CS60094 Computational Number Theory

Reference

'Computational Number Theory' by Abhijit Das, CRC Press

http://cse.iitkgp.ac.in/~somindu

Allotted Class hours

Wed 11-11:55 — Interactive session

Thur 12-12:55 — Tutorial

Fri 8-8:55

meeting links on course page

3 recorded lectures / week — links on course page

Evaluation

3 Tests

2-3 Programming Assignments } CSE Moodle account

Term paper

NUMBER THEORY

- study of properties of natural numbers $\mathbb{N}$ and integers $\mathbb{Z}$

$\mathbb{Q}, \mathbb{R}, \mathbb{C}, \ldots$

- concrete engineering applications
- Cryptography, coding theory

SYLLABUS

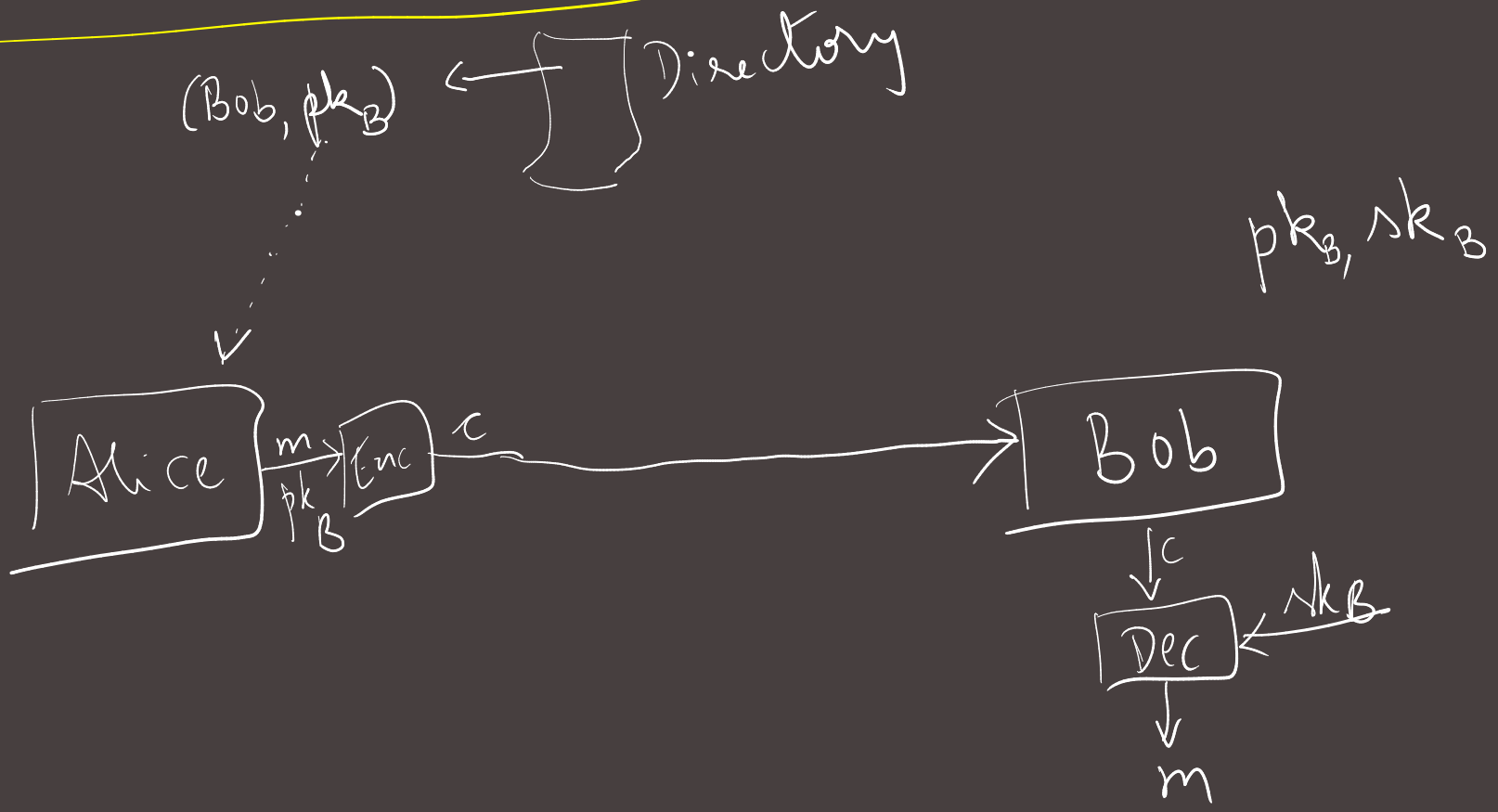
- Arithmetic of Integers
- Arithmetic of polynomials
- Finite fields
- Primality Testing Algorithms
- Integer Factorisation
- Applications (Public key Cryptography
  Coding theory

→ Elliptic curves
- Discrete logarithms

RSA Cryptosystem / Encryption System

↳ Rivest, Shamir, Adleman [1977]

Public-key Encryption



PKE

public ↑ pk, sk → secret

Key Generation : outputs pk, sk

Encryption : inputs message & pk
outputs ciphertext

Decryption : inputs ciphertext & sk
outputs message

Key Generation

→ Choose 2 distinct prime numbers p & q → primality testing

p, q: "large", "random", of the same magnitude

→ $N = pq$, $\phi(N) = (p-1)(q-1)$ → fast multiplication of "large" integers

↓ Euler's totient function

→ Choose $e \in \mathbb{Z}^*_{\phi(N)}$

compute $d = e^{-1} \pmod{\phi(N)}$ → Extended Euclidean algorithm

→ $pk = (N, e)$ $sk = d$

Encryption

$m \in \mathbb{Z}_N$

$c = m^e \pmod N$ → fast modular exponentiation

Decryption

$m = c^d \pmod N$

Correctness

$c^d = (m^e)^d = m^{ed} \equiv m \pmod N$

Security relies on the following assumptions

1. "Hard" to find d given (N, e)
2. "Hard" to find p, q given N

↓

integer factorisation