Here are some comments on reducibility and the blank tape halting problem, Example 12.2.

We proved (or rather, Turing did, by a rather mind-boggling proof by contradiction) that the Halting Problem is unsolvable. That is, there does not exist an algorithm that, given any Turing machine T and word w, can decide whether T run on w eventually halts. So the "input" to the Halting Problem is a (T,w) pair. This is what you want to hand to the oracle and wait for your yes/no answer, only no such oracle exists.

Now we want to show that the blank tape halting problem is unsolvable. That is, we want to prove that no algorithm exists that, given any Turing machine T, can decide whether T run on the blank tape eventually halts. This does not fall out automatically from the unsolvability of the Halting Problem. After all, this question is a bit easier - you are not asking for answers for every w on which you might run your T, you are only asking for the answer when starting T on a blank tape. Because this is a "restricted" version of the problem, there might be an oracle that could answer this question; what you would bring to such an oracle is just T.

But such is not the case; this seemingly simpler problem is also unsolvable - no such oracle (algorithm) exists. But to prove it, we do not have to do a proof "from scratch", as we did for the Halting Problem. Instead, we can use a reducibility argument and show that the Halting Problem is reducible to the Blank Tape problem.

A is reducible to B means if we could find a solution for B we would also have a solution to A. The problem of solving A "reduces to" the problem of solving B. In other words, B is "at least as hard to solve" as A. If we already know that A is unsolvable, however, this means that B must also be unsolvable (because if we could solve B, we could then solve A, which we can't do).

We want to show that the Halting Problem is reducible to the blank tape problem. [Note the direction here: we do not show that the blank tape problem is reducible to the Halting Problem, meaning if we could solve the Halting Problem we could solve the blank tape problem. This is trivial (and not useful); if we can answer, for any T and w whether T halts on w, we can tell, for any T, whether T halts on a blank tape; this is just a case of w = empty string.]

So again, we want to show that the Halting Problem is reducible to the blank tape problem. The confusing thing about these reducibility problems is that you don't start with the "ingredients" for the blank tape problem, which would be just a machine T; you start with the ingredients for the Halting Problem, which is a (T,w) pair. And you don't need to talk about ANY machine starting on a blank tape, you can talk about a particular machine that depends on the (T,w).

So, you are given a T and a w (arbitrary T and w). You can algorithmically build a new machine T*, based on this T and w, as follows:

step 1. T* starts on a blank tape and writes the word w on that tape. For example, if w begins with 101, then T* begins like this:

$\delta(q_0, B) = (q_1, 1, R)$
$\delta(q_1, B) = (q_2, 0, R)$
$\delta(q_2, B) = (q_3, 1, R)$
etc.

Note that w is "hard wired" into T* - a different w would require a different T*.

step 2. Move the read head of T* back over w so that you are reading the leftmost non-blank cell on the tape.

step 3: Mimic the actions of T when started on w. This means that you add the transitions for T on the back of the transitions you already have for T*, bumping up the subscript as needed. If you used states 0-9 to finish step 2 in T*, then state 10 in T* will act like state 0 in T, state 11 in T* will act like state 1 in T, etc.

Steps 1, 2, and 3 constitute an algorithm, call it H, to build T* from (T,w). Furthermore,

(1) T* halts when started on a blank tape $\leftrightarrow$ T halts when started on w

Statement (1) is true because you made "running T on w" the backend of what T* does on a blank tape.

Now assume that there is a solution to the blank tape halting problem. More specifically, let P be the algorithm that solves the blank tape problem. Then you have a solution to the Halting Problem, as follows:

Given (T, w), first apply algorithm H to build T*, then apply algorithm P to decide whether T* halts on a blank tape. If the answer is yes, then because of (1), T halts on w. If the answer is no, then because of (1) T does not halt on w. Therefore you have the answer to whether T halts on w. Contradiction (this problem is unsolvable). Therefore the assumed solution algorithm P for the blank tape problem cannot exist, and that problem is unsolvable.

In order to solve problem 5, Section 12.1, you must do something similar. Prove that the Halting Problem is reducible to the halts-on-everything problem. Find an algorithm (the equivalent of H) that converts (T,w) into a machine T** such that

T** halts on everything $\leftrightarrow$ T halts on w