Lecture slides of the course

Information hiding & secret sharing

Secret sharing

Phạm Trọng Nghĩa

ptnghia@fit.hcmus.edu.vn

Course content

- Data/information hiding
  - Steganography ✓
  - Watermarking ✓
- Secret sharing ← This lecture

Secret sharing information

- Suppose we want to share a secret information ***S*** with ***n*** people
  - We want to have at least ***k*** people ($k \leq n$) to reproduce ***S***
  - If there are less than ***k*** people, it will not be possible to reproduce ***S*** (don't know anything about ***S***)
- **The problem of secret sharing information:** divide the serect information S to n part $S_1, S_2, \ldots, S_n$ such as:
  - With at least any ***k*** part ($k \leq n$), we can reproduce S
  - With less than ***k*** part, we don’t know anything about S

Shamir method of sharing secret information

In Shamir's method, secret data is a number

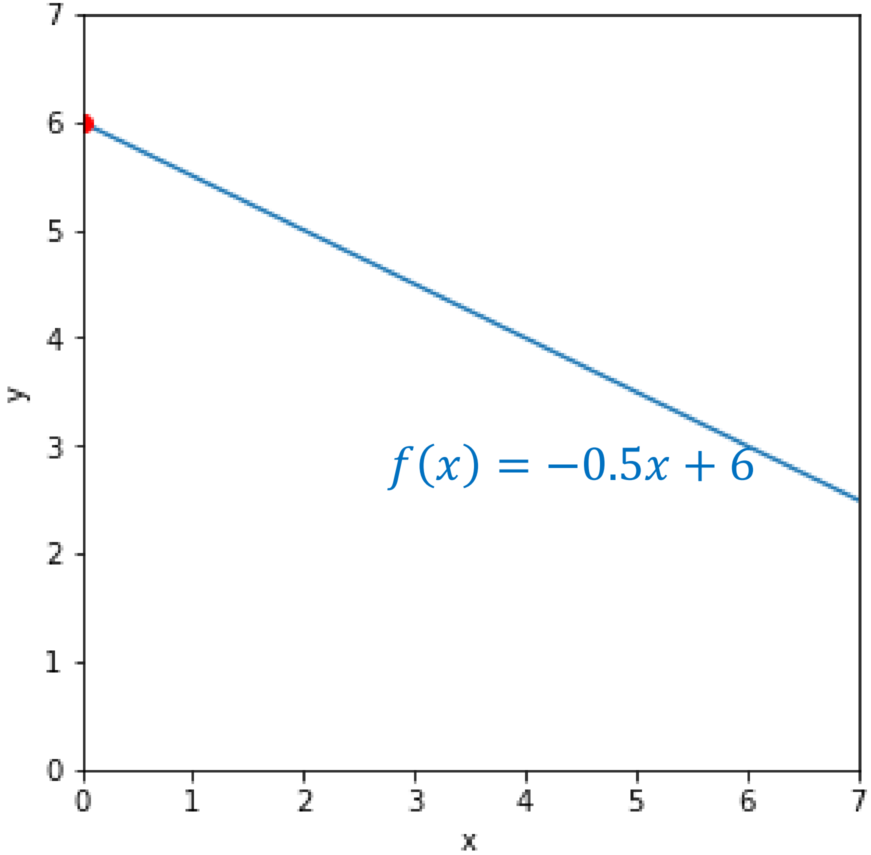
Shamir method of sharing secret information

Example 1: $S = 6, n = 3, k = 2$

How does the Shamir method work?

