Final exam (English)

Problem 1. Let $X = \{0,1,2,3,4,5,6,9\}$ (notice that 7 and 8 were left out). Using **only** the numbers in X, how many strings of length n are there such that the sum of their digits is congruent to $1 \pmod 3$?

Solution. We classify the strings according to the congruence of the sum of their digits in $\mathbb{Z}_3$. Let a_n, b_n, & c_n be the number of strings of length n whose digits add up to a number congruent to $0 \pmod 3$, $1 \pmod 3$, & $2 \pmod 3$, respectively. We seek b_n.

As we are splitting the set of all strings of length n into these three classes, we have that $a_n + b_n + c_n = 8^n$.

Moreover, when extending a string of length n into another of length $n+1$, we have that $b_{n+1} = 2a_n + 4b_n + 2c_n$.

This means that $b_{n+1} = 2b_n + 2 \cdot 8^n$. Hence, if we decompose the solution to the recurrence as $b_n = b_n^h + b_n^p$ (sum of the homogeneous-problem solution and the particular solution), we get that $b_n^h = B \cdot 2^n$, & $b_n^p = \alpha \cdot 8^n$, where $\alpha = 1/3$.

Lastly, we can check that $b_1 = 2$, thus $2 = 2B + 8/3$, from which $B = -1/3$.

All in all, there are

$$\boxed{b_n = \frac{1}{3}\left(8^n - 2^n\right)}.$$

Problem 2. A conference on comics is being organized, and it will hold eleven talks. Four of them are about Marvel sagas (and we can call them m_1, m_2, m_3, & m_4); other four are related to DC Comics (we call them d_1, d_2, d_3, & d_4); the last three are about independent comics (i_1, i_2, & i_3).

Moreover, we know that some speakers will give two talks: i_1 & m_3 are from the same speaker; d_2 & i_2 are from the same person. Lastly, it turns out that i_3 is a shared talk between the speakers from m_2 & d_3.

To hold the talks, the organizers need to rent some conference rooms, and they are rented in time slots of 45 minutes. They want to be able to hold all the talks. Also to allow the Marvel fans to attend to all the Marvel talks, the DC fans to attend all the DC talks, and for the *indie* fans to be able to attend all the independent talks.

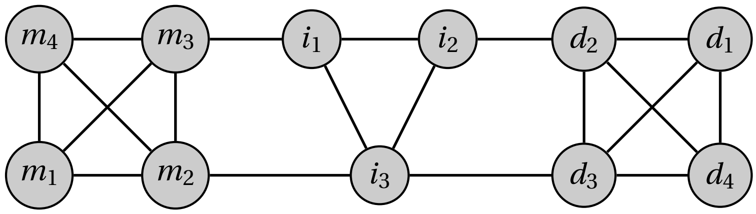
- Using graph theory techniques, find the minimum time required to hold all the talks.
- What is the largest number of conference rooms that are possibly needed at a single time slot?

Solution. In order to solve this problem we need to define a graph whose vertices are the talks, and whose edges represent "not being able to take place at the same time".

From the statement of the problem, we get that all the Marvel talks must be connected, thus forming a K_4; the same happens to the DC talks; the indie talks, instead, form a K_3.

As we also have to consider the constraints related to the speakers, we obtain the following graph:

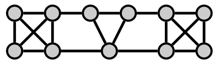
$G \equiv$

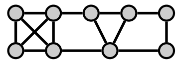


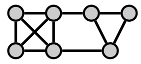
We seek proper colorings for this graph, and think of each color as a possible time slot.

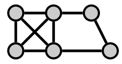
1. We need $\chi(G)$. A fast way to find is is to argue that, as there is a K_4, then $\chi(G) \geq 4$; on the other hand, with some luck we can color the graph using exactly 4 colors, hence $\chi(G) \leq 4$. All in all, $4 \leq \chi(G) \leq 4$, so $\chi(G) = 4$. That is, *at least* four time slots are needed, which can be thought of as **the convention needs to last at least 3 hours**.

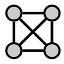
A more technical solution uses the chromatic polynomial as follows:

$P_G(q) = P\Big($  $\Big) =$

$= \Big[P\Big($  $\Big) \cdot P_{K_4}(q)\Big] \Big/ P_{K_2}(q) =$

$= \Big[P\Big($  $\Big) \cdot P_{C_4}(q) \cdot P_{K_4}(q)\Big] \Big/ \left[P_{K_2}(q)\right]^2 =$

$= \Big[P\Big($  $\Big) \cdot P_{K_3}(q) \cdot P_{C_4}(q) \cdot P_{K_4}(q)\Big] \Big/ \left[P_{K_2}(q)\right]^3 =$

$= \Big[P\Big($  $\Big) \cdot P_{K_3}(q) \cdot \left[P_{C_4}(q)\right]^2 \cdot P_{K_4}(q)\Big] \Big/ \left[P_{K_2}(q)\right]^4 =$

$$= \frac{P_{K_3}(q) \cdot \left[P_{C_4}(q)\right]^2 \cdot \left[P_{K_4}(q)\right]^2}{\left[P_{K_2}(q)\right]^4},$$

from which we can deduce that $q = 0, 1, 2, 3$ are the only integer roots. Hence $\chi(G) = 4$.

