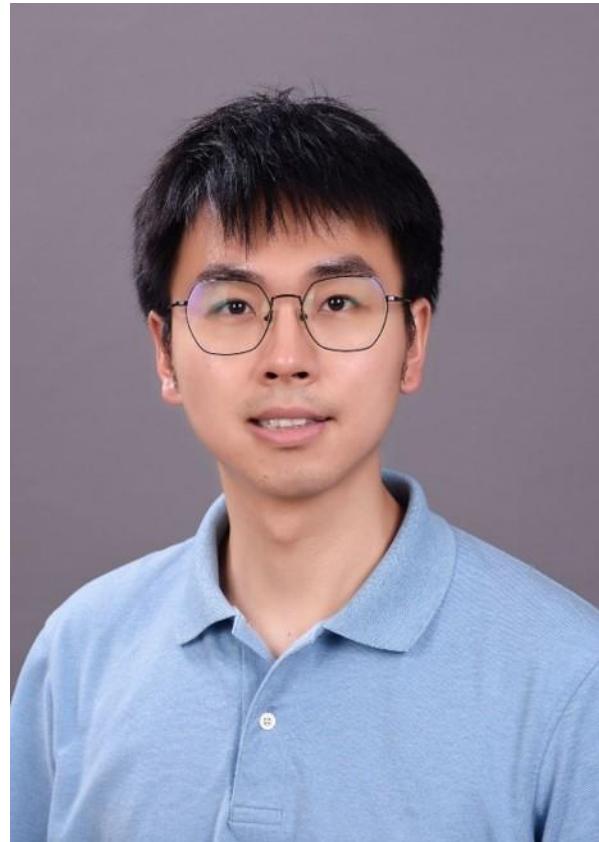
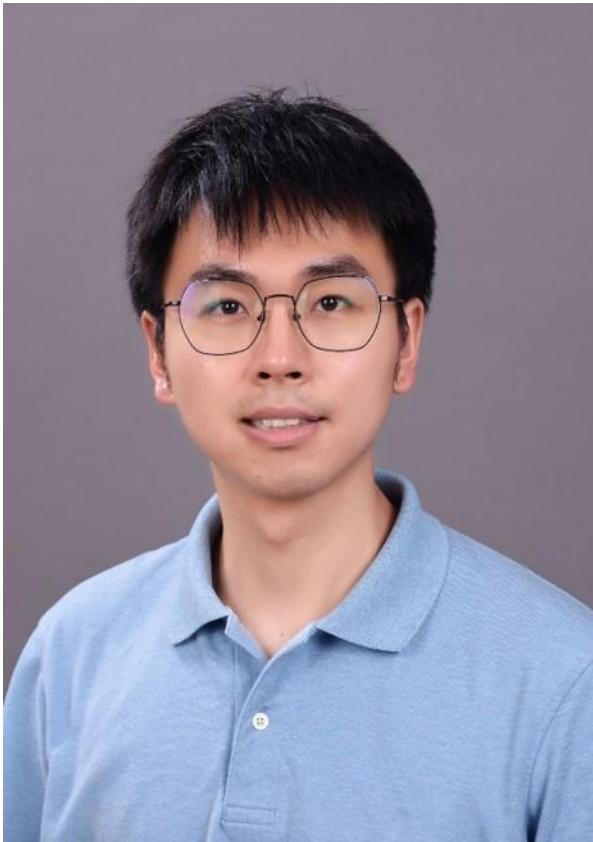


