" situation, involving the two (concurrent) processes depicted in Figure 1.

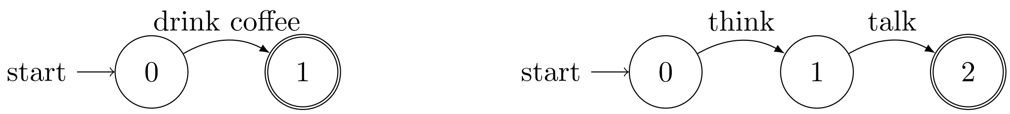

Figure 1: Two concurrent processes that are happening while having coffee with a friend

The *composition* of these two processes would be as given in Figure 2. The composition is somehow the "product" of these processes, and is a complete representation of this concurrent system. Its nodes represent the possible *states* that the system can be in, *e.g.*, $\langle 1, 1\rangle$ represents that I already drank my coffee, while I have thought about what to say, but have not said it yet. The edges give the possible transitions between these (composite) states, and so each path from the *initial state* $\langle 0, 0\rangle$ to the *final state* $\langle 1, 2\rangle$ represents a possible *trace* of this system, *e.g.*, think $\rightarrow$ drink coffee $\rightarrow$ talk.

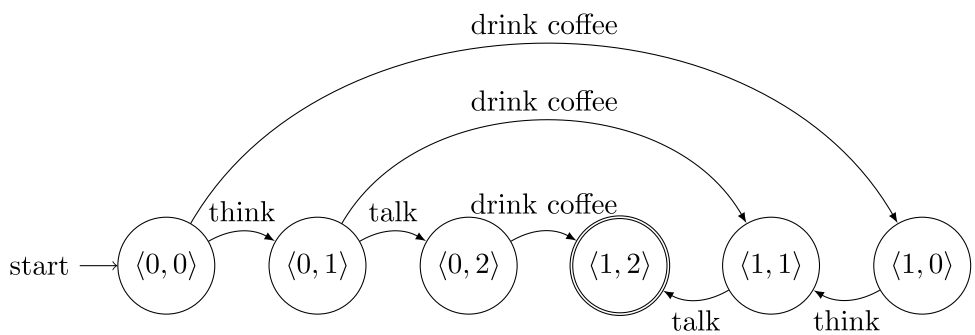

Figure 2: The *composition* of the two concurrent processes of Figure 1

One could imagine yet a third process happening concurrently to the first two, as depicted in Figure 3, while having coffee with a friend. Again, one cannot eat a cookie while drinking coffee, thinking or talking (or should not?), but any of these can happen in any order, with the exception that one must think before talking, as constrained by the second process depicted in Figure 1.

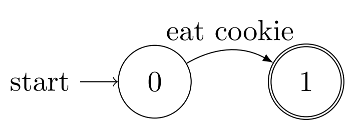

Figure 3: Another possible (concurrent) process among those of Figure 1

(Questions about this situation appear on the second page.)

Questions

Note that this assignment is out of 100 points (as indicated below for each question), and there is a bonus question worth 10 points, and so 110/100 is possible for this assignment.

1. (10 points). The composition of Figure 2 has 6 states — how many many *traces* does it have?
2. (30 points). Suppose we were to add a third (concurrent) process of "eat cookie" to our situation, as depicted in Figure 3 — draw out the resulting composition of the three concurrent processes "drink coffee", "think → talk" and "eat cookie" (you may do this on paper and take a picture).
3. (20 points: 10 points for each of (a) and (b)).
   (a) How many states does the composition of (2.) have?
   (b) How many *traces* does it have?
4. (10 points). In the above, we had the three concurrent processes p_1, p_2 and p_3, each with the number $s_1 = 2$, $s_2 = 3$ and $s_3 = 2$ of states, respectively.

   In general, for a set of n concurrent processes $p_1, p_2, \ldots p_n$, each with the number $s_1, s_2, \ldots, s_n$ of states, respectively, how many (composite) states would the composition of these n processes have?

   Bonus (10 points). How many *traces* would it have?
5. (30 points). Model this situation of three concurrent processes with Java threads, in a similar way as that of running several countdown timers concurrently[1,2]. That is, the "drink coffee" process will pause for a bit, then simply output "drink coffee" and exit (the "eat cookie" process will be similar), while the "think → talk" process will pause for a bit, output "think", pause for another bit, then output "talk" and exit. You may reuse the countdown timer code of footnotes 1 2 for your purposes — it will behave analogously, in that your code will output sequences like: drink coffee → think → eat cookie → talk, *i.e.*, *traces* of the composition of (2.)

[1] https://github.com/murraypatterson/CSC4330Code/blob/main/threads/ParamCountDown.java
[2] https://github.com/murraypatterson/CSC4330Code/blob/main/threads/RunParamCountDowns.java