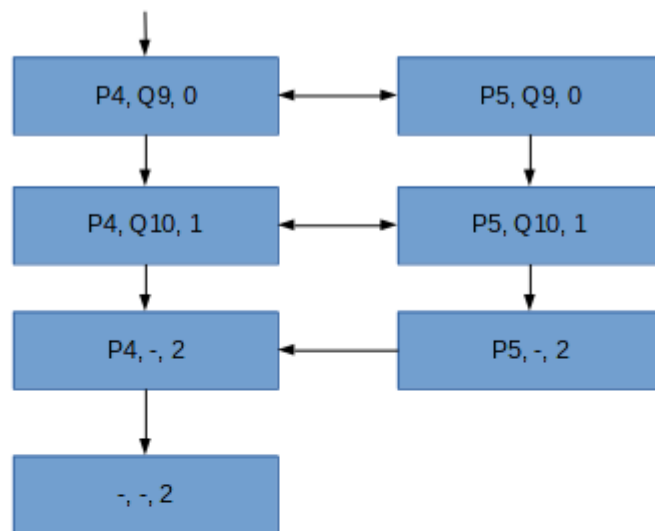


Jake Israel
CS 511
Prof. E Bonelli
September 9, 2016

Problem 1



```
1. global int n = 0
2.
3. thread P: {
4.     while (n < 2)
5.         print(n);
6. }
7.
8. thread Q: {
9.     n = n + 1;
10.    n = n + 1;
11.}
```

1. State Diagram
2. Traces

- (a) $\rightarrow [P4, Q9, 0] \rightarrow [P5, Q9, 0] \rightarrow [P4, Q9, 0] \rightarrow [P4, Q10, 1] \rightarrow [P5, Q10, 1] \rightarrow [P4, Q10, 1] \rightarrow [P5, Q10, 1] \rightarrow [P5, -, 2] \rightarrow [P4, -, 2] \rightarrow [-, -, 2]$
- (b) $\rightarrow [P4, Q9, 0] \rightarrow [P5, Q9, 0] \rightarrow [P4, Q9, 0] \rightarrow [P5, Q9, 0] \rightarrow [P4, Q9, 0] \rightarrow [P4, Q10, 1] \rightarrow [P5, Q10, 1] \rightarrow [P5, -, 2] \rightarrow [P4, -, 2] \rightarrow [-, -, 2]$
- (c) $\rightarrow [P4, Q9, 0] \rightarrow [P5, Q9, 0] \rightarrow [P4, Q9, 0] \rightarrow [P5, Q9, 0] \rightarrow [P5, Q10, 1] \rightarrow [P5, -, 2] \rightarrow [P4, -, 2] \rightarrow [-, -, 2]$
- 3. No. If thread Q executes entirely before thread P (and in other cases as well), 2 will never appear in the output
- 4. At most once. Once it has printed 2 and returns to check the condition, the thread will terminate
- 5. 0 to infinite times. Assuming thread Q never gets switched to after incrementing once, Thread P can loop an infinite number of times
- 6. 0 to infinite times. Assuming thread Q never gets switched to, Thread P can loop infinitely printing 0.
- 7. A length of 0, or an empty string. Assuming Thread Q is run entirely before Thread P ever runs, nothing will be printed.