- Step 1: create f is a first degree polynomial such that $f(0) = S$

  eg, $f(x) = -0.5x + 6$

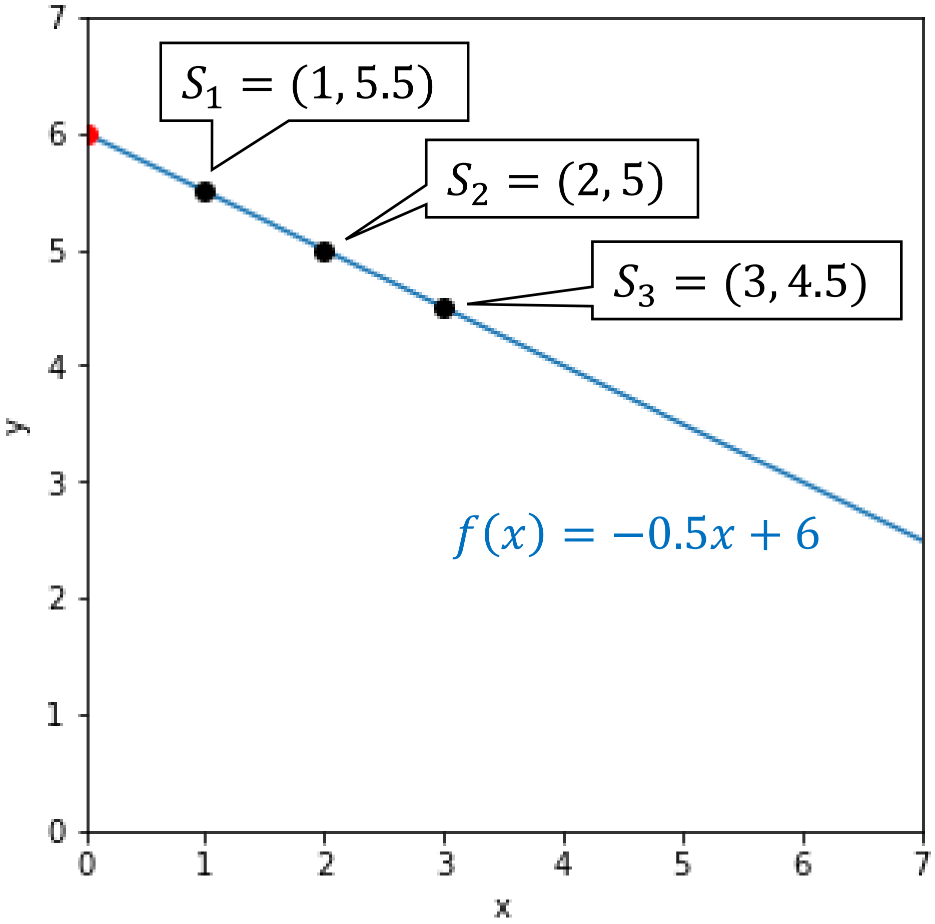


Shamir method of sharing secret information

Example 1: $S = 6, n = 3, k = 2$

How does the Shamir method work?

- Step 2: $S_1 = (1, f(1)), S_2 = (2, f(2)), S_3 = (3, f(3))$

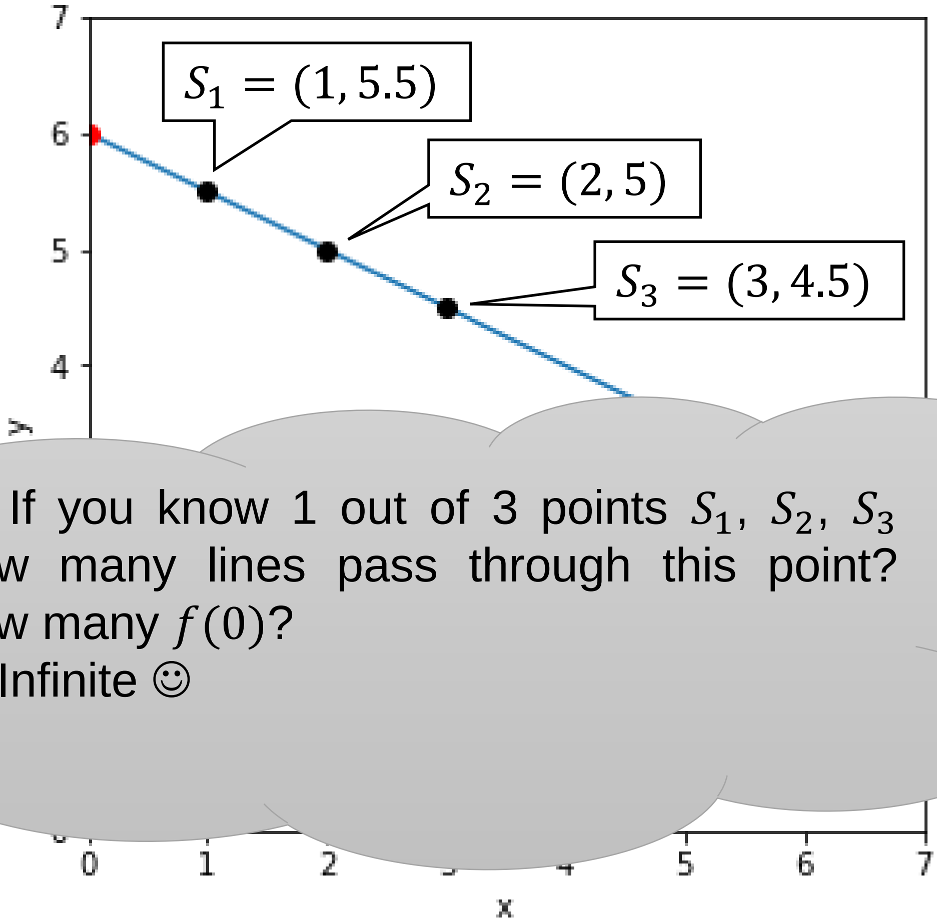


Shamir method of sharing secret information

Example 1: $S = 6, n = 3, k = 2$

How does the Shamir method work?

- Step 2: $S_1 = (1, f(1)), S_2 = (2, f(2)), S_3 = (3, f(3))$



$S_1 = (1, 5.5)$

$S_2 = (2, 5)$

$S_3 = (3, 4.5)$

Q: If you know 1 out of 3 points S_1, S_2, S_3 How many lines pass through this point? How many $f(0)$?

