MC Control: Constant-alpha

In an earlier quiz (**Quiz: Incremental Mean**), you completed an algorithm that maintains a running estimate of the mean of a sequence of numbers $(x_1, x_2, \ldots, x_n)$. The `running_mean` function accepted a list of numbers `x` as input and returned a list `mean_values`, where `mean_values[k]` was the mean of `x[:k+1]`.

$$
\begin{aligned}
&\mu \leftarrow 0 \\
&k \leftarrow 0 \\
&\textbf{While } k < n \\
&\quad k \leftarrow k + 1 \\
&\quad \mu \leftarrow \mu + \frac{1}{k}(x_k - \mu)
\end{aligned}
$$

When we adapted this algorithm for Monte Carlo control in the following concept (**MC Control: Policy Evaluation**), the sequence $(x_1, x_2, \ldots, x_n)$ corresponded to returns obtained after visiting the *same* state-action pair.

That said, the sampled returns (for the *same* state-action pair) likely corresponds to many *different* policies. This is because the control algorithm proceeds as a sequence of alternating evaluation and improvement steps, where the policy is improved after every episode of interaction. In particular, we discussed that returns sampled at later time steps likely correspond to policies that are more optimal.

With this in mind, it made sense to amend the policy evaluation step to instead use a constant step size, which we denoted by α in the previous video (**MC Control: Constant-alpha, Part 1**). This ensures that the agent primarily considers the most recently sampled returns when estimating the action-values and gradually forgets about returns in the distant past.

The analogous pseudocode (for taking a *forgetful* mean of a sequence $(x_1, x_2, \ldots, x_n)$) can be found below.

$$
\begin{aligned}
&\mu \leftarrow 0 \\
&k \leftarrow 0 \\
&\textbf{While } k < n
\end{aligned}
$$