Problem 6—Time Addition

Although Jack and his friends maintain that they operate in "real time," the fact remains that commercial breaks abound and the difference between the new time and the old time is always significantly longer than the commercial break.

Given the time on the screen just before the commercial break and the amount of time that has elapsed in Jack's universe during the commercial break (i.e. not the actual length of the commercial break), you are to compute the time on the screen immediately following the commercial break.

INPUT SPECIFICATION. Each input case will consist of two times separated by one space and followed by one **<EOLN>**. The first time will be of the form *[h]h:mm:ss* (the hours will not contain a leading zero) and represents the time just before the commercial break. The second time will be of the form *[m]m:ss* (always under an hour) and represents the elapsed time. **<EOF>** terminates the input.

OUTPUT SPECIFICATION. The output cases should appear in the same order as the input cases. Each output case will be of the form "Case *c*: The new time is *[h]h:mm:ss***<EOLN>**" (where *c* is the number of the input case and *[h]h:mm:ss* is the time just following the commercial break). The printed hours should not contain a leading zero, and the clock in question is a twelve hour clock without regard for A.M. or P.M. An extra **<EOLN>** follows each output case.

SAMPLE INPUT.

```
4:00:00·5:13<EOLN>
12:59:59·0:02<EOLN>
<EOF>
```

SAMPLE OUTPUT.

```
Case·1:·The·new·time·is·4:05:13<EOLN>
<EOLN>
Case·2:·The·new·time·is·1:00:01<EOLN>
<EOLN>
<EOF>
```