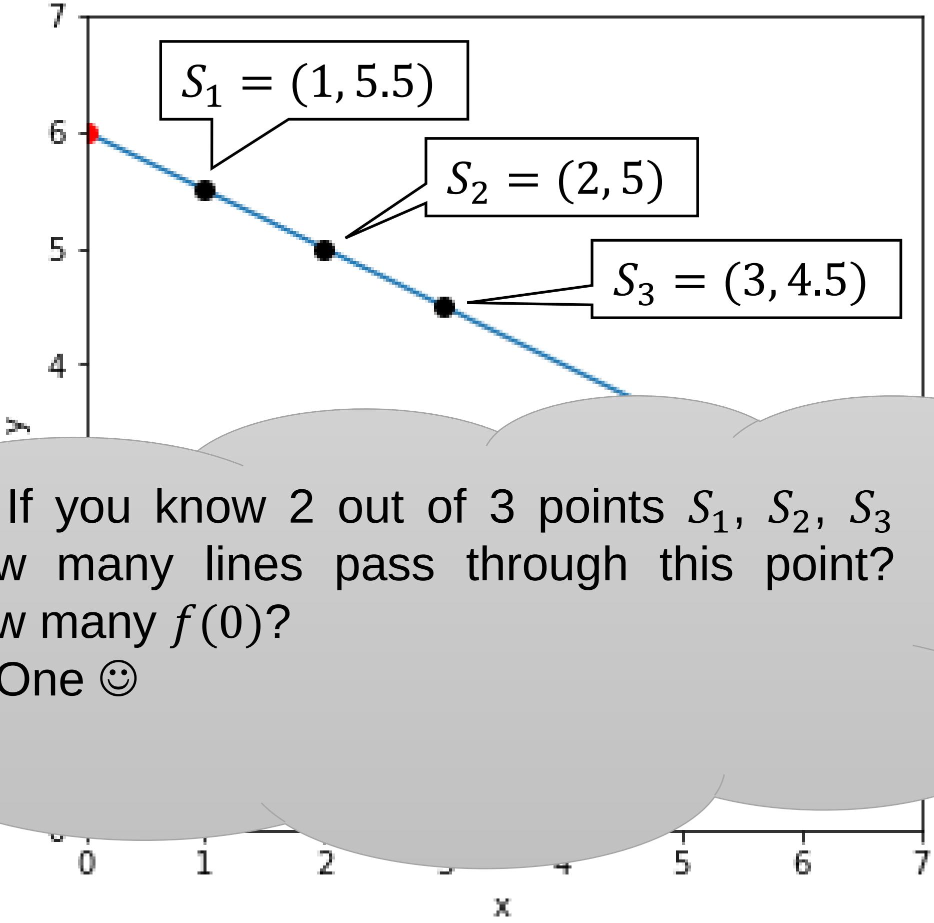
A: Infinite ☺

Shamir method of sharing secret information

Example 1: $S = 6, n = 3, k = 2$

How does the Shamir method work?

- Step 2: $S_1 = (1, f(1)), S_2 = (2, f(2)), S_3 = (3, f(3))$



$S_1 = (1, 5.5)$

$S_2 = (2, 5)$

$S_3 = (3, 4.5)$

Q: If you know 2 out of 3 points S_1, S_2, S_3 How many lines pass through this point? How many $f(0)$?

A: One ☺

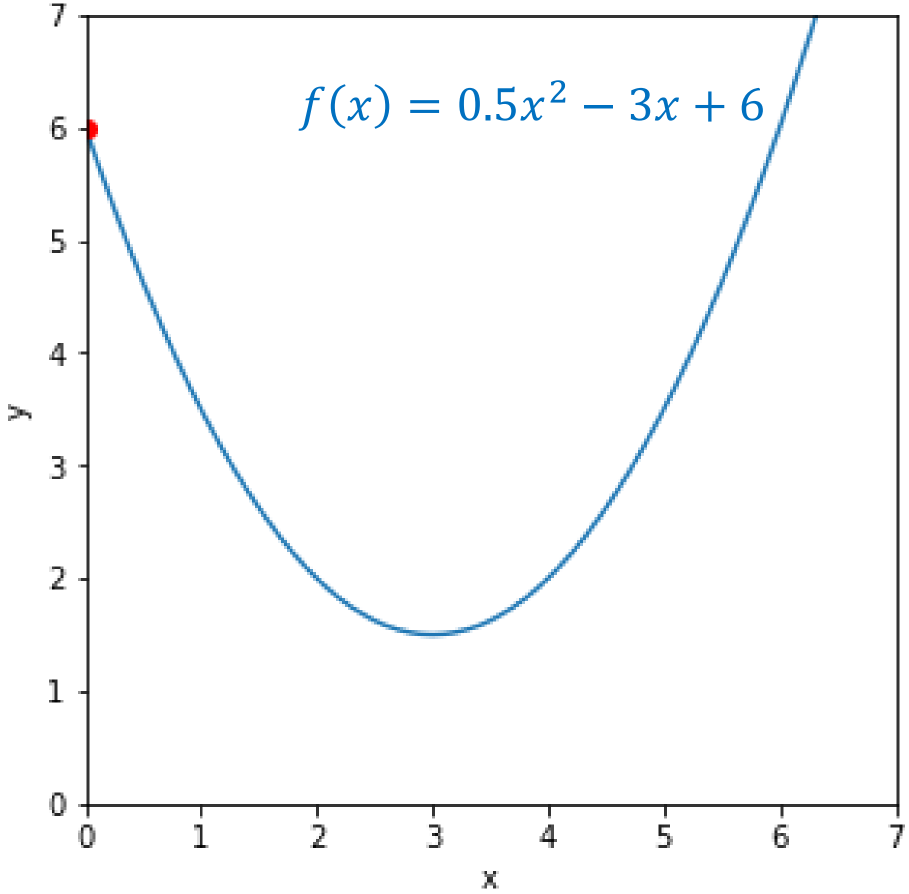
Shamir method of sharing secret information

Example 2: $S = 6, n = 5, k = 3$

How does the Shamir method work?