# »» 这两幅图有区别吗?



右边的图中有我给大家传递的秘密信息



西安交通大学  
XI'AN JIAOTONG UNIVERSITY

# 第二讲： 计算机的表示与计算

Bin Shi 师斌

School of Computer Science & Technology,  
Xi'an Jiaotong University

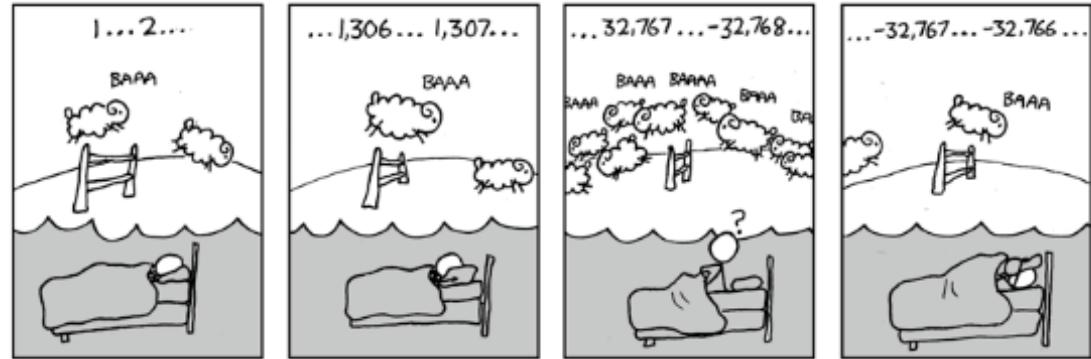


# >> 不可忽视的真像

## Ints are not Integers, Floats are not Reals

- Example 1: Is  $x^2 \geq 0$ ?

- Float's: Yes!



- Int's:

- $40000 * 40000 \rightarrow 1600000000$
    - $50000 * 50000 \rightarrow ?$

Example 1: Is  $(x + y) + z = x + (y + z)$ ?

- Unsigned & Signed Int's: Yes!
- Float's
  - $(1e20 + -1e20) + 3.14 \rightarrow 3.14$
  - $1e20 + (-1e20 + 3.14) \rightarrow ??$

# » 主要内容

- 数值性信息的表示与计算
- 非数值性信息的表示与计算

# »» 进位计数制

## 进位计数制与十进制

◆进位计数制：用数码和带有权值的数位来表示有大小关系的数值性信息的表示方法。

- ◆十进制
  - 有0,1,2,3,4,5,6,7,8,9共十个数码
  - 数码的位置规定了数码的等级 “权/数位” :  $10^i$
  - 逢十进一、借一当十。高数位的1相当于低数位的10
  - “十” ----基值，十进制

$$(2 \quad 4 \quad 5 \ . \ 2 \quad 5)_{+}$$
$$\begin{array}{cccccc} | & | & | & | & | \\ 2 & 1 & 0 & -1 & -2 \\ 10^2 & 10^1 & 10^0 & 10^{-1} & 10^{-2} \end{array}$$

$$(245.25)_{+} =$$
$$2 \times 10^2 + 4 \times 10^1 + 5 \times 10^0 + 2 \times 10^{-1} + 5 \times 10^{-2}$$

# >> 进位计数制

## 二进制

- 有0,1共两个数码
- 数码的位置规定了数码的等级 “权/数位” :  $2^i$
- 逢二进一、借一当二。高数位的1相当于低数位的2
- “二” ----基值，二进制

$2^7 \ 2^6 \ 2^5 \ 2^4 \ 2^3 \ 2^2 \ 2^1 \ 2^0 . \ 2^{-1} \ 2^{-2}$  ————— 数位的权值  
7 6 5 4 3 2 1 0 . -1 -2 ————— 数位

例如：  $(1\ 1\ 1\ 1\ 0\ 1\ 0\ 1\ .\ 0\ 1)_2$  ————— 二进制数

$$\begin{aligned} &= 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 \\ &\quad + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = (245.25)_+ \end{aligned}$$

# >> 进位计数制

## r进制转换为十进制

已知r进制的数码串 $d_m \dots d_{-n}$ , 求十进制的N

$$N = (d_{n-1}d_{n-2}\dots d_2d_1d_0.d_{-1}d_{-2}\dots d_{-m})_r \\ = d_{n-1}r^{n-1} + d_{n-2}r^{n-2} + \dots + d_2r^2 + d_1r^1 + d_0r^0 + d_{-1}r^{-1} + d_{-2}r^{-2} + \dots + d_{-m}r^{-m} = \sum_{i=-m}^{n-1} d_i r^i$$

$$\begin{array}{r} 7 \ 6 \ 5 \ 4 \ 3 \ 2 \ 1 \ 0 \ . -1 \ -2 \\ \hline (1 \ 1 \ 1 \ 1 \ 0 \ 1 \ 0 \ 1 \ . \ 0 \ 1)_2 \\ = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 \\ + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = (245.25)_+ \\ (F5.4)_{16} = F \times 16^1 + 5 \times 16^0 + 4 \times 16^{-1} = (245.25)_+ \end{array}$$

# >> 进位计数制

## 十进制转换为r进制—整数部分转换

已知十进制整数N，求r进制的数码串 $d_{n-1}d_{n-2}\dots d_1d_0$

$$\begin{aligned} N &= (d_{n-1}d_{n-2}\dots d_2d_1d_0)_r \\ &= d_{n-1}r^{n-1} + d_{n-2}r^{n-2} + \dots + d_2r^2 + d_1r^1 + d_0r^0 \end{aligned}$$

(N/r)的余数为  $d_0$

((N/r)/r)的余数为  $d_1$

(((N/r)/r)/r)的余数为  $d_2$

...

