CS-3812 HW 03

In=15th April 11:45pm; Out=6th April

*You may partner with up to four others (**a group of max 5**) to submit a single write-up in a single group. We encourage discussion with other students in class, even if they are not in your group. Each student must build a good understanding of all the questions even if they discuss and collaborate with others. Merely dividing the problems among group members and putting those together before submission will severely impact your learning, and we **strongly advice against it**. Furthermore, **blatant copying** from online or other resources is **forbidden**. If there are confusions or questions, post those on Piazza or see your TA or Instructor.*

***Submission Instructions:** You must submit a PDF on LMS in the appropriate tab. This includes hand-written content (which you may scan and upload). There will be NO late days or extensions. Please write down the names and roll numbers on the front page. Make only one submission per group. We will NOT accept any hard-copy submissions.*

Question 1 [6+4 marks]

a) After understanding the differences between the Bitcoin and Ethereum models, which one of the two do you regard as objectively better? You are encouraged to take your time and answer this question in a discussion-like format mentioning the similarities and differences between both models, weighing the pros and cons that arise, and finally drawing at the conclusion that results in your answer.

Solution: Mentioning the differences in protocol governance and reference models in both earns 2 marks. Mentioning the similarities in transaction processing etc, earns 2 marks. Concluding the better model objectively with a reasonable explanation earns the final 2 marks.

b) If you answered **Ethereum**, do you not think that there is a concern about accountability and centralization of power since a small number of people have significant influence over Ethereum's direction? If you answered **Bitcoin**, do you not think that there is a concern about the centralization of mining power since mining is increasingly centralized with a small number of mining pools accounting for a significant proportion of Bitcoin's mining rate?

Solution: 2 marks for acknowledging the existence of centralization and why it may be necessary or what benefits it brings. 2 marks for any way in which it has been mitigated e.g. on and off-chain governance or why centralization may be necessary.

Question 2 [3+4+3 marks]

a) What is the role of the Ethereum Virtual Machine (EVM)? **Solution:** - Environment for smart contracts to run - Stores and executes code - Enforces gas model requiring payments in ether for code execution

b) What is gas in terms of Ethereum transactions, why is it used, and how does its price and limit affect smart contract execution? **Solution:** - A sort of resource for computation - Prevents infinite computation - gas price high, higher payment needed and vice versa - if gas limit is low, transaction need executed completely

c) What are *state tries* and how does Ethereum use them to manage and store information about the network state? There is definitely an advantage to state tries but can you identify a disadvantage too? **Solution:** - Representation of the world state and track only the changes to the state - Better and more efficient scaling - Immutable - Can be slow and resource-intensive with growing network size - if gas limit low, transaction need executed completely

Question 3 [3+2 marks]

a) What is Ethereum burning and how does it work? **Solution:** Ethereum burning refers to the removal of Ethereum (ETH) from circulation. This process was introduced as part of the EIP 1559 which resulted in a base fee for each transaction on the Ethereum network. Instead of giving that fee to Ethereum miners, the network destroys the associated ETH.The idea behind Ethereum burning is to reduce the total supply of ETH, which unlike Bitcoin, has no eventual cap.

b) How has ETH burning affected the supply of ETH? **Solution:** Nowadays the burn rate of Ethereum is far higher than staking rewards as such the supply is reducing and ETH is becoming a deflationary asset.

Question 4 [8 marks]

a) What is Max and target gas limit? **Solution:** - Max gas limit: 30 million gas units - Target gas limit: 15 million gas units

b) How is the target gas limit maintained? **Solution:** The network adjusts the block base fee to maintain a target block gas limit of 15 million units If the last block used more than 50% gas, base fee for the next block will increase on the other hand if the last block used less than 50% gas, base fee for the next block will decrease. The percentage change in the gas fee from one block to the next can't be more than 12.5%

c) How is the block base fee calculated? **Solution:** Base fee for the next block = Base fee for current block * [1 + (% gas used in current block – 50) / 400)]

d) Base fee for block 143968 = 12.59346 Gwei and % Gas used is 5.382%. Calculate the base fee for the next block. **Solution:** 11.189

Question 5 [7+3 marks]

a) We know that the 'base fee per gas' in a block depends in the total gas in the previous block. Let's suppose that the 'base fee per gas' in block number 100,000 is 0.01 Gwei. Assuming that the gas percentage remains at 90% for **all** subsequent blocks, which block

will have a 'base fee per gas' of 10,000 Gwei?