- Step 1: create function f is a quadratic polynomial such that $f(0) = S$

eg., $f(x) = 0.5x^2 - 3x + 6$

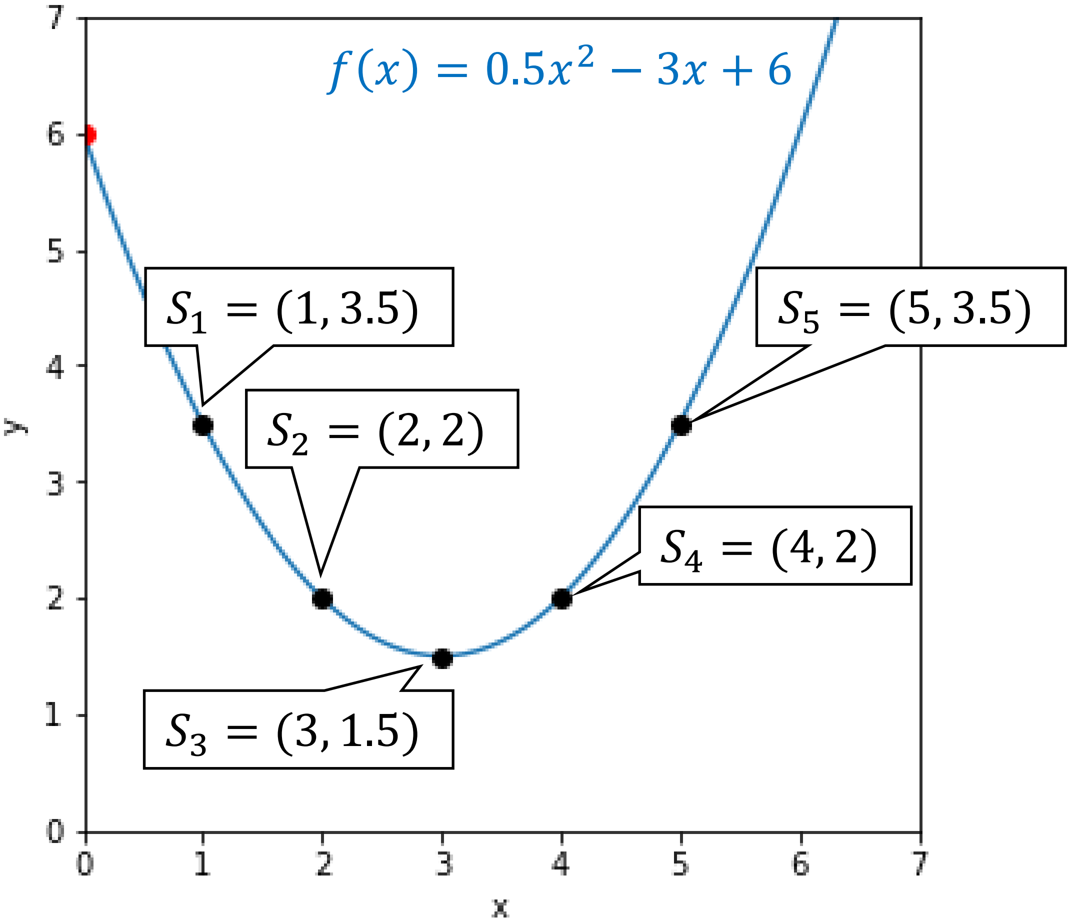


Shamir method of sharing secret information

Example 2: $S = 6, n = 5, k = 3$

How does the Shamir method work??

- Step 2: $S_1 = (1, f(1)), S_2 = (2, f(2)), S_3 = (3, f(3)), \ldots$



Shamir method of sharing secret information

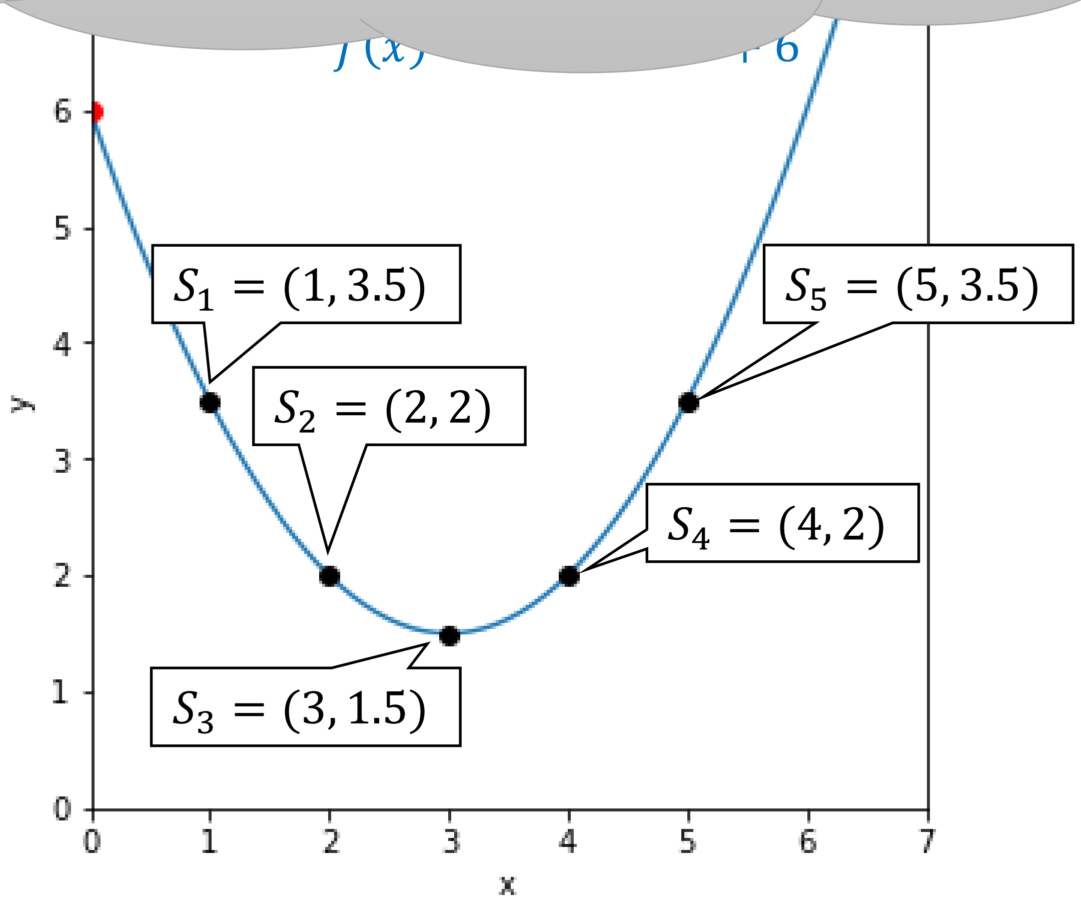
Example 2: S

How does th

□ S

Q: If 2 out of 5 points S_1, S_2, S_3, S_4, S_5 are known, how many quadratic curves pass through these two points? How many $f(0)$?