(...(((N/r)/r)/r).../r)的余数为  $d_{n-1}$

“除基取余”

换成16进制：

$$(245)_+ = (F5)_{\text{十六}}$$

$$\begin{array}{r} 16 \longdiv{245} \\ 16 \quad \boxed{15} \cdots 5 \\ 0 \cdots 15(F) \end{array}$$

# >> 进位计数制

## 十进制转换为r进制—小数部分转换

已知十进制小数N，求r进制的数码串 $0.d_{-1}d_{-2}\dots d_{-m}$

$$N = (0.d_{-1}d_{-2}\dots d_{-m})_r = d_{-1} r^{-1} + d_{-2} r^{-2} + \dots + d_{-m} r^{-m}$$

$(N \times r)$ 的整数部分为 $d_{-1}$

$((N \times r) \times r)$ 的整数部分为 $d_{-2}$

$((((N \times r) \times r) \times r))$ 的整数部分为 $d_{-3}$

....

$(\dots(((N \times r) \times r) \dots \times r))$ 的整数部分为 $d_{n-1}$

注：每次相乘都是去掉整数后的小数部分相乘。

若最终为0，则结束，否则计算到要求位数即可。

换成16进制：

$$(0.525)_+ \\ = (0.8666)_{\text{十六}}$$

$$\begin{array}{r} 0.525 \\ \times 16 \\ \hline 8 .4 \\ \times 16 \\ \hline 6 .4 \\ \times 16 \\ \hline 6 .4 \\ \times 16 \\ \hline 6 .4 \end{array}$$

“乘基取整”



# >> 思考

了解了进制转换后，问：78在计算机中如何表示？

- (1)  $(78)_{+} = (1001110)_{-}$  未考虑机器字长
- (2)  $(78)_{+} = (01001110)_{-}$  8位机器字长
- (3)  $(78)_{+} = (00000000\ 01001110)_{-}$  16位机器字长
- (4)  $(78)_{+} = (00000000\ 00000000\ 00000000\ 01001110)_{-}$  32位机器字长

## 机器字长

【机器字长】是指机器内部进行数据处理、信息传输等的基本单元所包含的二进制位数，通常是8位、16位、32位、64位等。



# >> 有符号数和无符号数

## 机器数：有符号数和无符号数

16位机器字长

无符号数

无符号机器数的表示范围： $0 \leq X < 2^{16}$ ，即0至65,535

(00000000 00000000) = —— (11111111 11111111) =

有符号  
数

数的符号也用0和1表达，0表“+”号，1表“-”号

(00000000 00000000) = (01111111 11111111) =

(10000000 00000000) = (11111111 11111111) =

有符号数的  
表示范围？

# >> 计算机的计算

## 4位机器字长（无符号数）

$$\begin{array}{r} 7 + 3 = 10 \\ \hline 0111 \\ + 0011 \\ \hline 1010 \end{array}$$

$10 - 3 = 7$  怎么算?  
能用加法实现吗?

$$10 - 3 = 10 + 16 - 3 - 16 = 10 + 13 - 16$$

$$\begin{array}{r} 10 + 13 - 16 = 7 \\ \hline 1010 \\ + 1101 \\ \hline 0111 \end{array}$$

