1.

First, we need to find appropriate invariants.

[inv 1] $m(p1) + m(p2) + m(p3) + m(p4) + m(p5) = Ph1 + Ph2 + Ph3 + Ph4 + Ph5$

- Each philosopher is either: in the thinking room ($p1$), they are without a fork in the dining room ($p2$), they have a left fork in the dining room ($p3$), they are eating in the dining room ($p4$), or they have finished eating but still have a right fork in the dining room ($p5$).

[inv 2] $RF(m(p4)) + LF(m(p4)) + m(p6) = F1 + F2 + F3 + F4 + F5$

- We can apply the same invariant used in slide 33 of LN12, as each fork is either unused ($p6$), or being used by a philosopher to eat with both hands ($p4, p6$).

[inv 3] $|m(p7)| + |m(p2)| = 4$

- From the given petri net and looking back at slide 15 of LN9, we know that any 4 philosophers are allowed inside the dining room at any given time.

Now, we consider the two cases

(a) $m(p4) + m(p5) \neq 0$. If philosophers are still eating and/or philosophers are finished eating but still have right forks in the dining room, then RETURN_LEFT_FORK or RETURN_RIGHT_FORK_AND_EXIT_DINING_ROOM can be fired.

(b) $m(p4) + m(p5) = 0$. If no philosophers are eating, and/or no philosophers are finished eating and still have a right fork in the dining room, then that means we have further sub-cases to consider. We Assume that m is reachable from the initial marking. Then m satisfies [inv 1] and [inv 2]. If $m(p1) = Ph1 + Ph2 + Ph3 + Ph4 + Ph5 = 5$, then $m(p2) = 0$ and by [inv 3], ENTER_DINING_ROOM can be fired. If $m(p2) \neq 0$, then TAKE_LEFT_FORK can be fired. If $m(p3) \neq 0$, then TAKE_RIGHT_FORK can be fired.

$\therefore$ mimicking the proof of Proposition from page 33 of LN12, we have shown that the given solution to Dining Philosophers is deadlock free.

7.

(a)

(i) $\neg p \Rightarrow r \equiv \neg(\neg p) \vee r \equiv p \vee r$. We have $L(s_0) = \{r\}$ so $M, s_0 \models \varphi$ holds. We have $L(s_2) = \{p, q\}$ so $M, s_2 \models \varphi$.

(ii) $\neg EG\ r$. This can be translated to "There does not exist at least one path from all future/global states leading to r". We have $L(s_0) = \{r\}$ and $L(s_1) = \{p, t, r\}$, so $r \in s_0, s_1$. Now s_1 is reachable from s_0, and we can follow this path infinitely to stay in s_1. Since we established that $r \in s_0, s_1, M, s_0 \models \varphi$ does NOT hold.

We have $L(s_2) = \{p, q\}$. Since we have that $r \notin s_2$, we can immediately assert that $M, s_2 \models \varphi$ definitely holds as s_2 would have to be included as a possible path.

(iii) $E(t\ U\ q)$. This can be translated to "There exists at least one path in which t can occur until q". We have $L(s_0) = \{r\}$ and $L(s_2) = \{p, q\}$. Although $q \in s_2, t \notin s_0$ and $t \notin s_2$. So $M, s_0 \models \varphi$ does NOT hold and $M, s_0 \models \varphi$ does NOT hold.

(iv) $F\ q$. This can be translated to "Some future state leads to q". We have $L(s_2) = \{p, q\}$, so $q \in s_2$. If we take a path from s_0, then there is an infinite path to s_2, which can be modeled by the trace $s_0 \rightarrow s_2 \rightarrow s_0 \rightarrow s_2 \rightarrow ...$, so $s_0 \models \varphi$ holds. Clearly $s_2 \models \varphi$ holds since $q \in s_2$.

NOTE: For the following parts of this question, they are some assumptions I am making to ensure the models are valid. I will assume the word "between" in a case such as "event q is between event p and event r" means that q is an event that does not take place in at the same time as p or r. I will also assume the word "precedes" in a case such as "event p precedes event q" means that p must happen before q, and they cannot happen at the same time. The opposite can be assumed for "followed" in a case such as "event q is by followed by event p".