A: Infinite ☺



Shamir method of sharing secret information

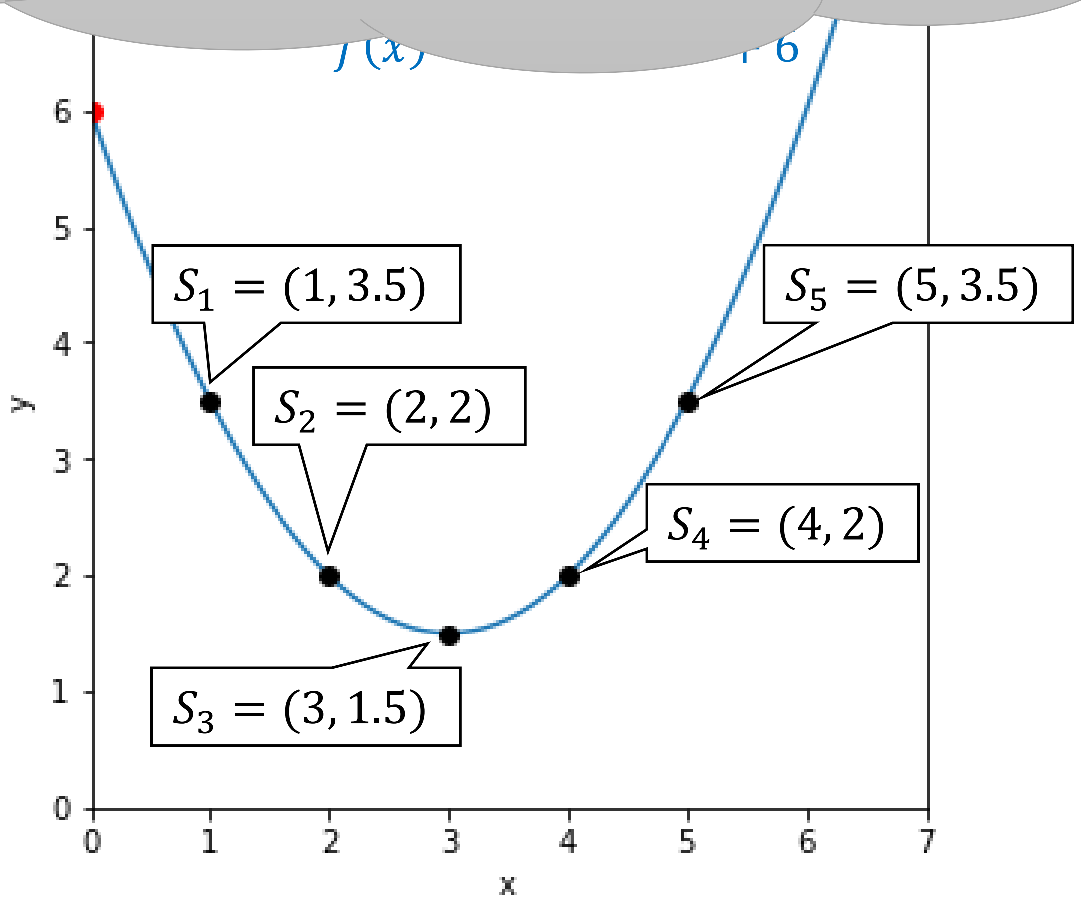
Example 2: S

How doe

- S

Q: If 3 out of 5 points S_1, S_2, S_3 , S_4 , S_5 are known, how many quadratic curves pass through these two points? How many $f(0)$?

A: One ☺



Shamir method of sharing secret information

Sharing secret information

- Input
  - Secret information S (a number)
  - Number of parts to be divided n
  - Threshold k
- Algorithm:
  - Create function f is a $k-1$ degree polynomial such that $f(0)=S$
  - $S_1=(1,f(1)), S_2=(2,f(2)), S_3=(3,f(3)), \ldots$

Shamir method of sharing secret information

Recreate secret information

- Input
  - Threshold k
  - n’ part of secret information ($n' \geq k$)
- Algorithm
  - Create function f (polynomial of degree $k-1$) from k in n' parts
  - $S = f(0)$

Recreate function f (polynomial of degree $k-1$) from k parts of secret information?

Find a polynomial function of degree $k-1$

$[f(x) = a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + \cdots + a_0)]$

From k points $(x_1, y_1), \ldots, (x_k, y_k)$ with $y_i = f(x_i)$

→ k equations, k variables:

- $a_{k-1}x_1^{k-1} + a_{k-2}x_1^{k-2} + \cdots + a_0 = y_1$
- $a_{k-1}x_2^{k-1} + a_{k-2}x_2^{k-2} + \cdots + a_0 = y_2$
- ...
- $a_{k-1}x_k^{k-1} + a_{k-2}x_k^{k-2} + \cdots + a_0 = y_k$

How to solve?