2. The largest number of rooms possibly needed will be the largest number of times a single time-slot could be used. That is, we need to check how many times a single color can be used *at most.*

   A single color, red for instance, can show up once in each K_4, K_3, & K_4 in the graph. So we can foretell that red can be used, at most, three times. But since it could be possible that there is no proper coloring using red three times, we need to check that possibility.

   Looking at the structure of G, we could paint in red m_3, i_3 & d_2. That is, $c(m_3) = c(i_3) = c(d_2) =$ red. Then we could color $c(i_1) =$ blue & $c(i_2) =$ purple. This means that the remaining vertices of the K_4 formed by the m_i's cannot used either red nor blue. Moreover, the remaining vertices d_i cannot be colored either by red nor purple. Therefore, if we paint these vertices in red, using a total of 5 colors we can achieve a proper coloing. We conclude that the largest number of times a single color can be used is three, so we are going to need **at most three rooms at the same time**.

We can lower the number of colors used and find that there are proper colorings using red three times and using a total of four colors. For instance: $c(m_1) = c(i_3) = c(d_4) =$ red, $c(m_2) = c(i_1) = c(d_1) =$ blue, $c(m_3) = c(i_2) = c(d_3) =$ purple, y $c(m_4) = c(d_2) =$ orange. In fact, we are using three colors three times.

Problem 3. We have built a quite basic calculator that only works with numbers that can be represented using a single byte (which is a length 8 bit-string), and we have coded in the basic operations of the integer numbers. After some testing we find out that, for this calculator, $255+1=0$, $100+200=44$, and $100-200=156$.

1. Justify which value will the calculator show when computing 35^{87047}.

2. Justify if the calculator can find a solution for the equation $20x = 32$. If that's the case, find the solutions that the calculator can represent.

Solution. With a single byte we can represent 256 different values. By looking at the tests, we can tell the calculator represents values in $\mathbb{Z}_{256}$.

1. Since $256 = 2^8$ is not prime, we can only use Euler's theorem. As $\gcd(256,35) = 1$, we can use it. Since $\phi(256) = \phi(2^8) = 2^7 = 128$, we can claim that $35^{128} \equiv 1 \pmod{256}$.

   Dividing, we get $87\,047 = 128{\cdot}680+7$. From which, $35^{87\,047} \equiv 35^7 \pmod{256}$). We are only left with some manual calculations: $35^2 \equiv 201$, $35^4 \equiv 209$, $35^6 \equiv 25 \pmod{256}$. Hence,

   $$\boxed{35^{87\,047} \equiv 107 \pmod{256}}.$$

2. We want to solve $20x \equiv 32 \pmod{256}$.

   Let $d = \gcd(20,256) = 4$. As d divides 32, there are solutions. In particular, there are $d = 4$ solutions in $\mathbb{Z}_{256}$.

First we solve $20\tilde{x} \equiv 4 \pmod{256}$, equations that we reinterpret as the Bezout's Identity $20\tilde{x} + 256\tilde{y} = 4$. Using Euclid's lemma multiple times we get the following list of equations:

$$256 = 20 \cdot 12 + 16,$$
$$20 = 16 \cdot 1 + 4,$$

from where

$$\begin{aligned} 4 &= 20 + (-1)16 = 20 + (-1)\left[256 + (-12)20\right] = \\ &= 20\left[1 + (-1)(-12)\right] + 256\left[(-1)\right] = \\ &= 20(13) + 256(-1), \end{aligned}$$

and thus, $\tilde{x} \equiv 13 \pmod{256}$.

Since $32 = 8d$, we have $x \equiv 8\tilde{x} \equiv 8 \cdot 13 \equiv 104 \pmod{256}$ is a solution to the main problem. We still need to find the other ones.

Let $x_0 = x = 104$. The remaining solutions can be found as $x_k = x_0 + \frac{256}{d}k = 104 + 64k$, with $k \in \mathbb{Z}$. We are only interested in the solutions that fall into $0 \leq x_k \leq 255$. In this case, x_{-1}, x_0, x_1, x_2.

All in all, the four solutions are:

$$\boxed{x \equiv 40,\ 104,\ 168,\ 232 \pmod{256}}.$$