Solution:

The increase in fee per block depends on the target gas percentage. This will be:

$$\frac{90-50}{50} \times 100\% = 80\%$$

This means the fee will increase by $1 + \frac{80}{800} = 1.1$ times per block.

The fee increase is thus 1.1 times the previous block.

$$0.01 \times 1.1^x = 10000$$

$$\Rightarrow x \approx 144.95$$

so in the 145th block after 100,000, the gas fee will rise above 10,000 Gwei.

b) What effect would the use of this mechanism (the one that dictates 'base fee per gas') have on transaction fees in the long term? Why?

Solution:

It discourages the Ethereum network from picking transactions that use a lot of gas. This helps keep transaction fees lower.

Question 6 [2+2+4+4 marks] Consider the Ethereum blockchain. Suppose that the gas used in the current block is 12 Million and the base fee per gas is 85.21 Gwei.

a) What is the percentage gas used? **Solution:**

% gas used $= (12/30) * 100 = 40$

b) What is the percentage gas target? **Solution:**

% gas target $= ((12-15)/15) * 100 = -20\%$

c) How much ETH was burnt in this block? **Solution:**

Burnt Gas $= 12000000 * 85.21 = 1022520000$

d) What will be the base fee per gas of the next block mined? **Solution:**

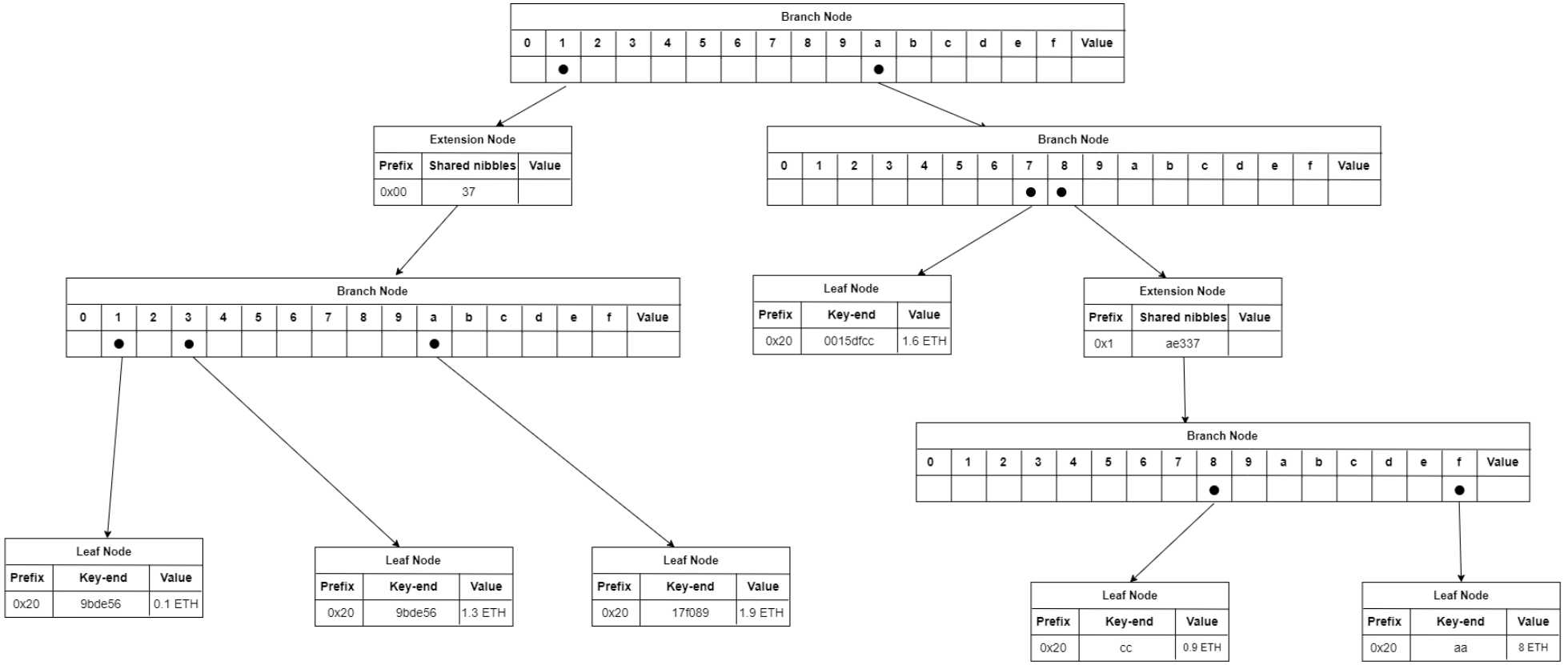
Base fee for next block $= 85.21 * [1 + (40-50)/400] = 83.08$