- An efficient solution is to use Lagrange interpolation

Lagrange interpolation

Example, find function f (polynomial function of degree 3) with:

- $f(5) = 3$
- $f(7) = 2$
- $f(12) = 6$
- $f(30) = 15$

We only need to find a function : (i) degree 3, and (ii) go through 4 points above

$$f(x) = 3\delta_5(x) + 2\delta_7(x) + 6\delta_{12}(x) + 15\delta_{30}(x)$$

with $\delta_i(x)$ is function of degree 3

$\delta_i(x) = ?$

$$\delta_i(x) = \begin{cases} 1 \text{ if } x = i \\ 0 \text{ if } x \in \{5,7,12,30\} - \{i\} \\ \text{"any" in other case} \end{cases}$$

Lagrange interpolation

Function with degree ***n*** have form:

$$f(x) = \sum_{i=0}^{n} \delta_{x_i}(x) \times y_i$$

With:

$$\delta_{x_i}(x) = \frac{\prod_{k=0,k\neq i}^{n}(x - x_k)}{\prod_{k=0,k\neq i}^{n}(x_i - x_k)}$$

Lagrange interpolation

Example, find function f (polynomial function of degree 3) with:

- $f(5) = 3$
- $f(7) = 2$
- $f(12) = 6$
- $f(30) = 15$

$$\boldsymbol{f(x) = \sum_{i=0}^{n} \delta_{x_i}(x) \times y_i}$$

We only need to find a function : (i) degree 3, and (ii) go through 4 points above

$$f(x) = 3\delta_5(x) + 2\delta_7(x) + 6\delta_{12}(x) + 15\delta_{30}(x)$$

Lagrange interpolation

Example, find function f (polynomial function of degree 3) with:

- $f(5) = 3, f(7) = 2, f(12) = 6, f(30) = 15$

Polynomial have form:

$$f(x) = 3\delta_5(x) + 2\delta_7(x) + 6\delta_{12}(x) + 15\delta_{30}(x)$$

With:

$$\delta_{x_i}(x) = \frac{\prod_{k=0,k\neq i}^{n}(x - x_k)}{\prod_{k=0,k\neq i}^{n}(x_i - x_k)}$$

$$\delta_{\color{red}{5}}(x) = \frac{x-7}{\color{red}{5}-7}\frac{x-12}{\color{red}{5}-12}\frac{x-30}{\color{red}{5}-30}$$

Lagrange interpolation

Example, find function f (polynomial function of degree 3) with:

- $f(5) = 3$
- $f(7) = 2$
- $f(12) = 6$
- $f(30) = 15$

We only need to find a function : (i) degree 3, and (ii) go through 4 points above

$f(x) = 3\delta_5(x) + 2\delta_7(x) + 6\delta_{12}(x) + 15\delta_{30}(x)$

with $\delta_i(x)$ is function of degree 3

$\delta_i(x) = ?$

$$\delta_i(x) = \begin{cases} 1 \text{ if } x = i \\ 0 \text{ if } x \in \{5,7,12,30\} - \{i\} \\ \text{"any" in other case} \end{cases}$$

$$\delta_7(x) = \frac{x-5}{7-5}\frac{x-12}{7-12}\frac{x-30}{7-30}$$

Lagrange interpolation

Example, find function f (polynomial function of degree 3) with:

- $f(5) = 3$
- $f(7) = 2$
- $f(12) = 6$
- $f(30) = 15$

We only need to find a function : (i) degree 3, and (ii) go through 4 points above

$$f(x) = 3\delta_5(x) + 2\delta_7(x) + 6\delta_{12}(x) + 15\delta_{30}(x)$$

with $\delta_i(x)$ is function of degree 3

$\delta_i(x) = ?$

$$\delta_i(x) = \begin{cases} 1 \text{ if } x = i \\ 0 \text{ if } x \in \{5,7,12,30\} - \{i\} \\ \text{"any" in other case} \end{cases}$$

$$\delta_{12}(x) = \frac{x-5}{12-5}\frac{x-7}{12-7}\frac{x-30}{12-30}$$

Lagrange interpolation

Example, find function f (polynomial function of degree 3) with:

- $f(5) = 3$
- $f(7) = 2$
- $f(12) = 6$
- $f(30) = 15$

We only need to find a function : (i) degree 3, and (ii) go through 4 points above

$f(x) = 3\delta_5(x) + 2\delta_7(x) + 6\delta_{12}(x) + 15\delta_{30}(x)$

with $\delta_i(x)$ is function of degree 3

$\delta_i(x) =?$

$$\delta_i(x) = \begin{cases} 1 \text{ if } x = i \\ 0 \text{ if } x \in \{5,7,12,30\} - \{i\} \\ \text{"any" in other case} \end{cases}$$

$$\delta_{30}(x) = \frac{x-5}{30-5}\frac{x-7}{30-7}\frac{x-12}{30-12}$$

Lagrange interpolation

Example:

$$f(x) = \frac{7}{6}x^2 - \frac{19}{6}x + 1$$

x = 0 ➔ y = ?

x = 1 ➔ y = ?

x = 3 ➔ y = ?

Lagrange interpolation

Example:

$$f(x) = \frac{7}{6}x^2 - \frac{19}{6}x + 1$$

x = 0 → y = 1

x = 1 → y = -1

x = 3 → y = 2

Lagrange interpolation

Example: find function f(x) go through 3 points: (0,1), (1,-1), (3,2)

$$f(x) = \sum_{i=0}^{n} \delta_{x_i}(x) \times y_i$$

$$\delta_{x_i}(x) = \frac{\prod_{k=0,k\neq i}^{n}(x - x_k)}{\prod_{k=0,k\neq i}^{n}(x_i - x_k)}$$

Demo ...

Extending the Shamir method to multi-digit secret information (eg, images)

The simplest way is to apply Shamir method to each number of secret information

Content

- Shamir method with finite field: solve the problem of incorrect representation of real numbers and overflow when calculating on computers
- Extended Shamir method for secret image sharing problem

Demo on the problem of incorrect representation of real numbers and overflow when calculating on a computer...

