

## Homework 3 Answers

### Answers

1. What are the advantages and disadvantages of the 256 bit word length in the EVM?

The 256 bit word length allows Keccak-256 hashing algorithm to run efficiently.

2. What would happen if the implementation of a precompiled contract varied between Ethereum clients?

The different clients would not come to consensus and you get end up with forks of the network.

3. Using Remix write a simple contract that uses a memory variable, then using the debugger step through the function and inspect the memory.

The screenshot displays the Remix IDE interface. On the left, the 'DEPLOY & RUN TRANSACTIONS' sidebar is visible, showing a 'Deploy' button, a 'Publish to IPFS' checkbox, and a 'Deployed Contracts' section listing 'EEELEVEL\_2\_SOLUTION AT 0XD91...'. The main editor shows a Solidity contract named 'eeeLevel\_2\_Solution' with a 'solution' function. The function takes a 'uint256 a' parameter and returns a 'uint256' value. It declares a 'uint256 s' variable and a 'uint256[1] memory b' array. The function body assigns 'b = [a];' and 's = b[0];'. A gas cost notification indicates 'PUSH1 costs 3 gas - this line costs 22166 gas - 28723 gas left'. The bottom status bar shows a successful transaction: '[vm] from: 0x5B3...eddC4 to: eeeLevel\_2\_Solution.solution(uint256) 0xd91...39138 value: 0 wei data: 0xdb5...00005 Logs: 0 hash: 0x437...69318'. A 'Debug' button is also present.

```
1 // SPDX-License-Identifier: UNLICENSED
2 pragma solidity 0.8.17;
3
4 contract eeeLevel_2_Solution {
5
6     uint256 s;
7
8     function solution(uint256 a) external {
9         uint256[1] memory b;
10        b = [a];
11        s = b[0];
12    }
13
14 }
```

The screenshot shows the Solidity IDE with the following components:

- Debugger Panel (Left):**
  - Use generated sources (Solidity >= v0.7.2):** Checked.
  - Stop debugging:** Button.
  - Memory View:**
    - 0x0: 0x00000000000000000000000000000000
    - 0x20: 0x00000000000000000000000000000000
    - 0x40: 0x00000000000000000000000000000000
    - 0x60: 0x00000000000000000000000000000000
    - 0x80: 0x00000000000000000000000000000000
    - 0xa0: 0x00000000000000000000000000000000
- Code Editor (Center):**

```

1 // SPDX-License-Identifier: UNLICENSED
2 pragma solidity 0.8.17;
3
4 contract eeeLevel_2_Solution {
5
6     uint256 s;
7
8     function solution(uint256 a) external {
9         uint256[1] memory b;
10        b = [a];
11        s = b[0]; // POP costs 2 gas - this line costs 22166 gas - 6600 gas left
12    }
13
14 }

```
- Transactions Panel (Bottom):**
  - Transaction 0: listen on all transactions
  - Search with transaction hash or address
  - Transaction: `transact to eeeLevel_2_Solution.solution pending ...`
  - Log: `[vm] from: 0x5B3...eddc4 to: eeeLevel_2_Solution.solution(uint256) 0xd91...39138`