Theorem for Calculating Time Duration

Shashwat Singh

Abstract: While developing an app the author came across the fact that while calculating time durations, certain variables are often overlooked which can result in errors. This paper provides an algorithm that considers the absolute values of the time slots and after a discrete set of steps results in the accurate time durations. The scope of this research is not limited to hours and minutes but, can be applied to seconds and milliseconds as well.

Keywords: This paper provides an algorithm that considers.

I. INTRODUCTION

For years, we assumed that calculating the duration between two time slots is easy and can be found by subtracting them. The reality is that certain cases and variables have to be considered before we get the correct answer, which is mentioned in this paper so that it is much easier to calculate the duration between two time slots.

I have tried to find out every variable that needs to be considered before we get the correct answer as it is not as straightforward as subtracting two numbers.

The need for this algorithm came up when I was developing a native app using Dart, where I had to find the duration between the current time and the time a user has entered which included the Hours and the Minutes.

As I could not find a simple and to-the-point solution to this, even after searching multiple online sources, I decided to finally develop it on my own.

II. OBSERVATION

The observation will be better explained with the help of two examples.

A. CASE-1:

Suppose there are two time period's t_1 and t_2.

$$\text{Let } t_1 = 13:30$$

$$\text{And, } t_2 = 15:45$$

Now for this case, both the time slots must be in either Post Meridiem or Ante Meridiem and can be calculated by just subtracting the hour of older times from the hour of newer time, and the minute of older time from the minute of newer time.

$$\text{Time difference (for hour)} = |t_{2hour} - t_{1hour}|$$

$$= 15\text{-}13$$

$$= 2 \text{ hours}$$

$$\text{Time difference (for minute)} = |t_{2minute} - t_{1minute}|$$

$$= 45\text{-}30$$

$$= 15 \text{ minutes}$$

Therefore, the total time difference is 2 hours and 15 minutes.

B. CASE-II:

Suppose there are two time period's t_1 and t_2.

$$\text{Let } t1 = 07:00$$

$$\text{And, } t2 = 18:00$$

Now, if we try the same approach used in CASE-I, we find out that the time difference that we receive is not accurate.

$$\text{Time difference (for hour)} = |t2hour - t1hour|$$

$$= |18 - 07|$$

$$= 11 \text{ hours}$$

This answer is not correct as the time between 6 p.m. and 7 a.m. is 13 hours.

III. PROCEDURE

In such cases, one must follow certain steps before calculating the time difference, which is:

We will first find out the minute and then we will calculate the hourly periods.

Here, let the starting time be current Minute and current Hour.

Let the ending time be new Minute and new Hour.

Both current Hour and new Hour are in 24-hour format.

Let the duration in minutes and hours be min Diff and hour Diff respectively.

Note: While comparing the two minutes and hours, we look at the numerical value and compare accordingly. For example, from 22:00 and 02:00, 22>2.

The solution is explained numerically and graphically below.

A. Numerically

a. For Minutes

We first need to check if *currentMinute > newMinute* ,

If this is the case, then minDiff = (60 - currentMinute) + newMinute

Otherwise if *currentMinute < newMinute*,

Then in this case, minDiff = (60 - currentMinute) + newMinute

Finally, if none of the above cases apply and *currentMinute = newMinute*,

Then the minDiff = currentMinute - newMinute

The minDiff will be the duration for minutes, which will be added to the final time after calculating for hours.

b. For hours

After calculating the time duration for minutes, we will calculate the time duration for minutes.

Here, we first need to check if *currentHour > newHour*,

If this is the case, we will subtract the currentHour from 24 (hourFrom24) = 24 - currentHour

The hourDiff = (-1) + (hourFrom24 + newHour)

Otherwise if *currentHour < newHour,*

Then in this case, the hourDiff = (-1) + (newHour - currentHour)

Finally, if none of the above cases apply and *currentHour = newHour*,

We look at the minutes again.