Question 7 [7+8+4] marks In class we learnt about constructing a Merkel Patricia tree from the Ethereum world state. Consider the following simplified world state of Ethereum to answer this question. (Ahsan)

Simplified world state										
Address										Value
a	7	0	0	1	5	d	f	c	c	1.6 ETH
a	8	a	e	3	3	7	8	c	c	0.9 ETH
a	8	a	e	3	3	7	f	a	a	8 ETH
1	3	7	a	1	7	f	0	8	9	1.9 ETH
1	3	7	1	9	b	d	e	5	6	0.1 ETH
1	3	7	3	9	b	d	e	5	6	1.3 ETH

a) Construct an MPT from the given simplified world state. You should draw all the nodes with their appropriate types and prefixes mentioned. You don't need to calculate the value field (with the hash), just leave it empty.

Solution:


Branch Node
0 1 2 3 4 5 6 7 8 9 a b c d e f Value
Extension Node
Prefix Shared nibbles Value
0x00 37
Branch Node
0 1 2 3 4 5 6 7 8 9 a b c d e f Value
Leaf Node
Prefix Key-end Value
0x20 0015dfcc 1.6 ETH
Extension Node
Prefix Shared nibbles Value
0x1 ae337
Branch Node
0 1 2 3 4 5 6 7 8 9 a b c d e f Value
Leaf Node
Prefix Key-end Value
0x20 9bde56 0.1 ETH
Leaf Node
Prefix Key-end Value
0x20 9bde56 1.3 ETH
Leaf Node
Prefix Key-end Value
0x20 17f089 1.9 ETH
Branch Node
0 1 2 3 4 5 6 7 8 9 a b c d e f Value
Leaf Node
Prefix Key-end Value
0x20 cc 0.9 ETH
Leaf Node
Prefix Key-end Value
0x20 aa 8 ETH