受字长限制自动舍弃

# >> 计算机的计算

## 带符号的机器数：原码、反码和补码



# >> 计算机的计算

## 补码的加减法：符号位与数值位一起参与运算

5位机器字长（1位符号位，4位数值位）

$$(+7) + (+3) = (+10)$$

0 0111

+ ) 0 0011

0 1010

$$(10) + (-3) = (7)$$

0 1010

+ ) 1 1101

0 0111

$$(-5) + (-7) = (-12)$$

1 1011

+ ) 1 1001

1 0100

受字长限制自动舍弃

$$(-7) + (-12) = \text{溢出}$$

1 1001

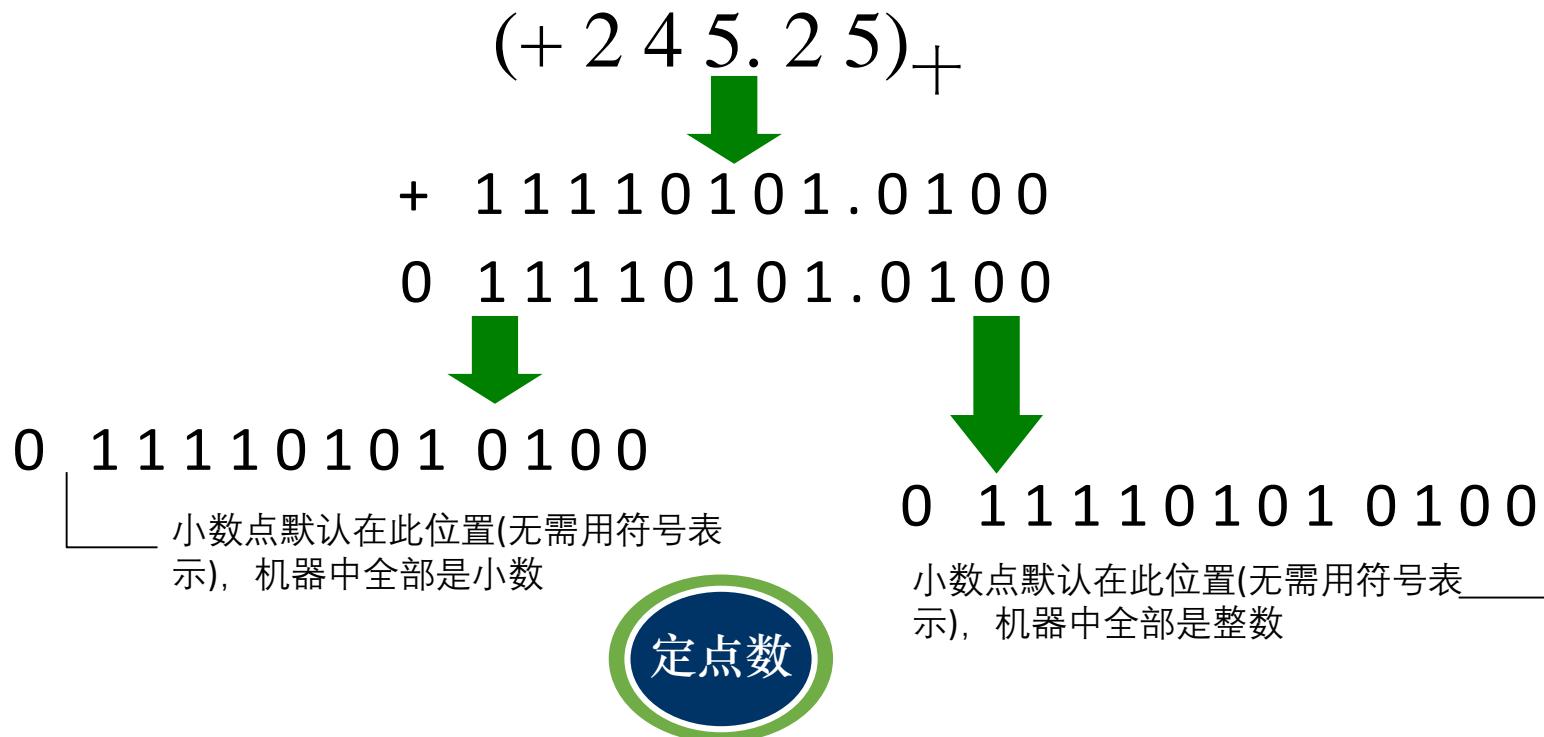
+ ) 1 0100

0 1101

补码再求补  
即为原码

# >> 小数的处理

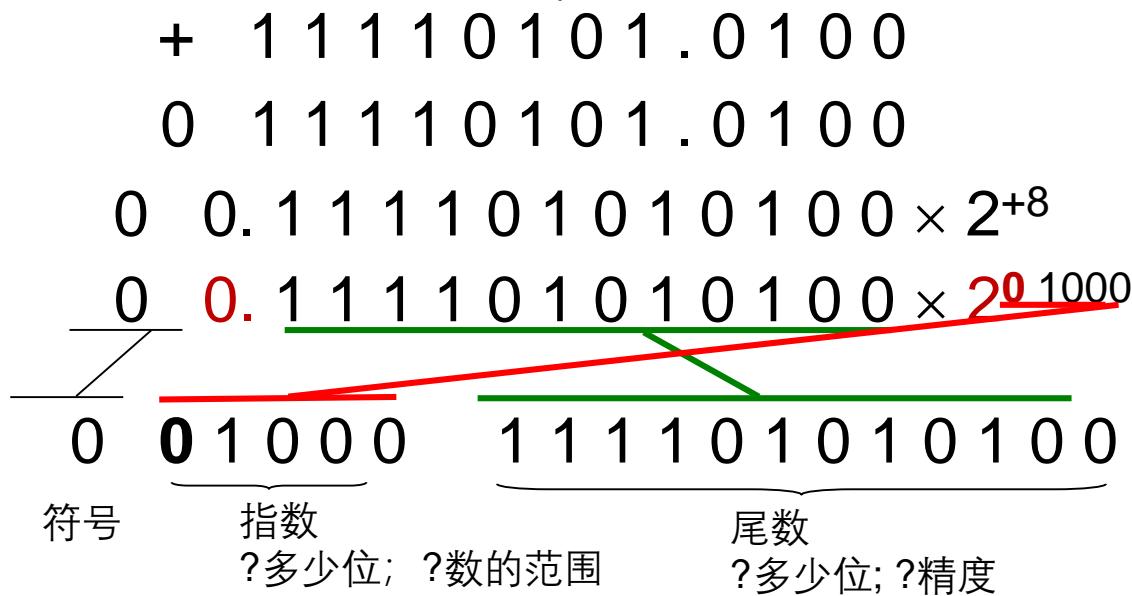
## 定点数：纯小数 vs. 纯整数



# >> 小数的处理

## 浮点数：小数点位置的处理

$$(+\ 2\ 4\ 5.\ 2\ 5)_+ = 0.24525 \times 10^3$$



# >> 小数的处理

## 浮点数：单精度数 vs. 双精度数

S	指数(8位)	尾数(后23位)
---	--------	----------

浮点数，32位表示单精度数(相当于科学计数法 $1.x \times 2^y$ )  
(S为符号位, x为23位尾数, y为8位指数)

S	指数(11位)	尾数(后52位)
---	---------	----------

浮点数，64位表示双精度数(相当于科学计数法 $1.x \times 2^y$ )  
(S为符号位, x为52位尾数, y为11位指数)



能否去掉指数  
的符号

# » 0和1的计算：基本逻辑运算

## 命题逻辑的符号化与计算

用0和1来表示逻辑运算

- 【AND】：“与”运算  
有0为0，全1为1
- 【OR】：“或”运算  
有1为1，全0为0
- 【NOT】：“非”运算  
非0则1，非1则0
- 【XOR】：“异或”运算  
相同为0，不同为1

注：1表示 真，0表示 假