Now, If currentMinute < newMinute,

Then hourDiff = 0

Else if currentMinute > newMinute,

Then hourDiff = 23

Else if currentMinute = newMinute,

hourDiff = 24

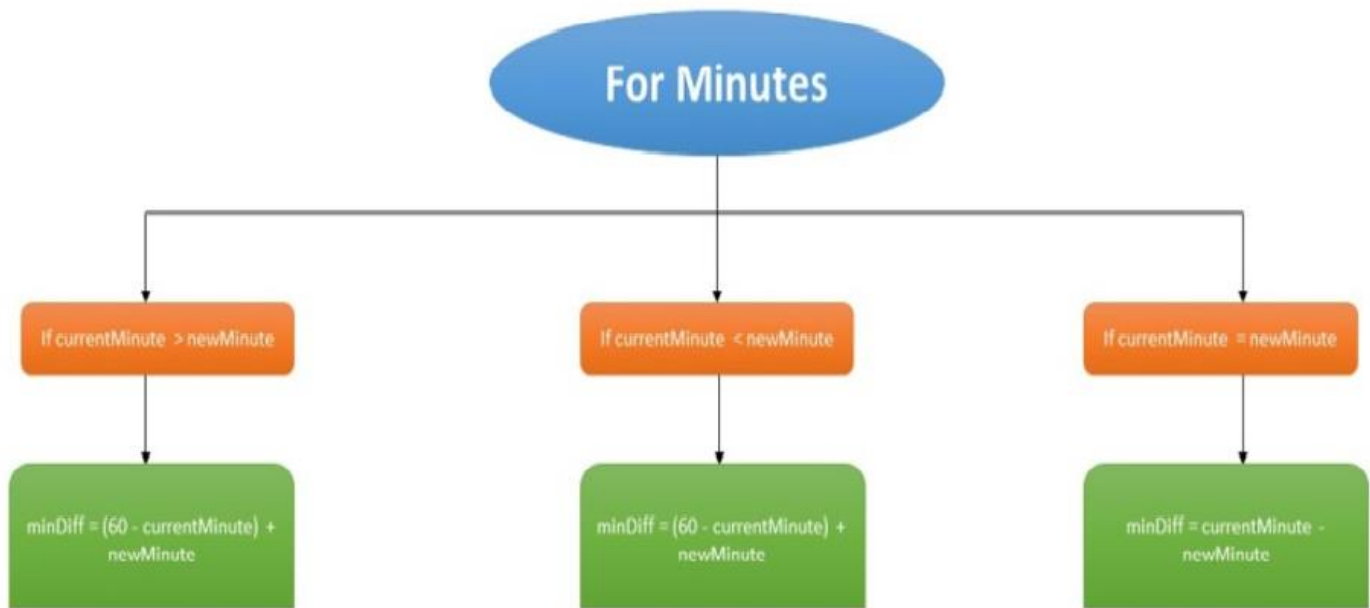Now, since minutes can't be more than 60 we have to check one last thing

If minDiff > 60,

minDiff changes to = minDiff - 60

And hourDiff changes to = hourDiff + 1

Therefore, the final duration between any two times is **hourDiff : minDiff.**

B. Flowchart

IV.   RESULTS

We need to check if the minute and hour slots are numerically equal or different. After which we need to check if they are greater than, less than, or equal to each other. Only then can we calculate the correct time difference. Each step takes us to a final step which gives the most accurate duration.

In the case of seconds, the same method which is being used for minutes shall be followed, and in the case of milliseconds, again the same method as minutes except for this time, the current millisecond shall be subtracted from 100 instead of 60 as 1 second = 100 milliseconds.

V.   CONCLUSION

We found out that to calculate the duration between two given time slots, certain steps are involved.

While traditional methods can be used for this, a computer cannot calculate it until given a set of rules which will ensure that the calculations are accurate. This paper aims to provide this knowledge to anyone and everyone who might need it in their future project development.

REFERENCES

1. Microsoft, (n.d.). Calculate the difference between two times [Online]. Available: https://support.microsoft.com/enus/office/calculate-the-difference-between-two-times-e1c78778-749b-49a3-b13e-737715505ff6,
2. On the Clock, (2018). How to calculate the duration between 2 times [Online]. Available: https://www.ontheclock.com/time-duration-calculator-hours-minutes.aspx,
3. Khan Academy, (2011) Time differences [YouTube video]. Available: https://www.youtube.com/watch?v=SnkUkc23YC0

AUTHOR PROFILE

Shashwat Singh, graduated from Ryan International School in the year 2021. He is an app developer and founder of Learn with Shash and DOPLOP. He is currently working with Amazon Web Services on an app that would help everyone wake up early in the morning or at their desired time period by eliminating the snooze button and adding some other interesting features.