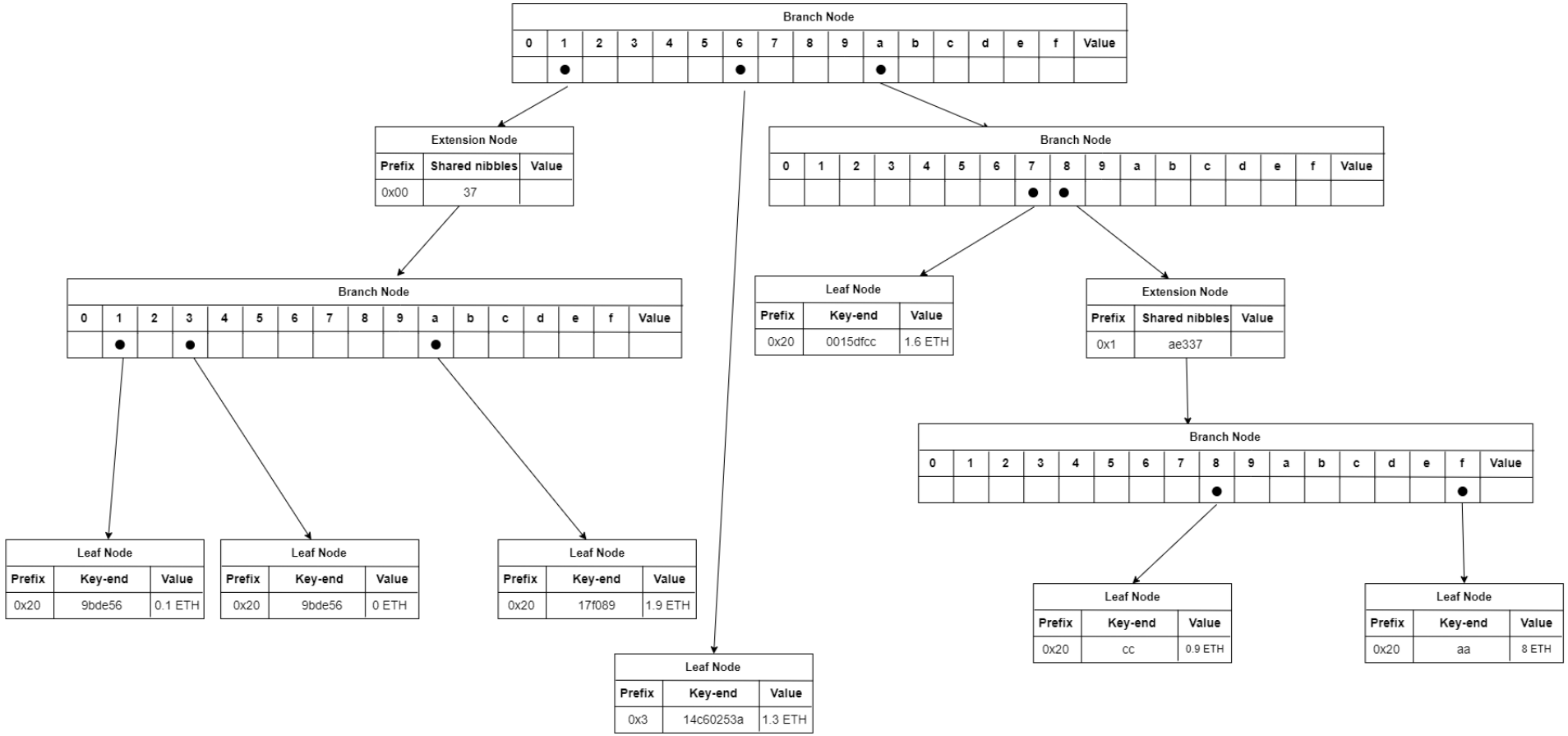
b) Suppose a new Ethereum account (address = 0xa70015efcc) is created with an initial balance of zero ETH. Then account 0x13739bde56 sends 1.3 ETH to this new account. Draw a new MPT with this updated information.

Recall that computing the address for an EoA involves calculating the hash of the EoA's public key and taking the right-most 160 bits. The key for the MPT is *then* generated by taking the hash of the address.

Solution:

Keccak 256 bit hash = 0x913becd8c475e1404ad7b331614c60253afdf36c583b328a1b06df3de5c557b0

Note: If students have used some other hash function it is also acceptable. They should just mention the final 10 nibbles they used and inserted it accordingly. Final nibbles = 614c60253a


Branch Node
0 1 2 3 4 5 6 7 8 9 a b c d e f Value
Extension Node
Prefix Shared nibbles Value
0x00 37
Branch Node
0 1 2 3 4 5 6 7 8 9 a b c d e f Value
Leaf Node
Prefix Key-end Value
0x20 0015dfcc 1.6 ETH
Extension Node
Prefix Shared nibbles Value
0x1 ae337
Branch Node
0 1 2 3 4 5 6 7 8 9 a b c d e f Value
Leaf Node
Prefix Key-end Value
0x20 9bde56 0.1 ETH
Leaf Node
Prefix Key-end Value
0x20 9bde56 0 ETH
Leaf Node
Prefix Key-end Value
0x20 17f089 1.9 ETH
Leaf Node
Prefix Key-end Value
0x3 14c60253a 1.3 ETH
Branch Node
0 1 2 3 4 5 6 7 8 9 a b c d e f Value
Leaf Node
Prefix Key-end Value
0x20 cc 0.9 ETH
Leaf Node
Prefix Key-end Value
0x20 aa 8 ETH

c) *Precisely* explain why the key is generated by hashing the address and not taking the address itself. *Hint: The length of the key vs address does not matter.*

Solution: If we were just using the address without taking the hash then an attacker can exploit by making multiple accounts and making the state trie grow in one direction by adding new accounts thus decreasing the performance of Ethereum. Hashing gives it a degree of randomness thus making sure the above exploit is not possible.