$\begin{array}{r} 0 \\ \text{AND} \\ \hline 0 \end{array}$	$\begin{array}{r} 0 \\ \text{AND} \\ \hline 0 \end{array}$	$\begin{array}{r} 1 \\ \text{AND} \\ \hline 0 \end{array}$	$\begin{array}{r} 1 \\ \text{AND} \\ \hline 1 \end{array}$
$\begin{array}{r} 0 \\ \text{OR} \\ \hline 0 \end{array}$	$\begin{array}{r} 0 \\ \text{OR} \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ \text{OR} \\ \hline 0 \end{array}$	$\begin{array}{r} 1 \\ \text{OR} \\ \hline 1 \end{array}$
$\begin{array}{r} 0 \\ \text{NOT} \\ \hline 1 \end{array}$		$\begin{array}{r} 1 \\ \text{NOT} \\ \hline 0 \end{array}$	
$\begin{array}{r} 0 \\ \text{XOR} \\ \hline 0 \end{array}$	$\begin{array}{r} 0 \\ \text{XOR} \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ \text{XOR} \\ \hline 0 \end{array}$	$\begin{array}{r} 1 \\ \text{XOR} \\ \hline 0 \end{array}$

# » 0和1的计算：基本逻辑运算

二进制加法运算可用逻辑计算来实现

二进制的加法运算规则

$$\begin{array}{r} 0 & 1 & 0 & 1 \\ + 0 & + 0 & + 1 & + 1 \\ \hline 0 & 1 & 1 & 0 \end{array}$$

