Problem 1. Missile or not missile...

(t, w)-threshold scheme will be used. Given each the general 10 shares, each colonel 5 shares, each desk clerk 2 share, thus we have $t = 10$ and $w = 30$.

Problem 2. Asmuth-Bloom Threshold Secret Sharing Scheme

In order to build secret sharing using CRT, we let $2 \leqslant k \leqslant n$, a sequence of relatively prime integers $m_0 < m_1 < \cdots < m_n$ such that

$$m_0 \cdot m_{n-k+2} \cdots m_n < m_1 \cdots m_k$$

Given secret in the set Z/m_0Z as S, we pick a random integer α so that $S + \alpha \cdot m_0 < m_1 \cdots m_k$. After computing $s_i \equiv S + \alpha m_0 \mod m_i$ for $1 \leqslant i \leqslant n$, we can get the shares $I_i = \langle s_i, m_i \rangle$. Now we can take any of k different shares from n shares, $I_{i_1}, I_{i_2}, \ldots, I_{i_k}$, so that

$$\begin{cases} x \equiv s_{i_1} \mod m_{i_1} \\ \vdots \\ x \equiv s_{i_k} \mod m_{i_k} \end{cases}$$

According to the CRT, we can decide a unique $x < m_{i_1} \cdot m_{i_2} \cdots m_{i_k}$.
By the construction of the shares, we can get

$$S \equiv x \bmod m_0$$

Problem 3. Shamir's Threshold Secret Sharing Scheme

To recover the shared secret, given a set of data points

$$(x_1, y_1), ..., (x_{t-1}, y_{t-1})$$

The interpolation polynomial in Lagrange form is a linear combination

$$L(x) = \sum_{i=1}^{t-1} y_i L_i(x)$$

$$L_i(x) = \frac{(x - x_0) \cdots (x - x_{i-1})(x - x_{i+1}) \cdots (x - x_{t-1})}{(x_i - x_0) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x_i - x_{t-1})}$$

where $0 \leq i \leq t - 1$.
To solve lecture's example, we have $p = 1234567890133$, $m = 190503180520$, $r_1 = 482943028839$, $r_2 = 1206749628665$.
Since we want to construct a (3,8)-threshold scheme, we need 3 data pairs to recover the polynomial. We choose

$$< x_0, y_0 >=< 2, 1045116192326 >, < x_1, y_1 >=< 3, 154400023692 >, < x_2, y_2 >=< 7, 973441680328 >$$

$$
\begin{aligned}
L_0(x) &= \frac{(x-3)(x-7)}{(2-3)(2-7)} = \frac{1}{5}(x-3)(x-7) \\
L_1(x) &= \frac{(x-2)(x-7)}{(3-2)(3-7)} = -\frac{1}{4}(x-2)(x-7) \\
L_2(x) &= \frac{(x-2)(x-3)}{(7-2)(7-3)} = \frac{1}{20}(x-2)(x-3)
\end{aligned}
$$

$$
\begin{aligned}
L(x) &= \sum_{i=0}^{2} y_i L_i(x) \\
&= \frac{1045116192326}{5}(x-3)(x-7) - \frac{154400023692}{4}(x-2)(x-7) + \frac{973441680328}{20}(x-2)(x-3) \\
&= \frac{1095476582793}{5}x^2 - 1986192751427x + \frac{20705602144728}{5}
\end{aligned}
$$

This yields

$$(m, r_1, r_2) = 190503180520, 482943028839, 1206749628665$$

Problem 4. Simple questions

1. Given $z_1 = 2x + 3y + 13$, $z_2 = 5x + 3y + 1$

$$
\begin{aligned}
z_1 &= z_2 \\
2x + 3y + 13 &= 5x + 3y + 1 \\
x &= 4 \\
z &= 3y + 21
\end{aligned}
$$

   The secret value x is 4.

