# COMP 2804 — Assignment 2

**Due:** Sunday October 14, before 11:55pm.

**Assignment Policy:**

- Your assignment must be submitted as one single PDF file through cuLearn.
- **Late assignments will not be accepted. I will not reply to emails of the type "my internet connection broke down at 11:53pm" or "my scanner stopped working at 11:54pm".**
- You are encouraged to collaborate on assignments, but at the level of discussion only. When writing your solutions, you must do so in your own words.
- Past experience has shown conclusively that those who do not put adequate effort into the assignments do not learn the material and have a probability near 1 of doing poorly on the exams.
- When writing your solutions, you must follow the guidelines below.
  - You must justify your answers.
  - The answers should be concise, clear and neat.
  - When presenting proofs, every step should be justified.

**Question 1:**

- Write your name and student number.

**Question 2:** The function $f : \mathbb{N} \rightarrow \mathbb{N}$ is defined by

$$\begin{aligned} f(0) &= 1, \\ f(n) &= 7 \cdot f(n-1) + (2n-1) \cdot 7^{n-1} \quad \text{if } n \geq 1. \end{aligned}$$

- Prove that for every integer $n \geq 0$,

$$f(n) = \left(n^2 + 7\right) \cdot 7^{n-1}.$$

**Question 3:** The function $f : \mathbb{N} \rightarrow \mathbb{N}$ is defined by

$$\begin{aligned} f(0) &= 0, \\ f(n) &= f(n-1) + 3 \cdot (f(n-1))^{2/3} + 3 \cdot (f(n-1))^{1/3} + 1 \quad \text{if } n \geq 1. \end{aligned}$$

- Solve this recurrence, i.e., express $f(n)$ in terms of $n$ only.

**Question 4:** Let $n \geq 1$ be an integer and consider the set $S = \{1, 2, \ldots, n\}$.

- Assume we arrange the elements of $S$ in sorted order on a horizontal line. Let $B_n$ be the number of subsets of $S$ that do not contain any two elements that are neighbors on this line. For example, if $n = 4$, then both subsets $\{1, 3\}$ and $\{1, 4\}$ are counted in $B_4$, but neither of the subsets $\{2, 3\}$ and $\{2, 3, 4\}$ is counted.

  For each integer $n \geq 1$, express $B_n$ in terms of numbers that we have seen in class.

- Assume we arrange the elements of $S$ in sorted order along a circle. Let $C_n$ be the number of subsets of $S$ that do not contain any two elements that are neighbors on this circle. For example, if $n = 4$, then the subset $\{1, 3\}$ is counted in $C_4$, but neither of the subsets $\{2, 3\}$ and $\{1, 4\}$ is counted.

  For each integer $n \geq 4$, express $C_n$ in terms of numbers that we have seen in class.

**Question 5:** In class, we have seen algorithm $\text{EUCLID}(a, b)$, which takes as input two integers $a$ and $b$ with $a \geq b \geq 1$, and returns their greatest common divisor.

- Assume we run this algorithm with two input integers $a$ and $b$ that satisfy $b > a \geq 1$. What is the output of this algorithm? As always, justify your answer.

**Question 6:** The Fibonacci numbers are defined by

$$\begin{aligned} f_0 &= 0, \\ f_1 &= 1, \\ f_n &= f_{n-1} + f_{n-2}, \text{ if } n \geq 2. \end{aligned}$$

The goal of this exercise is to prove that there exists a Fibonacci number whose 2018 rightmost digits (when written in decimal notation) are all zero.

In the rest of this exercise, $N$ denotes the number $10^{4036}$. For any integer $n \geq 0$, define

$$g_n = f_n \bmod 10^{2018}.$$

- Consider the ordered pairs $(g_n, g_{n+1})$, for $n = 0, 1, 2, \ldots, N$. Use the Pigeonhole Principle to prove that these ordered pairs cannot all be distinct. That is, prove that there exist integers $m \geq 0$, $p \geq 1$, such that $m + p \leq N$ and

$$(g_m, g_{m+1}) = (g_{m+p}, g_{m+p+1}).$$

- Prove that $(g_{m-1}, g_m) = (g_{m+p-1}, g_{m+p})$.
- Prove that $(g_0, g_1) = (g_p, g_{p+1})$.
- Consider the decimal representation of $f_p$. Prove that the 2018 rightmost digits of $f_p$ are all zero.