不考虑进位

$$\left\{ \begin{array}{l} S_i = A_i \text{ XOR } B_i \\ C_{i+1} = A_i \text{ AND } B_i \end{array} \right.$$

$$\begin{array}{c} A_i \\ B_i \\ \hline C_{i+1} & S_i \end{array}$$

考虑进位

$$\left\{ \begin{array}{l} S_i = (A_i \text{ XOR } B_i) \text{ XOR } C_i \\ C_{i+1} = ((A_i \text{ XOR } B_i) \text{ AND } C_i) \text{ OR } (A_i \text{ AND } B_i) \end{array} \right.$$

$$\begin{array}{c} A_i \\ B_i \\ C_i \\ \hline C_{i+1} & S_i \end{array}$$

启示

乘除可转换为加减，减法又可转换为加法，而加法又可以用逻辑运算的组合来实现。  
机器只要能完成加法运算，再由一个可执行组合的机构，就可以实现任何运算。

# » 主要内容

- 数值性信息的表示与计算
- 非数值性信息的表示与计算

# » 非数值性信息的表达：编码与组合

## 非数值性信息表达：编码

编码

编码是以若干位数码或符号的不同组合来表示非数值性信息的方法，它是人为地将若干位数码或符号的每一种组合指定一种唯一的含义。

ASCII码

英文字母与符号的0, 1型编码方法

规律性：英文字母与符号是按照二进制次序进行编码，连续排列

$b_6 b_5 b_4 b_3 b_2 b_1 b_0$   
**0** $b_6 b_5 b_4 b_3 b_2 b_1 b_0$

机器字长8  
位及其倍数

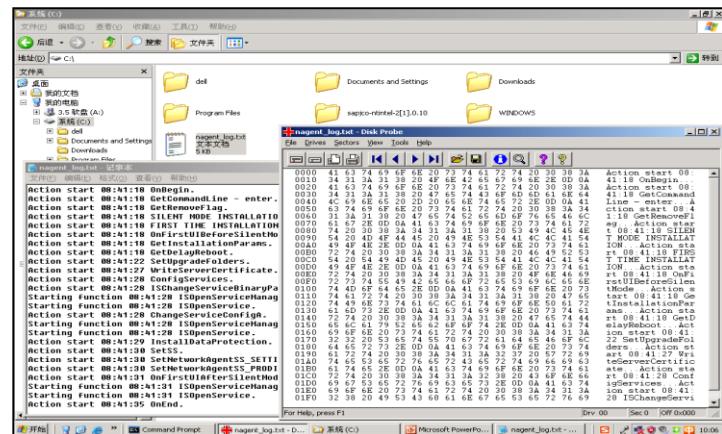
$b_6 b_5 b_4$ $b_3 b_2 b_1 b_0$	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	.	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	:	K	[	k	{
1100	FF	FS	,	<	L	]	l	
1101	CR	GS	-	=	M	\	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	-	o	DEL

# » 非数值性信息的表达：编码与组合

ASCII码存储的文件，即纯文本文件，是可以自动解读的

信息	机器内部存储	解析规则	十六进制显示存储内容
We are students	01010111 01100101 00100000 01100001 01110010 01100101 00100000 01110011 01110100 01110101 01100100 01100101 01101110 01110100 01110011	0/1串按8位分隔一个字符，查找ASCII码表映射成相应符号	57 65 20 61 72 65 20 73 74 75 64 65 6E 74 73

