Excercise 1 - Build a hash function from compression function (10 Punkte)

Consider the compression function $f : B^8 \to B^4$ given by
$f(b_0b_1b_2b_3b_4b_5b_6b_7) = e_0e_1e_2e_3$
where $e_i = b_{2i} \oplus b_{2i+1}$
Let further $x = 10111101010001010111$.
Apply the Merkle-Damgaard procedure to construct a hash function $h : B^* \to B4$ from f and calculate $h(x)$.

- Explain each step in this calculation.
- Give a collision of the hash function h.

Excercise 2 - Bloom filter - understanding (4 Punkte)

Consider a Bloom filter after entering all elements of a set M. What is the output of the check(x) operation of the Bloom filter in the following cases?

1. Suppose that $x \in M$.
2. Suppose that $x \notin M$.

Excercise 3 - Bloom filter - using (20 Punkte)

Let $m = 15$ and consider the following hash functions
$h_1(x) = (17x + 1)\%15$
$h_2(x) = (11x + 12)\%15$
$h_3(x) = (13x + 6)\%15$
Enter the numbers 17, 42, 21 into the empty Bloom filter with these parameters.
Write down the resulting bitvector using the format $BITVECTOR$: 000000000000000 (change 0 to 1 for bits that are set, the leftmost bit has index 0).
For each of the numbers 4, 5, 6, 7, 8, 9 determine whether the filter contains them! Write your answers in the form 4:YES or 4:NO (if 4 is/is not contained) etc up to 9:YES or 9:NO.
Separate answers by a single space, but do not use spaces inside an answer!
The order of the answers doesn't matter.

Solution:
BITVECTOR:001111000110110 für 8 Punkte
4:NO für 2 Punkte
5:NO für 2 Punkte
6:YES für 2 Punkte
7:NO für 2 Punkte
8:YES für 2 Punkte
9:NO für 2 Punkte

Excercise 4 - Key reconstruction from two fragments (10 Punkte)

Alice has distributed fragments of her secret key using the linear interpolation scheme, where the key can be reconstructed from any two fragments. Her calculations are performed modulo 3947. Max obtained the following two fragments

- (5, 276)
- (250, 889)

Help Bob to calculate the secret and the random number used to hide the secret.

Write your answers in the format SECRET1111 and RANDOM1111 where you replace 1111 by the respective answer. Separate by spaces. Use exactly four digits. Fill in leading zeroes if necessary.

Solution:
SECRET1069 RANDOM2999

Excercise 5 - Mining Strategies (15 Punkte)

1. Explain the concepts proof-of-work and proof-of-stake.
2. Discuss the advantages and disadvantages of POW and POS.
3. Explain the motivation behind the design of the SCRYPT hash function.

Excercise 6 - Contracts Ethereum vs Tezos (10 Punkte)

A contract on Ethereum/Solidity can directly transfer money or invoke a method in another contract. In contrast, in contract on Tezos/Michelson cannot directly transfer money or invoke another contract.

Explain how a Michelson contract can initiate a transfer. Give some reasons why Tezos/Michelson avoids direct transfers.

Excercise 7 - Michelson Typing (10 Punkte)

Each sequence of instructions in Michelson has a type that indicates which elements it expects on the stack and what kind of elements it leaves behind. Consider the following sequence of instructions

0: DUP;
1: CAR;
2: DIP(CDR);
3: CONCAT
4:

Give valid stack typings in front of each instruction in the following sequence and after the sequence (that is, for each numbers such that the type of each instruction fits with your typings.

Hint: start with a valid stack typing at the end and push the information forward according to the type of the instruction. The numbers do not belong to the instruction sequence, but you may use them to refer to instructions in your solution.

Excercise 8 - Hashed Datastructures for Ethereum (20 Punkte)

Research the internet to find out about the data structure which is used by the Ethereum blockchain to store its state (e.g., account balances, state of the contracts). Let?s call this data structure ES. Some parts of ES are discussed in the lecture.

1. Give the official name of ES and explain those parts which are not discussed in the lecture.
2. Give an algorithm (in pseudocode) to insert data into an instance of ES.
3. How does Ethereum store maps in ES?

Excercise 9 - A lottery contract (15 Punkte)

Consider a contract for a lottery on the Ethereum blockchain. Each participant chooses a secret number. To avoid publishing the number prematurely, they submit the hash of the secret along with their stake to the contract. Once all bids are in, participants are asked to reveal their secrets by submitting their secret number and having the contract check that it hashes to the previously submitted hash. Once all secrets are revealed, the contract calculates the winner from the ?secret? numbers and transfers all collected stakes to the winner.

This outline of the contract has two problems. Identify the problems and explain how to change the contract to avoid the problems.

Excercise 10 - Solidity - Crowdfunding (60 Punkte)

Write a contract for crowdfunding on Ethereum using the Solidity language. The contract will be created with a funding goal and will start in an open state where deposits are accepted. The contract can be closed as soon as the funding goal is met. When the contract is closed (by having the owner call the method **close**, but this must not happen before the funding goal is reached), all deposits are transferred to the owner of the contract and no further interaction with the contract will be possible. Only the owner can invoke **close**.

While the contract is open,

- everyone can use the **pay_in** method (as often as they want) to deposit funds in the contract;
- everyone can use the **withdraw** method to get their total deposit back.

Your contract may have further private methods beyond the required public methods **close**, **pay_in**, withdraw, and the constructor.

Upload your solution as a solidity file with extension .sol

Excercise 11 - Michelson-coding (60 Punkte)

See attached file Tezos-Michelson-coding.pdf for the question

Excercise 12 - Examcoin (90 Punkte)

See the attached file examcoin.pdf for the question.

The attached examcoin.py contains a code template to get you started.