Question 8 [12 marks] This task requires identifying the appropriate function type for given Solidity functions. You need to select one scope and one behaviour type from the provided options: internal, external, public, private, pure, view, and payable. In some cases, there may be multiple correct function types, but you should choose the **most** restrictive one.

a) A function that takes an input a and sets the value of the state variable myMoney equals to input a. This function can only be called by the smart contract itself or any other contracts deriving from it.

Solution: Internal

b) A function that returns the state variable b. This may be called by EOAs.

Solution: public, view

c) A function that is solely called by a function myFunc() (which is in the same contract), takes an input x and returns (myNum + 32) % x where myNum is a state variable.

Solution: private/internal, view

d) An external contract called ExternalCont creates a private state variable obj of type ContractOne (a contract available to us) to use its external function called getmultiply(). The function getmultiply() in ContractOne takes two numbers num1 and num2 as input and returns their product.

Solution: external pure

e)
```
contract Test {
   function getResult(uint a, uint b) <enter type here> returns(uint product, uint sum){
      product = a * b;
      sum = a + b;
   }
}
```

What is the type of this function (Choose one scope and one behaviour type)?

Solution: private pure

f)
```
contract Test {
   uint number1 = 10;
```

```
    function getSum(uint number2, uint number3) <enter type here> returns(uint) {
        uint sum = number1 + number2 + number3;
        return sum;
    }
}
```

What is the type of this function (Choose one scope and one behavior type)?

Solution: private view

Question 9 [5 marks] Suppose you are a miner trying to mine blocks for the Ethereum blockchain. You have a multitude of transactions in the mempool to pick and form a block with. Assume that the maximum gas per block is 30 Million and the 'base fee per gas' is 65 Gwei. You have three varieties of transactions to choose from, transaction variety A, variety B and variety C. Assume that only variety A, variety B and variety C exist and are abundant in the mempool.

Variety A has the following properties:

- This is a basic transaction transferring some ETH.
- a gas consumption of 42,000
- max priority fee per gas = 11 Gwei.
- max fee per gas = 100 Gwei.

Variety B has the following properties:

- This transaction does something more complicated, such as handling an NFT.
- a gas consumption of 156,000
- a max priority fee per gas = 14 Gwei.
- max fee per gas = 90 Gwei.

Variety C has the following properties:

- This transaction creates basic smart contract.
- a gas consumption of 19,000
- a max priority fee per gas = 18 Gwei.
- max fee per gas = 95 Gwei.

Which Variety will you include in your block?

Solution: Ans:
Let's consider Variety A:

- The total number of transactions per block (from the block gas limit) is $\frac{30Million}{42000} \sim 714$
- The amount of Gwei we earn is $11 \times 42,000 \times 714 = 330$ Million Gwei

Let's consider Variety B:

- The total number of transactions per block (from the block gas limit) is $\frac{30Million}{156000} \sim 192$

- The amount of Gwei we earn is $14 \times 156000 \times 192 = 420$ Million Gwei

Let's consider Variety C:

- The total number of transactions per block (from the block gas limit) is $\frac{30Million}{19000} \sim 1579$
- The amount of Gwei we earn is $18 \times 19000 \times 1579 = 540$ Million Gwei

We pick transaction Variety C as it is providing more Gwei per block.

Question 10 [20 marks] Bitcoin relied on the property that *anyone* with a computer could join the network and mine a block in-order to ensure decentralization. However, due to the huge size of the bitcoin blockchain and the steep computational requirements, this property is considerably weaker.

a) Would you consider this a strong property of the Ethereum blockchain three years ago?

Yes, anyone could mine Ethereum three years ago. As such, it had a strong decentralization property.

After "The Merge", Ethereum switched to Proof-Of-Stake. As such, anyone with access to 32 Ether can become a part of the consensus mechanism.

b) Would you consider current-day Ethereum to have the strong property (mentioned above)?