2. The proof can be done using mathematical induction and cramer's rule.

   (a) When $n = 2$,

$$\det V_2 = x_2 - x_1 = \prod_{1 \leqslant j \leqslant k \leqslant 2} (x_k - x_j)$$

   (b) When $n = m \geqslant 2$, suppose

$$\det V_m = \prod_{1 \leqslant j \leqslant k \leqslant m} (x_k - x_j)$$

   (c) When $n = m + 1$, we calculate

$$\det V_{m+1} = \begin{vmatrix} 1 & x_1 & \cdots & x_1^{m-1} & x_1^m \\ 1 & x_2 & \cdots & x_2^{m-1} & x_2^m \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_m & \cdots & x_m^{m-1} & x_m^m \\ 1 & x_{m+1} & \cdots & x_{m+1}^{m-1} & x_{m+1}^m \end{vmatrix}$$

For every ith column, multiply the entry by $-x_{m+1}$ and add to $(i+1)$th column $(1 \le i \le m)$, we have

$$\begin{aligned}
\det V_{m+1} &= \begin{vmatrix} 1 & x_1 - x_{m+1} & \cdots & x_1^{m-2}(x_1 - x_{m+1}) & x_1^{m-1}(x_1 - x_{m+1}) \\ 1 & x_2 - x_{m+1} & \cdots & x_2^{m-2}(x_2 - x_{m+1}) & x_2^{m-1}(x_2 - x_{m+1}) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_m - x_{m+1} & \cdots & x_m^{m-2}(x_m - x_{m+1}) & x_m^{m-1}(x_2 - x_{m+1}) \\ 1 & 0 & \cdots & 0 & 0 \end{vmatrix} \\
&= \prod_{i=1}^{m}(x_{m+1} - x_i) \begin{vmatrix} 1 & x_1 & \cdots & x_1^{m-2} & x_1^{m-1} \\ 1 & x_2 & \cdots & x_2^{m-2} & x_2^{m-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x_{m-1} & \cdots & x_{m-1}^{m-2} & x_{m-1}^{m-1} \\ 1 & x_m^2 & \cdots & x_m^{m-2} & x_m^{m-1} \end{vmatrix} \\
&= \prod_{i=1}^{m}(x_{m+1} - x_i) \det V_m \\
&= \prod_{i=1}^{m}(x_{m+1} - x_i) \prod_{1 \leqslant j \leqslant k \leqslant m} (x_k - x_j) \\
&= \prod_{1 \leqslant j \leqslant k \leqslant m+1} (x_k - x_j)
\end{aligned}$$

3. Just finished with all 5 out of 5 :)

Problem 5. Reed Solomon codes

1. Reed-Solomon codes are a group of error-correcting codes, which every code is characterized by three parameters: an alphabet size q, a block length n and a message length k with $k < n \leqslant q$. q is a prime power as the alphabet symbol is interpreted as a finite field of order q. The block length is usually some constant multiple of the message length and is equal to or one less than the alphabet size, that is, $n = q$ or $n = q - 1$.

   Every codeword of the Reed Solomon code is a sequence of function values of a polynomial p of degree less than k. The message is interpreted as the description of a polynomial p of degree less than k over the finite field F with q elements. In turn, the polynomial p is evaluated at n distinct points $a_1, a_2, \dots, a_n$ of the field F, and the sequence of values is the corresponding codeword.

   Formally, the set $\mathcal{C}$ of codewords of the Reed Solomon code is defined as follows:

$$\mathcal{C} = \{(p(a_1), p(a_1), \dots, p(a_3) | \text{p is a polynomial over F of degree} < k\}$$

2. The minimal distance D is found since any two distinct polynomials of degree less than k agree in at most $k - 1$ points, which gives $D = n - k + 1$, result in the positions of any two of the Reed Solomon code disagree with.

It is possible to identify a parent of a descendant of $\mathcal{C} \subset (F_q)^n$ with $D > n(1 - \frac{1}{w^2})$ where n is the length of code. When coalition size $w = 2$,

$$\begin{aligned} n - k + 1 &> \frac{3}{4}n \\ n &> 4k - 4 \end{aligned}$$

Therefore, the condition on the length of code should be greater than $4k - 4$.