One solution: use a finite field

- (Field) is a set on which :
  - Addition and subtraction operations (plus the number with the opposite sign), multiplication, division (multiply by the inverse) is defined (the result of these calculations must also be in the set of the field which has the potential to avoid the overflow problem)
  - And has properties like calculations on real numbers
- Is an integer a field?
  - Nope, because there are integers when divided by each other, the result is not an integer
- A real number is a field with infinitely many elements
  - Computers can only represent a finite number of elements ☹
- Is there a field that has a finite number of elements and is sufficient for a computer to represent? ;-)

Which field is a finite field?

- Consider the set $\mathbb{Z}_{100}$ including 100 interger 0, 1, …, 99 with operations defined as follows: :
  - a plus b: $a + b$ mod 100
  - a times b: $a \times b$ mod 100
  - a minus b: a plus $-b$ with $-b \in \mathbb{Z}_{100}$ and $-b$ plus $b = 0$

    (equivalent definition : $a - b$ mod 100)
  - a divided by b: a times b^{-1} with $b^{-1} \in \mathbb{Z}_{100}$ and b^{-1} times $b = 1$
- $\mathbb{Z}_{100}$ is a field?
  - The inverse of 3?

    67
  - The inverse of 5?

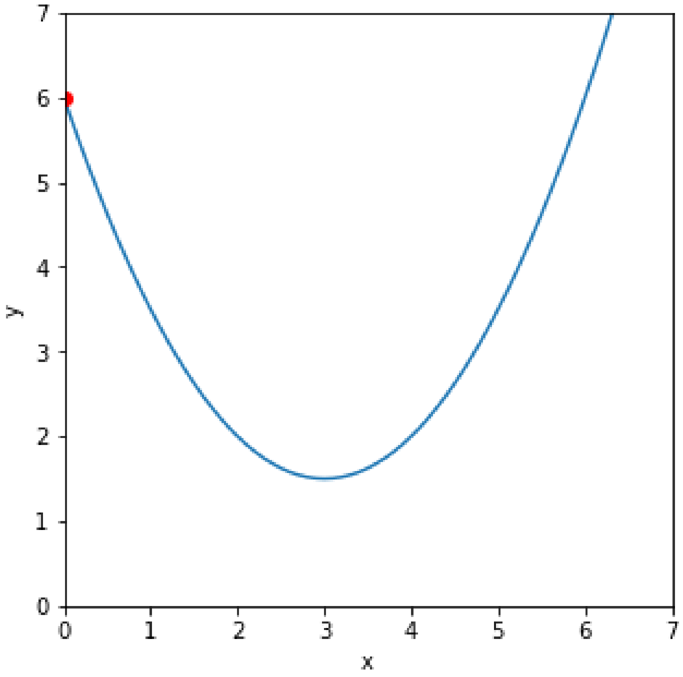
    None → $\mathbb{Z}_{100}$ is not a field

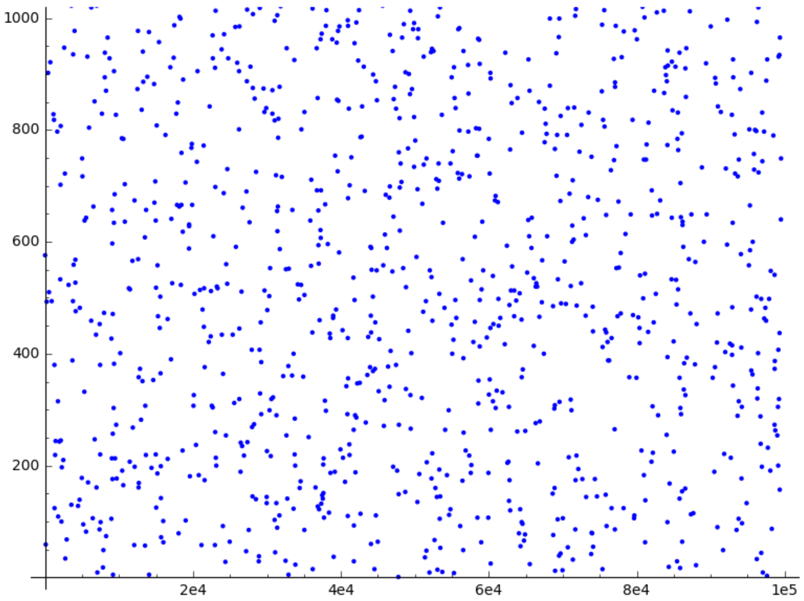
Which field is a finite field?

- It has been proven : $\mathbb{Z}_p$ is a field if and only if p is prime number
- Use field $\mathbb{Z}_p$ for Shamir's method will avoid the problem of incorrect calculation on the computer
  - Choose p small enough that the computer can represent elements of the field
  - Choose p large enough to avoid the problem that the field has too few elements, the attacker can try each element
  - Secret information S and number of parts n must be interger ∈ $[0, p)$
- In fact, in Shamir's original paper, finite fields $\mathbb{Z}_p$were used

Which field is a finite field?

- The normal curve vs finite field curve





Demo implementation of Shamir's method with finite field...

