COMP307 Assignment 4 David Burrell:300209541

Part One:

1. Init(At(M:Monkey, A) ^ Level(M, Low) ^ Have(M, ¬Ban : ¬Bananas))
   Goal(Have(M, Ban))

2. Action (Move (M, from, to))

   PRECON: At (M, from) ^ Level (M, Low)

   EFFECT: ¬At (M, from) ^ At (M, to)

   Action (Push (M, b: Box, from, to))

   PRECON: At (M, from) ^ At (b, from)

   EFFECT: ¬At (M, from) ^ ¬At(b, from) ^ At(M, to) ^ At(b, to)

   Action (ClimbUp (M, b, location))

   PRECON: At (M, location) ^ At (b, location) ^ Level (M, Low)

   EFFECT: At (M, location) ^ At (b, location) ^ Level (M, High)

   Action (ClimbDown (M, b, location))

   PRECON: At (M, location) ^ At (b, location) ^ Level (M, High)

   EFFECT: At (M, location) ^ At (b, location) ^ Level (M, Low)

   Action (Grasp (M, Ban, location))

   PRECON: At (M, location) ^ At (Ban, location) ^ Level (M, High) ^ ¬Have (M, Ban)

   EFFECT: Have (M, Ban)

   Action (Ungrasp (M, Ban,))

   PRECON: Have (M, Ban)

   EFFECT: ¬Have (M, Ban)

3. **LAYER ONE**.

   NODE 1-1
   At(M : Monkey, A) ^ At(b : Box, B) ^ At(Ban : Banana, C) ^
   Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^
   ¬Have(M, Ban)

   LAYER TWO.

   NODE 2-1 (Edge from 1-1)

   Action: Move(M, A, B)

   !At(M, A) ^ At(M, B) ^ At(b, B) ^ At(Ban, C) ^ Level(M, Low) ^
   Level(b, Low) ^ Level(Ban, High) ^ ¬Have(M, Ban)

NODE 2-2 (Edge from 1-1)

Action: Move (M, A, C)

!At(M, A) ^ At(M, C) ^ At(b, B) ^ At(Ban, C) ^ Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^ ¬Have(M, Ban)

LAYER THREE

NODE 3-1 (Edge from 2-1)

Action: Push (M, b, B, C)

!At(M, B) ^ !At(b, B) ^ At(M, C) ^ At(b, C) ^ Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^ ¬Have(M, Ban)

NODE 3-2 (Edge from 2-1)

Action: Move(M, B, C)

!At(M, B) ^ At(M, C) ^ At(b, B) ^ At(Ban, C) ^ Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^ ¬Have(M, Ban)

NODE 3-3 (Edge from 2-1)

Action: Move(M, B, A)

!At(M, B) ^ At(M, A) ^ At(b, B) ^ At(Ban, C) ^ Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^ ¬Have(M, Ban)

NODE 3-4 (Edge from 2-1)

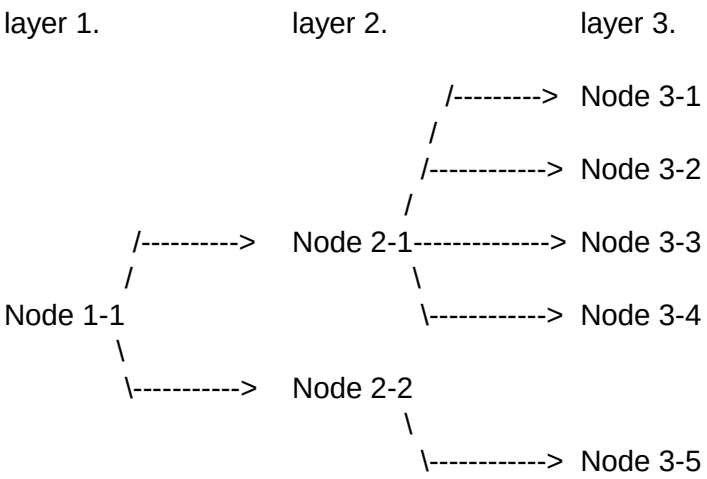
Action: ClimbUp(M, b, B)

At(M, B) ^ At(b, B) ^ At(Ban, C) ^ Level(M, High) ^ Level(b, Low) ^ Level(Ban, High) ^ ¬Have(M, Ban)

NODE 3-5 (Edge from 2-2)

Action: Move(M, C, B)

!At(M, C) ^ At(M, B) ^ At(b, B) ^ At(Ban, C) ^ Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^ ¬Have(M, Ban)



