Learning Objectives

By the end of this worksheet, you will:

- Analyse the average running time of an algorithm.

1. **Average-case analysis.** Consider the following algorithm that we studied a few weeks ago. The input is an array \( A \) of length \( n \), containing a list of \( n \) integers.

   ```python
   def hasEven(A):
       """A is a list of integers."""
       n = len(A)
       even = False
       for i in range(n):
           if A[i] % 2 == 0:
               print('Even number found')
               return i
       print('No even number encountered')
       return -1
   ```

   In class we proved that the worst-case complexity of this algorithm is \( \Theta(n) \). In this problem we will examine the average case complexity of this algorithm. 

   For simplicity, we will assume that the input is a binary array \( A \) of length \( n \). That is, \( A \) is an array containing a list of \( n \) integers, where each integer is either 0 or 1.

   (a) For each \( n \in \mathbb{Z}^+ \), let \( T_n \) be the set of all binary arrays of length \( n \). Write an expression (in terms of \( n \)) for \( |T_n| \), the size of \( T_n \).

---

1 Depending on your lecture section, you may have seen this example already. Even so, please treat this question as a good opportunity for review!
(b) For each $n \in \mathbb{Z}^+$ and each $i \in \{0, 1, \ldots, n - 1\}$, let $S_n(i)$ denote the set of all binary arrays $A$ such that the first 0 occurs in position $i$. More precisely, let $S_n(i)$ denote the binary arrays that satisfy the following two properties:

(i) $A[i] = 0$.

(ii) for all $j \in \mathbb{N}$, if $j < i$ then $A[j] = 1$.

Also let $S_n(n)$ be the set of binary arrays that contain no 0’s. For each $i$, $0 \leq i \leq n$, write an expression for $|S_n(i)|$.

(c) Argue that for each $n \in \mathbb{Z}^+$, each binary array of length $n$ is in exactly one set $S_i$ (for some $i \in \{0, \ldots, n\}$).

Use this to show that $\sum_{i=0}^{n} |S_n(i)| = |T_n|$.
(d) Let the runtime of the algorithm on a binary list \( A \) be the number of executions of the loop. Give an exact expression for the average runtime of the above algorithm using the quantities that you calculated. You should get a summation; do not simplify the summation in this part.

(e) Show that the runtime that you calculated is in \( O(1) \). You may use without proof that for all \( x \in \mathbb{R} \) such that \(|x| < 1\),

\[
\sum_{i=1}^{\infty} ix^i = \frac{x}{(1-x)^2}.
\]