

Name+Surname: _____ Univ. Code: _____

Q1 - Let P be an EC point. What is the **minimum** number of EC operations necessary to compute [63]P? And more specifically which are these operations?

Q2 - Consider both commitments introduced in our classes (**Feldman and Pedersen**), and assume they “commit” a value x. Under which (eventually different) assumptions they can be considered secure?

Feldman Pedersen

- | | | |
|-----------------------|-----------------------|---|
| <input type="radio"/> | <input type="radio"/> | a) no specific assumptions |
| <input type="radio"/> | <input type="radio"/> | b) must use a large prime p in the modular exponentiations |
| <input type="radio"/> | <input type="radio"/> | c) require that the committed value x is drawn from a large space |
| <input type="radio"/> | <input type="radio"/> | d) both large prime p and x drawn from large space |

Q3 - A strong prime p is defined as:

- a) a prime number p much larger than usual
- b) a prime p such as $2p+1 = q$ is also prime
- c) a prime p such as $p = 2q+1$ and q is also prime
- d) a prime p such as the Euler $\phi(p)$ is also prime

Q4 - Describe the Boneh-Franklin Identity Based Encryption scheme, specifying in particular, i) how a message is encrypted, ii) how a message is decrypted, and iii) what is the private key used by the receiver.

Name+Surname: _____ Univ. Code: _____

Q5 - Consider an RSA digital signature based on a (2,2) secret sharing, and assume all following operations are based on modulo n, with n being the RSA parameter. The tag $H(m)^d$ is reconstructed by:

- a) Summing the tags constructed using the two shares
- b) Multiplying the tags constructed using the two shares
- c) Interpolating the tags constructed using the two shares using Lagrange coefficients
- d) Using a special approach proposed by Shoup.

Q6 - Assume arithmetic modulus 100. A Linear secret sharing scheme involving 3 parties is described by the following access control matrix:

A:	1	1	0
B:	0	1	1
C:	0	0	-1

Assume that the following shares are revealed:

- A → 51
B → 63
D → 11

What is the secret?

- a) 1 b) 3 c) 23 d) 25 e) 75 f) 77 g) 97 h) 99 i) another result = _____

Q7 - A same message M is RSA-encrypted using two different public keys $e_1 = 5$ and $e_2 = 7$, but same RSA modulus $n=143$. The two resulting ciphertexts are: $c_1=23$ and $c_2=4$. Decrypt the message applying the Common Modulus Attack (show the detailed computations required).

Just in case you need to rapidly compute inverses modulus 143, here a few ones:

$$x = \{4, 5, 7, 17, 20, 23, 29, 92\} \quad \rightarrow \quad x^1 \quad \text{mod} \quad 143 \quad = \\ \{36, 86, 41, 101, 93, 56, 74, 14\}$$

Name+Surname: _____

Univ. Code: _____

Q8 - A Shamir Secret Sharing scheme uses a non-prime modulus $p=55$ (if you need modular inverses see table on the right). Of the 5 participating parties P_1, \dots, P_5 , with respective x coordinates $x_i = \{1,2,3,4,5\}$, parties P_1 , P_3 and P_5 aim at reconstructing the secret.

- compute the Lagrange Interpolation coefficients for parties 1,3,5;
- Reconstruct the secret, assuming that the shares are:

$$P_1 \rightarrow 46$$

$$P_3 \rightarrow 51$$

$$P_5 \rightarrow 2$$

- Prove that the system is NOT unconditionally secure, by showing that the knowledge of the two shares P_3 and P_5 leak information about the secret – specifically, after knowing shares P_3 and P_5 which would be the only possible remaining secret values?

x	$1/x \bmod 55$
1	1
2	28
3	37
4	14
6	46
7	8
8	7
9	49
12	23
13	17
14	4
16	31
17	13
18	52
19	29
21	21
23	12
24	39
26	36
27	53
28	2
29	19
31	16
32	43
34	34
36	26
37	3
38	42
39	24
41	51
42	38
43	32
46	6
47	48
48	47
49	9
51	41
52	18
53	27
54	54

Name+Surname: _____ Univ. Code: _____

Q9 - Prove that any linear secret sharing scheme is homomorphic with respect to the sum operation.

Q10 – 1) Determine the access control matrix that implements the policy: $\pi = (A \cap B) \cup (C \cap D \cap E)$, and then 2) turn it into a linear secret sharing scheme, by computing the shares to assigned to the 5 parties (use modulus 100, share secret S=10, invent your own random values if/when necessary)