No, in order to become part of the consensus mechanism, you must already own some Ethers. This makes a *new* blockchain inherently weaker as it encourages centralization. This is not the case with ethereum, however.

Let's do a thought experiment in which we evaluate the tendency of Proof-Of-Stake to centralize. Consider that there are *only* three validators in an entire blockchain network when block number 100,000 is appended to the blockchain. The stake of the first validator is 1 Eth, the stake of the second validator is 2 Eth and the stake of the third validator is 3 Eth. Given these major assumptions:

- A validator is randomly chosen to pick a block (based on their stake).
- The Ethers staked by a validator is always equal to the amount of Ethers in their account.
- That there is no minimum amount of Ethers staked.
- Successfully becoming the validator selected to append a block earns you 0.1 Eth.
- There is no other entity in the network.

c) Which validator is *most* likely to become selected for block number 100,001?

Solution: The third validator.

d) Consider block number 100,002. Which validator is the most likely to be selected? Show your working.

Solution: We have to calculate the expected Ethereum available to everyone after block # 100,001 and the chance that a validator has that much Ethereum.

For Validator 1, the probability is $\frac{1}{1+2+3} = \frac{1}{6}$ and the value is $1 + 0.1 = 1.1$
For Validator 2, the probability is $\frac{2}{1+2+3} = \frac{2}{6}$ and the value is $2 + 0.1 = 2.1$
For Validator 3, the probability is $\frac{3}{1+2+3} = \frac{3}{6}$ and the value is $2 + 0.1 = 3.1$

In every scenario, the third validator is most likely to be selected. This seems to imply that Ethereum has a tendency for centralization as the validator with the most Ethereum is likely to remain as such.

e) Which validator would be most likely to be selected for block number 200,000? Do you think Proof-Of-Stake has a tendency to centralize? *Hint: It will be impossible to calculate the exact probabilities by hand. You could either mathematically attempt this or write a simple script in Python to run a simulation. This script is not longer than 8-9 lines and may use numpy.random.choice().*

Solution: With the wonders of the digital age, we don't have to do math. If we write a script in python to see this, we notice that the *ratio* of Ethereum belonging to each validator remains the same. This implies that Ethereum does not have a tendency to centralize. Infact, if a pool sells its Ethereum rewards, then it is possible that Proof-Of-Stake encourages decentralization and discourages hoarding in the long run. Of course this is a simple example and doesn't consider a large number of validators, the variation in block reward and the fact that block rewards can not be used instantly etc.

Question 11 [6 marks] Let's suppose that block number 394151104 is Epoch number 12317222.

a) If all the blocks in Epoch number 12317222 are present and the first block in Epoch number 12317223 is absent, which block number is the checkpoint for Epoch number 12317223?

Solution: There are two check points. For this epoch, the source would be 394151104 and target would be 394151104+31 = 394151135.

b) Assuming that a block is generated in exactly 12 seconds, how long will it take for block number 394155012 to be finalized? Assume that block number 394151104 was generated at t=0.

Solution: It will need two checkpoints beyond it. So we will need 32 + 32 + 28 blocks infront of it, each taking 12 seconds. This will be $87 \times 12 = 1104$ seconds.

Question 12 [8 marks] Consider that a Liquidity Pool has 200 Bitcoin and 430000 Dogecoin.

a) What would the price of Dogecoin be in BTC (according to the pool)?

Solution: 1 dogecoin will be $\frac{200}{430000} \approx 0.000465$ Dogecoin

b) Suppose that the price of BTC drops to 2000 Dogecoin in the real world. Assuming that people do not supply Dogecoin to the LP, how much Bitcoin will they provide to the LP before the price equalizes?

Solution: The price inside the LP is $\frac{430000}{200} = 2150$.

Let $y = 200$ and $x = 430,000$. We want to increase y until $\frac{x}{y} \approx 2000$.

For each (small) Δy increase in y, the corresponding increase in x is $-\Delta y \times \frac{x}{y}$.

If we write a small script to solve this, we get x $\approx$ 414728 and y $\approx$ 207.36. So they give approximately 7.4 BTC to the LP.