Daily Coding Problem #303

Problem

This problem was asked by Microsoft.

Given a clock time in `hh:mm` format, determine, to the nearest degree, the angle between the hour and the minute hands.

Bonus: When, during the course of a day, will the angle be zero?

Solution

This is more of a mathematical puzzle than a coding problem, but since these kinds of puzzles show up occasionally in interviews, it is useful to be able to recognize and solve them.

We can start by recalling that there are `360` degrees in a circle. Since there are `60` ticks that the minute hand sweeps through, each tick corresponds to `360 / 60 = 6` degrees. Similarly, since there are twelve hours represented on a clock face, each one corresponds to `360 / 12 = 30` degrees.

In addition, note that in between each hour, the hour hand moves across one extra tick every `12` minutes. Therefore, we must add an extra six degrees to the hour angle for every `12` ticks represented by the minute hand.

Finally, if the hands do not form a straight line, there will always be two ways of looking at angle: acute and obtuse. We must remember to take the smaller of the two.

```
def angle(hour, minute):
    h = 30 * hour + 6 * (minute / 12)
    m = 6 * minute

    return min(abs(h - m), 360 - abs(h - m))
```

For the bonus question, it may seem as though we can apply this function to each hour and minute combination in a day, and collect all the times when the angle is zero.

However, most of the time when the two hands intersect, it will not be when a minute hand is perfectly aligned on a tick. Instead, we can solve this mathematically, by setting the hour and minute variables above equal to each other and simplifying.

```
30 * hour + 6 * (minute / 12) = 6 * minute
30 * hour = 6 * minute - (1 / 2) * minute
30 * hour = (11 / 2) * minute
hour = (11 / 60) * minute
```

In other words, the two hands will overlap when the hours passed is exactly `11/60` of the minutes passed. With this formula, we can loop through each hour and determine the matching times.

```
def matching_hands():
    times = []

    for hour in range(11):
        minute = int((60 / 11) * hour)
        times.append((hour, minute))

    return times
```

For completeness, the times we will find are as follows: `12:00, 1:05, 2:10, 3:16, 4:21, 5:27, 6:32, 7:38, 8:43, 9:49, 10:54`.