(b)

"Event p precedes s and t on all computational paths."
LTL: $G(F\ p \wedge (p \Rightarrow F\ s) \wedge (P \Rightarrow F\ t))$
CTL: $AG(AF\ p \wedge AG(p \Rightarrow F\ s)) \wedge AG(P \Rightarrow F\ t))$

(c)

"Between the events q and r, p is never true but t is always true."
LTL: $G(F\ q \wedge F\ r \wedge (q \Rightarrow (\neg p\ U\ r) \wedge q \Rightarrow (F\ t\ U\ r)))$
CTL: $AG(AF\ q \wedge AF\ r) \wedge AG(q \Rightarrow A(\neg p\ U\ r) \wedge q \Rightarrow A(AF\ t\ U\ r)))$

(d)

"Φ is true infinitely often along every paths starting at s."
LTL: $G(F\Phi)$
CTL: $AG(AF\Phi)$

(e)

"Whenever p is followed by q (after some finite amount of steps), then the system enters an 'interval' in which no r occurs until t."
LTL: $G(P \Rightarrow XG(q \Rightarrow (\neg r\ U\ t)))$
CTL: $AG(P \Rightarrow AXAG(q \Rightarrow A(\neg r\ U\ t)))$

(f)

"Between the events q and r, p is never true."
LTL: $G(F\ q \wedge F\ r \wedge (q \Rightarrow (\neg p\ U\ r)))$
CTL: $AG(AF\ q \wedge AF\ r) \wedge AG(q \Rightarrow (\neg p\ U\ r))$

8.

We can use the solution for Readers-Writers as discussed in chapter 7 of the textbook to help provide a model. We assume the following atomic predicates that characterise properties of processes (I assume this means to represent the behaviour of the readers and writers in different states, n represents them 'idling', t represents them requesting to read/write, and c represents them reading or writing):

- n_1 = LPR$_i$ - local processing of reader $i, i = 1, 2$
- n_2 = LPW$_i$ - local processing of writer $i, i = 1, 2$
- t_1 = TR$_i$ - reader $i, i = 1, 2$, requests reading
- t_2 = TW$_i$ - writer $i, i = 1, 2$, requests writing
- c_1 = R$_i$ - reader $i, i = 1, 2$, is reading
- c_2 = W$_i$ - writer $i, i = 1, 2$, is writing

To avoid issues that may arise if we do not consider mutual exclusion, we have to introduce additional boolean variables (or atomic predicates):

- turn = W_1 (indicating the worlds where writer 1 will write)
- turn = W_2 (indicating the worlds where writer 2 will write)
- turn = R (indicating the worlds where one or both readers will read)

Now, we can establish the states of a state machine that defines the model:

$$\text{States the readers could be in} = \begin{cases} S_0 = \{LPR_1, TR_1, R_1\} \\ S_1 = \{LPR_2, TR_2, R_2\} \end{cases}$$

$$\text{States the writers could be in} = \begin{cases} S_2 = \{LPW_1, TW_1, W_1\} \\ S_3 = \{LPW_2, TW_2, W_2\} \end{cases}$$

$$\text{State of turns} = \begin{cases} S_4 = \{\text{turn} = W_1, \text{turn} = W_2, \text{turn} = R\} \end{cases}$$

Finally we establish safety and liveness properties, according to the ones presented in LN12 (I chose to write in LTL):

$$\text{We have two similar safety properties} \begin{cases} G(W_1 \Rightarrow \neg(W_2 \vee R_1 \vee R_2)) & W_1 \text{ must have exclusive access to the db.} \\ G(W_2 \Rightarrow \neg(W_1 \vee R_1 \vee R_2)) & W_2 \text{ must have exclusive access to the db.} \end{cases}$$

$$\text{and four similar liveness properties} \begin{cases} G(TR_1 \Rightarrow F\ R_1) & R_1 \text{ follows this path} \\ G(TR_2 \Rightarrow F\ R_2) & R_2 \text{ follows this path} \\ G(TW_1 \Rightarrow F\ W_1) & W_1 \text{ follows this path} \\ G(TW_2 \Rightarrow F\ W_2) & W_2 \text{ follows this path} \end{cases}$$