# Exercise 1.1

## Task 1

We want to show that the algorithm inside **CTXT** is not *Real-or-Random* secure.

Let the message space be

$$M = \{0, 1, \dots, p-1\}^\lambda.$$

We may choose any $p$ and $\lambda$.
The only condition we use is that the message is the all-zero vector.

### Example choice

$$m = [0,0,0,0,0], \qquad p = 5, \qquad \lambda = 5.$$

### Real cipher

For $i = 0, \dots, \lambda - 1, c[i] = m[i] \cdot k[i] \pmod p$.

Since $m[i] = 0$ for all $i$, it follows that $\forall i \;\; c[i] = 0$, so the ciphertext is always the all-zero vector $0^\lambda$.

### Random cipher

Each coordinate is sampled uniformly at random:
$c[i] \leftarrow \mathrm{Uniform}(\{0, 1, \dots, p-1\}), \quad i = 0, \dots, \lambda - 1.$

### Distinguishing advantage

Consider the adversary that outputs **real** iff the observed ciphertext is the all-zero vector $0^\lambda$.

- Under **Real**: $\Pr[\text{Adversary says real} \mid \text{Real}] = 1.$
- Under **Random**: $\Pr[\text{Adversary says real} \mid \text{Random}] = \Pr[c = 0^\lambda] = \left(\frac{1}{p}\right)^\lambda.$

Hence the distinguishing advantage is $\left|1 - \frac{1}{p^\lambda}\right| = 1 - \frac{1}{p^\lambda}.$

For $p = 5$ and $\lambda = 5, \Pr[c = 0^\lambda] = \left(\frac{1}{5}\right)^5 = \frac{1}{3125} \approx 0.00032$, so the adversary has distinguishing advantage $1 - \frac{1}{3125} \approx 0.99968.$

The overall guessing correctly probability is
$\Pr = \frac{1}{2} * 1 + \frac{1}{2} * (1 - \frac{1}{3125}) = \frac{1}{2} + \frac{1}{2} - \frac{1}{2 * 3125} = 1 - \frac{1}{6250} = 0.00016$

## Task 2

### Change in message space size

If $M = \{1, ...., p-1\}^{\lambda}$, then $C = \{1, ...., p-1\}^{\lambda}$ as well, as there is no number between $1$ and $p-1$ that would result in modulo 0 for $p = 5$.

---

# Exercise 1.2

Given a modified Fiestel construction where the round function is defined as:

$$MF(\vec{k}, \vec{x}) = (\vec{x}_1)||F(\vec{k}, \vec{x}_1) \ \ \& \ \ \vec{x}_0$$

The adversary can query the oracle with the all-zero vector of length $2\lambda$

$$\vec{q} = 0^{2\lambda} = 0^{\lambda}||0^{\lambda}$$

The attack exploits a simple property of the **AND** operation: $0 \ \ \& \ \ a = 0$ for any $a$.

**Distinguishing test**:

- if $y$ is the all-zero vector, then it came from $L^{P}_{PRP-real}$
- if $y$ is not the all-zero vector, then it came from $L^{P}_{PRP-rand}$

## Feistel Cipher Example with AND Operation

### Parameters

- **λ = 4** (block size)
- **Message**: 00000000 (8 bits total)
- **Operation**: AND (&) instead of XOR ($\oplus$)
- **Rounds**: 3 rounds for this example

### Setup

#### Message Division

With λ = 4, we split the 8-bit message into two halves:

- **$L_0$ = 0000** (left half, 4 bits)
- **$R_0$ = 0000** (right half, 4 bits)

#### Round Function F

Let's define a simple round function F(R, K) where:

- **F(R, K) = R & K**

### Round Keys

- $\mathbf{K_1}$ **= 1010** (round 1 key)
- $\mathbf{K_2}$ **= 1100** (round 2 key)
- $\mathbf{K_3}$ **= 1111** (round 3 key)

## Encryption Process

### Round 1

**Input**: $L_0 = 0000$, $R_0 = 0000$

1. **Compute F($R_0$, $K_1$)**:

   ```
   F(0000, 1010) = 0000 & 1010 = 0000
   ```

2. **Update values**:

   ```
   L₁ = R₀ = 0000
   R₁ = L₀ & F(R₀, K₁) = 0000 & 0000 = 0000
   ```

**After Round 1**: $L_1 = 0000$, $R_1 = 0000$

### Round 2

**Input**: $L_1 = 0000$, $R_1 = 0000$

1. **Compute F($R_1$, $K_2$)**:

   ```
   F(0000, 1100) = 0000 & 1100 = 0000
   ```

2. **Update values**:

   ```
   L₂ = R₁ = 0000
   R₂ = L₁ & F(R₁, K₂) = 0000 & 0000 = 0000
   ```

**After Round 2**: $L_2 = 0000$, $R_2 = 0000$

### Round 3

**Input**: $L_2 = 0000$, $R_2 = 0000$

1. **Compute F($R_2$, $K_3$)**:

   ```
   F(0000, 1111) = 0000 & 1111 = 0000
   ```

2. **Update values**:

   ```
   L₃ = R₂ = 0000
   R₃ = L₂ & F(R₂, K₃) = 0000 & 0000 = 0000
   ```

**After Round 3**: $L_3 = 0000$, $R_3 = 0000$

**Final ciphertext =** $\mathbf{L_3 \| R_3}$ **= 0000 || 0000 = 00000000**