Problem 1:

$\boxed{1}$

Given, $(T^* V)(s) := \max_a \{R(s,a) + \gamma \sum_{s'} P(s'|s,a) V(s')\}$

for V', $a_s'^* = \arg\max_a \{R(s,a) + \gamma \sum_{s'} P(s'|s,a) V'(s')\}$

for V, $a_s^* = \arg\max_a \{R(s,a) + \gamma \sum_{s'} P(s'|s,a) V(s')\}$

let us consider,

for V, $a_s'^* = \{R(s,a) + \gamma \sum_{s'} P(s'|s,a) V(s')\}$

We get,

Value function V for a_s^* $\geq$ value function V for $a_s'^*$

~~Because~~ The $\arg\max_a$ of V' $(a_s'^*)$ ~~can~~ may or may not be the same action as $\arg\max_a$ of V (a_s^*) ~~. Since~~ because these actions are from the same action space.

Now, if $(T^* V')(s) \geq (T^* V)(s)$

then,

$$|(T^* V')(s) - (T^* V)(s)| = (T^* V')(s) - (T^* V)(s)$$

$$= R(s, a_s'^*) + \gamma \sum_{s'} P(s'|s, a_s'^*) V'(s') - \left(R(s, a_s^*) + \gamma \sum_{s'} P(s'|s, a_s^*) V(s') \right)$$

$$= R(s, a_s'^*) + \gamma \sum_{s'} P(s' | s, a_s'^*) V'(s') -$$
$$R(s, a_s^*) - \gamma \sum_{s'} P(s' | s, a_s^*) V(s')$$

$$\leq R(s, a_s'^*) + \gamma \sum_{s'} P(s' | s, a_s'^*) V'(s') -$$
$$R(s, a_s'^*) - \gamma \sum_{s'} P(s' | s, a_s'^*) V(s')$$
$$\left[\because a_s^* \geq a_s'^* \text{ for } V \right]$$

$$\leq 0 + \gamma \left\{ \sum_{s'} P(s' | s, a_s'^*) V'(s') - \sum_{s'} P(s' | s, a_s'^*) V(s') \right\}$$

$$\leq \gamma . \sum_{s'} P(s' | s, a_s'^*) . \{ V'(s') - V(s') \}$$

The s' under sigma belongs to S $\left(\sum_{s' \in S} \right)$. Sum of the transition probability for a single action $(a_s'^*)$ over all state space $s \in S$ is 1.

$$\leq \gamma . 1 \; \{ V'(s') - V(s') \}$$

this is true for every $s' \in S$. if we take $\max_{s' \in S}$.

We get,

$$\leq \gamma . \max_{s' \in S} \{V'(s') - V(s')\}$$

$$\leq \gamma . \| V' - V \|_\infty$$

So, $|(T^* V')(s) - (T^* V)(s)| \leq \gamma \| V' - V \|_\infty$

For the else cas, where, $(T^* V')(s) < (T^* V)(s)$

we can consider,

for V', $a_s^* = R(s,a) + \gamma \sum_{s'} P(s'|s,a) V'(s')$

and then roll-out the proof in similar way using.

$a_s'^*$ for V' $>$ a_s^* for V'

<u>Problem 3 (c):</u>

How iteration is affected:
When stochasticity is introduced, the number of iterations increases. In the deterministic condition, the policy iteration loops one time, and the value iteration loops seven times. Whereas in the stochastic condition, the policy iteration loops two times (double that of deterministic), and the value iteration loops 23 times (more than triple of deterministic).

How policy is affected:

In the deterministic condition, the policy yields the following state-action policies for both the policy iteration and value iteration.

0: Left 1: Down 2: Right 3: Up

Down	Right	Down	Left
Down	Left	Down	Left
Right	Down	Down	Left
Left	Right	Right	Left

But in stochastic conditions, the policy yields confusing state-action policies for policy iteration and value iteration. The policy and value iteration is different as well.

Policy iteration:

Down	Up	Left	Up
Left	Left	Left	Left

Up	Down	Left	Left
Left	Right	Down	Left

Value iteration:

Left	Up	Left	Up
Left	Left	Left	Left
Up	Down	Left	Left
Left	Right	Down	Left

```
###############################################
#  Results of Deterministic-4x4-FrozenLake-v0  #
###############################################

# Policy Iteration:

# Episode reward: 1.000000
# value function:  [0.59   0.656 0.729 0.656 0.656 0.    0.81  0.    0.729
0.81  0.9   0.
#  0.    0.9   1.    0.   ]
# Policy:  [1 2 1 0 1 0 1 0 2 1 1 0 0 2 2 0]
# Policy iteration count:  1
```

```
#
--------------------------------------------------------------------------
--------------
# Value Iteration:

# Episode reward: 1.000000
# value function:  [0.59  0.656 0.729 0.656 0.656 0.    0.81  0.    0.729
0.81  0.9   0.
#  0.    0.9   1.    0.   ]
# Policy:  [1 2 1 0 1 0 1 0 2 1 1 0 0 2 2 0]
# Value iteration count:  7

################################################
#   Results of Stochastic-4x4-FrozenLake-v0    #
################################################

# Policy Iteration:

# Episode reward: 1.000000
# value function:  [0.021 0.021 0.039 0.019 0.03  0.    0.071 0.    0.072
0.156 0.197 0.
#  0.    0.251 0.431 0.   ]
# Policy:  [1 3 0 3 0 0 0 0 3 1 0 0 0 2 1 0]
# Policy iteration count:  2

#
--------------------------------------------------------------------------
--------------


# Value Iteration:

# Episode reward: 0.000000
# value function:  [0.064 0.058 0.072 0.054 0.088 0.    0.111 0.    0.143
0.246 0.299 0.
#  0.    0.379 0.639 0.   ]
# Policy:  [0 3 0 3 0 0 0 0 3 1 0 0 0 2 1 0]
# Value iteration count:  23
```