4. Initial State: At(M : Monkey, A) ^ At(b : Box, B) ^ At(Ban : Banana, C) ^
   Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^
   ¬Have(M, Ban) ^ (M : Monkey) ^ (b : Box) ^ (Ban : Banana)
   Action 1: Move(M, A, B)
   State 1: At(M, B) ^ At(b, B) ^ At(Ban, C) ^
   Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^
   ¬Have(M, Ban) ^ (M : Monkey) ^ (b : Box) ^ (Ban : Banana)
   Action 2: Push(M, b, B, C)
   State 2: At(M, C) ^ At(b, C) ^ At(Ban, C) ^
   Level(M, Low) ^ Level(b, Low) ^ Level(Ban, High) ^
   ¬Have(M, Ban) ^ (M : Monkey) ^ (b : Box) ^ (Ban : Banana)
   Action 3: ClimbUp(M, b, C)
   State 3: At(M, C) ^ At(b, C) ^ At(Ban, C) ^
   Level(M, High) ^ Level(b, Low) ^ Level(Ban, High) ^
   ¬Have(M, Ban) ^ (M : Monkey) ^ (b : Box) ^ (Ban : Banana)
   Action 4: Grasp(M, Ban, C)
   State 4: At(M, C) ^ At(b, C)^ At(Ban, C) ^
   (goal) Level(M, High) ^ Level(b, Low) ^ Level(Ban, High) ^
   Have(M, Ban) ^ (M : Monkey) ^ (b : Box) ^ (Ban : Banana)

Part Two:

1. Process(O11, M1, t1) -> Process(O21, M2, t2) -> Process(O31, M1, t3) ->
   Process(O12, M2, t4) -> Process(O22, M1, t5) -> Process(O32, M2, t6).

   Pr(O11, M1, t1) arrives at 0, uses M1 for 50. (M1 in use from 0-50)
   Pr(O21, M2, t2) arrives at 10, uses M2 for 30 (M2 in use from 10-40)
   Pr(O31, M1, t3) arrives at 20, needs M1
   M2 available at 40 Pr(O22, M1, t5) ready
   M2 idle 40-50 (10)
   M1 available at 50. Pr(O12, M2, t4) ready
   Pr(O31, M1, t3) uses M1 for 40 (M1 in use from 50-90)
   Pr(O12, M2, t4) uses M2 for 25 (M2 in use from 50-75)
   M2 available at 75 [J1 complete]
   M2 idle 75-90 (15)
   M1 available at 90 Pr(O32, M2, t6) ready
   Pr(O22, M1, t5) uses M1 for 35 (M1 in use from 90-125) [J2 complete]
   Pr(O32, M2, t6) uses M2 for 20 (M2 in use from 90-110) [J3 complete]

   t1 = 0, t2 = 10, t3 = 50, t4 = 50, t5 = 90, t6 = 90

2. **Completion times**: J1 = 75, J2 = 125, J3 = 110.
   Makespan = 125.

3. **Step 0**:
   Partial solution: (empty, no action is scheduled)
   earliestIdleTime(M1) = 0, earliestIdleTime(M2) = 0
   earliestReadyTime(O11) = 0, earliestReadyTime(O12) = ∞
   earliestReadyTime(O21) = 10, earliestReadyTime(O22) = ∞
   earliestReadyTime(O31) = 20, earliestReadyTime(O32) = ∞

Step 1:
Partial solution: Pr(O11,M1,0) scheduled
earliestIdleTime(M1) = 50, earliestIdleTime(M2) = 0
earliestReadyTime(O11) = 0, earliestReadyTime(O12) = 50
earliestReadyTime(O21) = 10, earliestReadyTime(O22) = ∞
earliestReadyTime(O31) = 20, earliestReadyTime(O32) = ∞

Step 2:
Partial solution Pr(O11,M1,0) Pr(O21,M2,10) scheduled
earliestIdleTime(M1) = 50, earliestIdleTime(M2) = 40
earliestReadyTime(O11) = 0, earliestReadyTime(O12) = 50
earliestReadyTime(O21) = 0, earliestReadyTime(O22) = 40
earliestReadyTime(O31) = 20, earliestReadyTime(O32) = ∞

Step 3:
Partial solution: Pr(O11,M1,0) Pr(O21,M2,10)complete, Pr(O22,M1,50),
Pr(O12,M2,50) scheduled.
earliestIdleTime(M1) = 85, earliestIdleTime(M2) = 75
earliestReadyTime(O11) = 0, earliestReadyTime(O12) = 0
earliestReadyTime(O21) = 0, earliestReadyTime(O22) = 0
earliestReadyTime(O31) = 20, earliestReadyTime(O32) = ∞

4. **Completion times**:
   J1 = 75, J2 = 85, J3 = 145.
   Makespan: 145.
5. No it doens't mean that one is strictly better, there could always be a time in which one of the solutions would outperform the other, in more complicated systems.

Part Three:

1. R1 = (1, 2, 3, 5, 1)
   R2 = (1, 6, 8, 4, 1)
   R3 = (1, 7, 9, 10, 1)

2. L(1,2)+L(2,3)+L(3,5)+L(5,1) = 5.24
   L(1,6)+L(6,8)+L(8,4)+L(4,1) = 7.39
   L(1,7)+L(7,9)+L(9,10)+L(10,1) = 12.78
   Total Euclidean Length: 25.41