How Far Can You Run Before Sundown?

1. You're participating in a trail run that ends at sundown at 7 p.m. There are four loops: 1 mile, 3 miles, 3.5 miles, and 4.5 miles. After completing any given loop, you are randomly assigned another loop to run—this next loop could be the same as the previous one you just ran, or it could be one of the other three. Being assigned your next loop doesn't take a meaningful amount of time; assume all your time is spent running.

Your "score" in the race is the total distance you run among all completed loops you are assigned. If you're still out on a loop at 7 p.m., any completed distance on that loop does not count toward your score!

It is now 5:55 p.m. and you have just completed a loop. So far, you've been running 10-minute miles the whole way. You'll maintain that pace until 7 p.m.

On average, what score can you expect to earn between 5:55 p.m. and 7 p.m.?

Solution:

Let $\mathscr{T}(t)$ be the expected completed running time you obtain when starting a loop with t minutes left. From 5:55-7:00, there are 65 minutes left. Since we take 10 minutes for every mile, the expected number of miles we run in this time is:

$$\mathbb{E}[\text{Miles}] = \frac{\mathscr{T}(t)}{10}$$

Each loop will take times: $\mathscr{L} \in \{10, 30, 35, 45\}$, and they are chosen uniformly at random. For t minutes on the clock, we say that if $\mathscr{L} \leq t$, we collect $\mathscr{L}$ minutes, and continue racing for $t - \mathscr{L}$ minutes. If $\mathscr{L} > t$, then we will not complete the loop, and we will score 0 for this. Thus, using the law of total expectation, we can express $\mathscr{T}(t)$ recursively as:

$$\mathscr{T}(t) = \mathbb{E}\left[\mathbb{1}_{\{\mathscr{L} \leq t\}} \left(\mathscr{L} + \mathscr{T}(t - \mathscr{L})\right)\right] = \sum_{\mathscr{L} \in \{10,30,35,45\}} \mathbb{P}(\mathscr{L}) \mathbb{1}_{\mathscr{L} \leq t} \left(\mathscr{L} + \mathscr{T}(t - \mathscr{L})\right)$$

And since the times from $\mathscr{L}$ are chosen uniformly at random, we have:

$$\mathscr{T}(t) = \frac{1}{4} \sum_{\mathscr{L} \in \{10,30,35,45\}} \mathbb{1}_{\mathscr{L} \leq t} \left(\mathscr{L} + \mathscr{T}(t - \mathscr{L})\right)$$

We will always complete loops at times that are multiples of 5, so we only need to check for times $t = 0, 5, 10, \ldots$ and so on. Some useful base cases: $\mathscr{T}(0) = \mathscr{T}(5) = 0$ obviously, and too. As an example, if there was less than 30 minutes remaining, only $\mathscr{L} = 10$ minute loops count, and so we compute

$$
\begin{aligned}
\mathscr{T}(10) &= \frac{1}{4}(10 + \mathscr{T}(0)) = 2.5 \\
\mathscr{T}(15) &= \frac{1}{4}(10 + \mathscr{T}(5)) = 2.5 \\
\mathscr{T}(20) &= \frac{1}{4}(10 + \mathscr{T}(10)) = 3.125 \\
&\vdots
\end{aligned}
$$

And I'm already bored. We would continue by adding subsequent terms in our summation for different $\mathscr{L}$ value times allowed, but this is a long process to do by hand. Below is small python program that will use this recursive formula to find the desired output $\mathscr{T}(65)$, and then also print the expected miles we run (by dividing $\mathscr{T}(65)$ by 10).

```python
from functools import lru_cache

LOOP_TIMES = (10, 30, 35, 45)
P = 1.0 / 4.0  # Probability a given loop is selected

@lru_cache(None)
def expected_completion_times(t: int) -> float:
    if t < min(LOOP_TIMES):
        return 0.0
    return P * sum((L + expected_completion_times(t - L)) if L
        <= t else 0.0 for L in LOOP_TIMES)

val_minutes = expected_completion_times(65)
val_miles = val_minutes / 10.0

print(f"T(65) = {val_minutes:.10f}")
print(f"Expected score = {val_miles:.10f} miles.")
```

And we get $\mathscr{T}(65) = 48.6645...$, and therefore our expected score is 4.866.., or rounded as $\boxed{4.87}$.