Exercise 6.1

$$
\begin{aligned}
G_t - V_t(S_t) &= R_{t+1} + \gamma G_{t+1} - V_t(S_t) + \gamma V_t(S_{t+1}) - \gamma V_t(S_{t+1}) \\
&= \delta_t + \gamma(G_{t+1} - V_t(S_{t+1})) \\
&= \delta_t + \gamma(G_{t+1} - V_{t+1}(S_{t+1})) + \gamma(V_{t+1}(S_{t+1}) - V_t(S_{t+1})) \\
&= \sum_{k=t}^{T-1} \gamma^{k-t}\delta_k + \sum_{k=t}^{T-1} \gamma^{k-t+1}(V_{k+1}(S_{k+1}) - V_k(S_{k+1}))
\end{aligned} \tag{1}
$$

Exercise 7.1

$$
\begin{aligned}
G_t^{(n)} - V(S_t) &= R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{n-1}R_{t+n} + \gamma^n V(S_{t+n}) - V(S_t) \\
&= R_{t+1} + \gamma V(S_{t+1}) - V(S_t) \\
&\quad + \gamma(R_{t+2} + \gamma V(S_{t+2}) - V(S_{t+1})) \\
&\quad \ldots \\
&\quad + \gamma^{n-1}(R_{t+n} + \gamma V(S_{t+n}) - V(S_{t+n-1})) \\
&= \sum_{k=t}^{t+n-1} \gamma^{k-t}\delta_k
\end{aligned} \tag{2}
$$

Exercise 7.2

$$
\begin{aligned}
G_t^{(n)} - V_t(S_t) &= R_{t+1} + \gamma R_{t+2} + \cdots + \gamma^{n-1}R_{t+n} + \gamma^n V_t(S_{t+n}) - V_t(S_t) \\
&= (R_{t+1} + \gamma V_t(S_{t+1}) - V_t(S_t)) - \gamma V_t(S_{t+1}) \\
&\quad + \gamma(R_{t+2} + \gamma V_{t+1}(S_{t+2}) - V_{t+1}(S_{t+1})) + \gamma V_{t+1}(S_{t+1}) - \gamma^2 V_{t+1}(S_{t+2}) \\
&\quad + \gamma^2(R_{t+3} + \gamma V_{t+2}(S_{t+3}) - V_{t+2}(S_{t+2})) + \gamma^2 V_{t+2}(S_{t+2}) - \gamma^3 V_{t+2}(S_{t+3}) \\
&\quad \ldots \\
&\quad + \gamma^{n-1}(R_{t+n} + \gamma V_{t+n-1}(S_{t+n}) - V_{t+n-1}(S_{t+n-1})) + \gamma^{n-1}V_{t+n-1}(S_{t+n-1}) - \gamma^n V_{t+n-1}(S_{t+n}) \\
&\quad + \gamma^n V_t(S_{t+n}) \\
&= \sum_{k=t}^{t+n-1} \gamma^{k-t}\delta_k + \sum_{k=1}^{n} \gamma^k(V_{t+k\%n}(S_{t+k}) - V_{t+k-1}(S_{t+k}))
\end{aligned} \tag{3}
$$

From (3) we can learn that the difference between the true n-step TD error $G_t^{(n)} - V_t(S_t)$ and the sum of n TD errors $\sum_{k=t}^{t+n-1} \gamma^{k-t}\delta_k$ is $Diff = \sum_{k=1}^{n} \gamma^k(V_{t+k\%n}(S_{t+k}) - V_{t+k-1}(S_{t+k}))$. At update time $t+k-1$, TD algorithm will update the value for S_{t+k-1}, then we get V_{t+k} from V_{t+k-1}. So in most cases, $V_{t+k}(S_{t+k})$ is the same as $V_{t+k-1}(S_{t+k})$, unless S_{t+k} is the same as S_{t+k-1}. The last term of $Diff$ is slightly different, $V_t(S_{t+n})$ will be different from $V_{t+n-1}(S_{t+n})$ if $S_{t+n} \in \{S_t, S_{t+1}, \ldots, S_{t+n-2}\}$.

To collect the sum of n TD errors, at each update time k, we make a copy V_k' of V_k, then perform n TD updates to V_k' and use the cumulative TD errors to update V_k.

We use the 19-state random walk task to benchmark the two algorithms. However, as discussed above, we need an extra action $STAY$. So in our experiment, we have three actions in total, $\{LEFT, STAY, RIGHT\}$ w.p. $\{0.25, 0.5, 0.25\}$. Even though we add an action $STAY$ and give it high probability to make $S_{t+k} = S_{t+k-1}$ happen more often, there still isn't siginificant difference between the performance of the two algorithms. The term $Diff$ contributes too little to the total error.