Content

- Shamir method with finite field: solve the problem of incorrect representation of real numbers and overflow when calculating on computers
- Extended Shamir method for secret image sharing problem

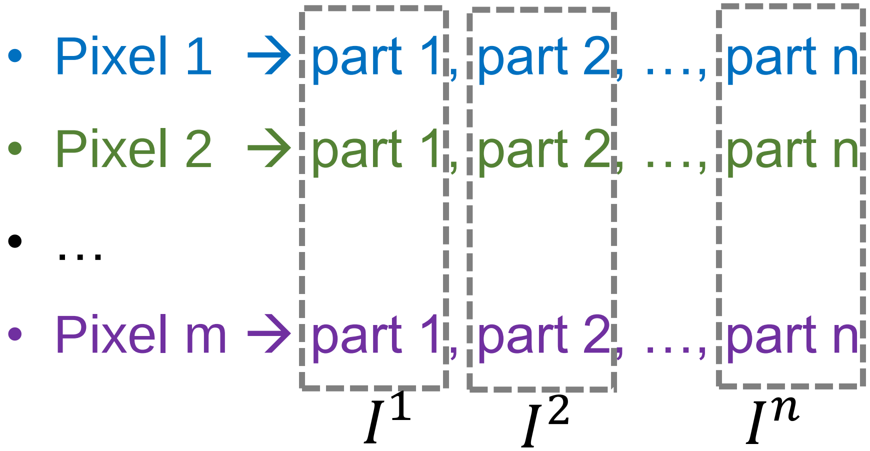
The problem of sharing secret photos

Share secret image I (grayscale, each pixel is an integer $\in [0, 255]$), devide into n parts $I^1, I^2, \ldots, I^n$ such that:

- With at least any k parts ($k \leq n$), we can recreate I
- With less than k parts, we don't know anything about I

Extending Shamir method for secret image sharing problem?

- A simple way is to apply Shamir's method to each pixel in the secret image I (assume image I have m pixel)



  - Pixel 1 → part 1, part 2, …, part n
  - Pixel 2 → part 1, part 2, …, part n
  - …
  - Pixel m → part 1, part 2, …, part n

- What is the size of each part I^i compare to image I?
- If using field $\mathbb{Z}_p$ how we can choose p?
- If you want each element in I^i still be a integer $\in [0, 255]$ then how to choose p?

Thien & Lin's method(*) for the problem of sharing secret images

- Also based on the method of Shamir
- Each part I^i has a smaller number of elements than the image I
- Elements in I^i still a integer ∈ $[0, 255]$
- Thien & Lin propose 2 version: lossy and lossless

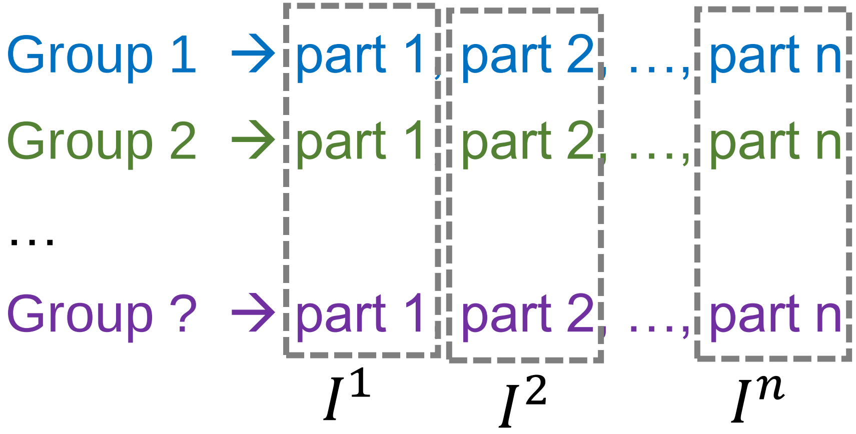
(*): Thien & Lin, Secret image sharing, 2002

The lossy version of Thien & Lin

The process of sharing secret images

- Use $p = 251$, is the largest prime number ≤ 255
- The pixel with value > 250, change to 250
- Traverse the pixels in a certain order, every k pixels will form a group (does not intersect with other groups). For each group of k pixels $p_1, p_2, \dots, p_k$:
  - Generate polynomials: $f(x) = p_1 + p_2 x + \cdots + p_k x^{k-1}$
  - From the above polynomial, generate n parts with Shamir method

Thus, we have :



Group 1 → part 1, part 2, …, part n

Group 2 → part 1, part 2, …, part n

…

Group ? → part 1, part 2, …, part n

I^1 I^2 I^n

What size of I^i compare to I?
Equal to I divided by k ☺

The lossy version of Thien & Lin

The process of recreating secret images

With n' part I^i , just need any k part - I^i to recreating secret image

- Iterate over first k elements of I^i part, second k elements of I^i part, …

  With each iteration: reconstruct the polynomial function of degree $k-1$ from k element of k parts I^i, k coefficient of this polynomial is k pixel of secret image
- Once all the pixels of the secret image have been obtained, rearrange the order of these pixels based on the order iteration when dividing.

Thien & Lin's lossless version

The process of sharing secret images

Same as the lossy version, with minor different:

- There is no conversion of pixels that are > 250 ➔ 250
- Instead of working on the array of pixels of, it will convert the array of pixels of to a different array and will working on this other array

How to convert: iterate array of pixels of I:

- If pixel value $<$ 250: write the pixel value to the corresponding position of other array
- If pixel value $\geq$ 250: split the pixel value into 2 values of 250 and the rest, and then write these 2 values in 2 corresponding positions of other array

  → The number of elements of other array $\geq$ the number of elements of I

Thien & Lin's lossless version

The process of recreating secret images

- If we follow the steps of the lossy version correctly, we will eventually get other array
- To get the array of pixels of, we will traverse the elements in this other array:
  - If the element has value $\neq$ 250: write this value to the corresponding position in the array I
  - If the element has the value $=$ 250: get the next element in other array, Add these 2 values and write the result to the corresponding position in the array I