**Question 7:** In this exercise, we consider strings of characters, where each character is an element of $\{a, b, c\}$. Such a string is called *aa-free*, if it does not contain two consecutive $a$'s. For any integer $n \geq 1$, let $F_n$ be the number of $aa$-free strings of length $n$.

- Determine $F_1$, $F_2$, and $F_3$.
- Let $n \geq 3$ be an integer. Express $F_n$ in terms of $F_{n-1}$ and $F_{n-2}$.
- Prove that for every integer $n \geq 1$,

  $$F_n = \left(\frac{1}{2} + \frac{1}{\sqrt{3}}\right)\left(1+\sqrt{3}\right)^n + \left(\frac{1}{2} - \frac{1}{\sqrt{3}}\right)\left(1-\sqrt{3}\right)^n .$$

  *Hint:* What are the solutions of the equation $x^2 = 2x + 2$? Using these solutions will simplify the proof.

**Question 8:** Let $m \geq 1$ and $n \geq 1$ be integers and consider an $m \times n$ matrix $A$. The rows of this matrix are numbered $1, 2, \ldots, m$, and its columns are numbered $1, 2, \ldots, n$. Each entry of $A$ stores one number and, for each row, all numbers in this row are pairwise distinct. For each $i = 1, 2, \ldots, m$, define

$$g(i) = \text{ the position (i.e., column number) of the smallest number in row } i.$$

We say that the matrix $A$ is *awesome*, if

$$g(1) \leq g(2) \leq g(3) \leq \ldots \leq g(m).$$

In the matrix below, the smallest number in each row is in boldface. For this example, we have $m = 4$, $n = 10$, $g(1) = 3$, $g(2) = 3$, $g(3) = 5$, and $g(4) = 8$. Thus, this matrix is awesome.

$$A = \begin{pmatrix} 13 & 12 & \mathbf{5} & 8 & 6 & 9 & 15 & 20 & 19 & 7 \\ 3 & 4 & \mathbf{1} & 17 & 6 & 13 & 7 & 10 & 2 & 5 \\ 19 & 5 & 12 & 7 & \mathbf{2} & 4 & 11 & 13 & 6 & 3 \\ 7 & 4 & 17 & 10 & 5 & 14 & 12 & \mathbf{3} & 20 & 6 \end{pmatrix}.$$

From now on, we assume that the $m \times n$ matrix $A$ is awesome.

- Let $i$ be an integer with $1 \leq i \leq m$. Describe, in plain English and a few sentences, an algorithm that computes $g(i)$ in $O(n)$ time.
- Describe, in plain English and a few sentences, an algorithm that computes all values $g(1), g(2), \ldots, g(m)$ in $O(mn)$ total time.

In the rest of this exercise, you will show that all values $g(1), g(2), \ldots, g(m)$ can be computed in $O(m + n \log m)$ total time.

- Assume that $m$ is even and assume that you are given the values

$$g(2), g(4), g(6), g(8), \ldots, g(m).$$

Describe, in plain English and using one or more figures, an algorithm that computes the values

$$g(1), g(3), g(5), g(7), \ldots, g(m-1)$$

in $O(m+n)$ total time.

- Assume that $m = 2^k$, i.e., $m$ is a power of two. Describe a recursive algorithm FindMinima that has the following specification:

**Algorithm** FindMinima$(A, i)$**:**
**Input:** An $m \times n$ awesome matrix $A$ and an integer $i$ with $0 \leq i \leq k$.
**Output:** The values $g\left(j \cdot m/2^i\right)$ for $j = 1, 2, 3, \ldots, 2^i$.

For each $i$ with $0 \leq i \leq k$, let $T(i)$ denote the running time of algorithm FindMinima$(A, i)$. The running time of your algorithm must satisfy the recurrence

$$\begin{aligned} T(0) &= O(n), \\ T(i) &= T(i-1) + O\left(2^i + n\right), \text{ if } 1 \leq i \leq k. \end{aligned}$$

You may use plain English and figures to describe your algorithm, but it must be clear how you use recursion.

- Assume again that $m = 2^k$. Prove that all values $g(1), g(2), \ldots, g(m)$ can be computed in $O(m + n \log m)$ total time.

*Hint:* $1 + 2 + 2^2 + 2^3 + \cdots + 2^k \leq 2m$.