Question 2

Consider the ***HarryPorter*** Database again. To handle multi-user updates, Sylvia moved this database to Oracle 19.

She wants to link up the table ***Dialogue*** and ***Chapter*** via attribute *Chapter ID*.

She updates the database in the following sequence:

1) At first, she inserts the ***Dialogue*** record by setting *Chapter ID* = -1.
2) Then, she inserts a new record ***Chapters***.
3) However, *Chapter ID* in ***Chapters*** is an auto generating number (incrementing by 1 for every new record), she then writes the third statement to select the **latest added** ***ChapterID*** in ***Chapters*** (with the largest value) and then update the **latest inserted** ***Dialogue*** **record** (with *ChapterID* = -1).

The following are the SQL statements designed according to the above scenario.

```
insert into Dialogue (ChapterID, PlaceID, CharacterID,
Dialogue) values (-1, 71, 84, "Yes I am UserX");

# UserX is the user to execute this SQL


insert into Chapters (Chapter Name, Movie ID, Movie Chapter)
values ("Hong Kong", 8, 26);


update Dialogue set ChapterID = select max(ChapterID) from
Chapters where ChapterID = -1;
```

During testing, Sylvia finds that when **two users** (User1 and User2) execute the above SQL statements at the same time, sometimes the following values would be appeared **after execution**. (Assume that the *ChapterID* in ***Chapter*** is 10 before the two users executing the above SQL statements.)

Records in table ***Chapter***

```
(11, “Hong Kong”, 8, 26)

(12, “Hong Kong”, 8, 26)
```

Records in table ***Dialogue***

```
(34, 12, 71, 84, "Yes I am User1") # the chapter ID should be 11

(35, 12, 71, 84, “Yes I am User2”)
```

DialogueID* in ***Dialogue* is an auto generating number (incrementing by 1 for every new record)

(a) Explain why this happen when two users execute above statement almost at the same time (*ChapterID* in ***Dialogue*** with the same value in two records***)***. Draw timeline to explain.

(b) To resolve the problem in (a), we suggest Sylvia to use a 5-CPU computer such that the system could process five programs at the same time. The following two solutions were proposed.

Solution 1:

Whenever enter this program, user locks two tables, ***Dialogue*** and ***Chapters*** at the beginning. Then, release the locks of tables when the updates are completed.

Solution 2:

Whenever enter this program, user locks the records to be selected or updated at the beginning. Then, release the locks of the records when the updates are completed.

Consider each request requires ***T*** seconds to be completed. Compare the total processing time of above two solutions by considering 10 requests entered simultaneous. Explain and conclude which of the proposed solutions would be a better solution.