1a)

???

1b)

(M1, s0) ⊧ pUr

iff for all paths ρ(s0): (M1, ρ) ⊧ pUr

iff there exists a j such that (M1, ρj) ⊧ pUr and for all 0 <= k < j (M1, ρk) ⊧ pUr

true for j = 1 because any path ρ is of the form s0s1… or s0s3… and s0 ∈ 𝜋(p), s1,s3 ∈ 𝜋(r)

(M1, s0) ⊧ GF(p)

iff for all paths ρ(s0): (M1, ρ) ⊧ GF(p)

iff for all i >= 0 such that (M1, ρi) ⊧ F(p)

iff there exists a j >= 0 s.t. (M1, ρj) ⊧ p

Not true because the path ρ = s0s1[s2]ω doesn’t satisfy this. For i >= 2 there is no j for which (M1, ρj) ⊧ p because all states are s2 and s2 ∉ 𝜋(p)

Same as above

c)

The constraint is fairness.

The user can add the line ‘FAIRNESS !t’ to ensure that t becomes false infinitely often. This will disregard paths that loop on s2 forever because t is true in s2.

The model checker would verify the specification by doing …

Guess: We can express infinitely often as GF (or AGAF) so we can check that the fairness condition is met using the normal methods

It would hold because…

d) We need to show we can’t construct a model and path where one holds and one doesn’t