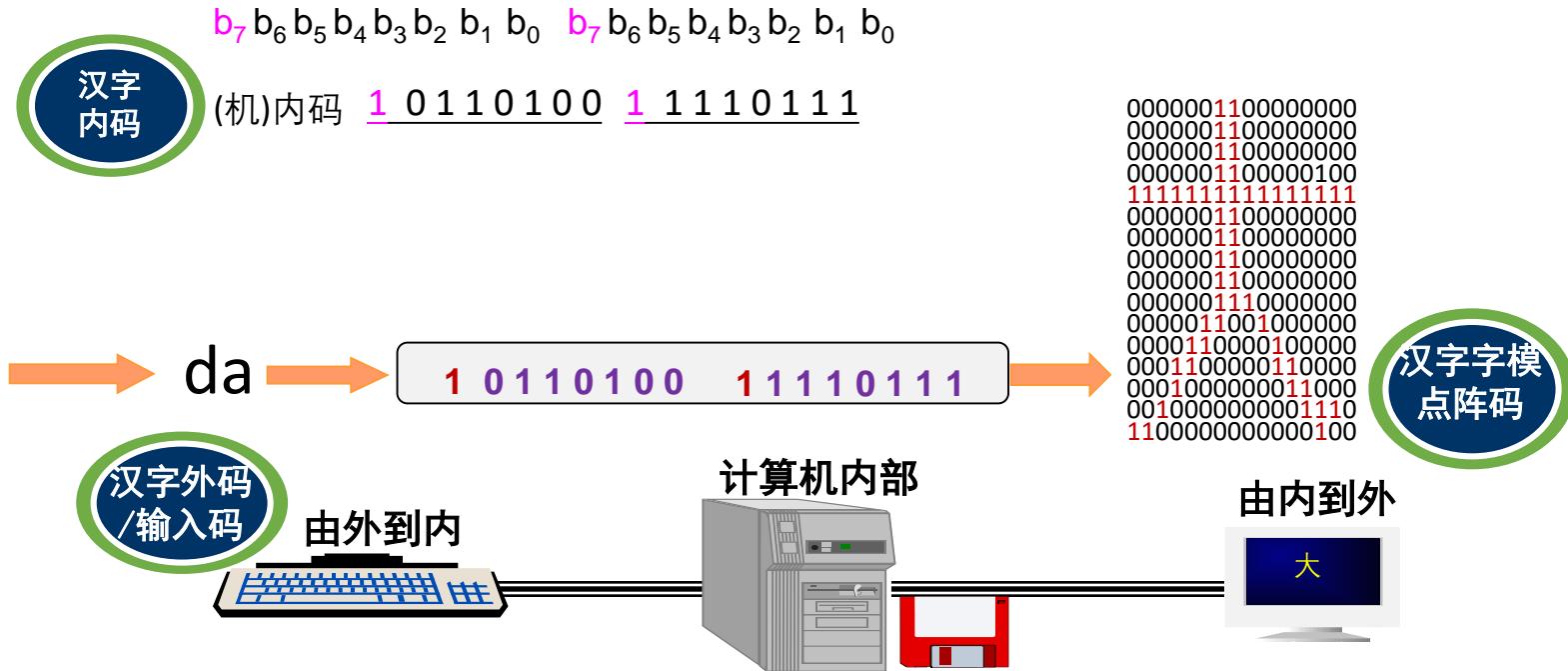
b <sub>5</sub> b <sub>4</sub> b <sub>3</sub> b <sub>2</sub> b <sub>1</sub> b <sub>0</sub>	000	001	010	011	100	101	110	111
0000	NUL	DLE	SP	0	@	P	'	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	.	7	G	W	g	w
1000	BS	CAN	<	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	:	K	[	k	{
1100	FF	FS	,	<	L	]	l	
1101	CR	GS	-	=	M	\	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	-	o	DEL



# » 非数值性信息的表达：编码与组合

## 汉字的处理过程：由外码、到机器内码、到字形码

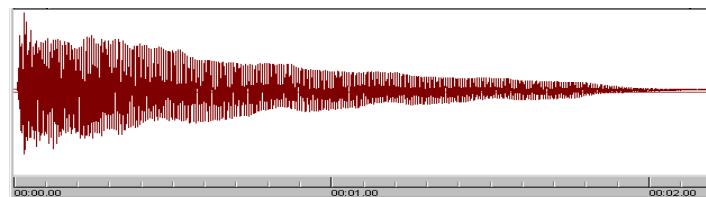
用0和1编码汉字，每个汉字在计算机内部由 2个字节表示



# Representation

## ➤ Audio

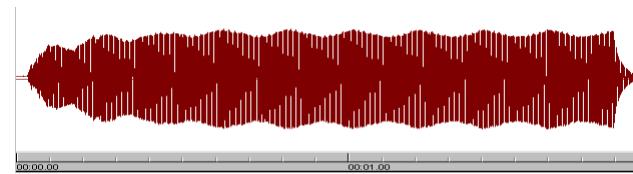
- A vibration that propagates as a typically audible mechanical wave of pressure and displacement, through a medium such as air or water.



Piano



Snare drum

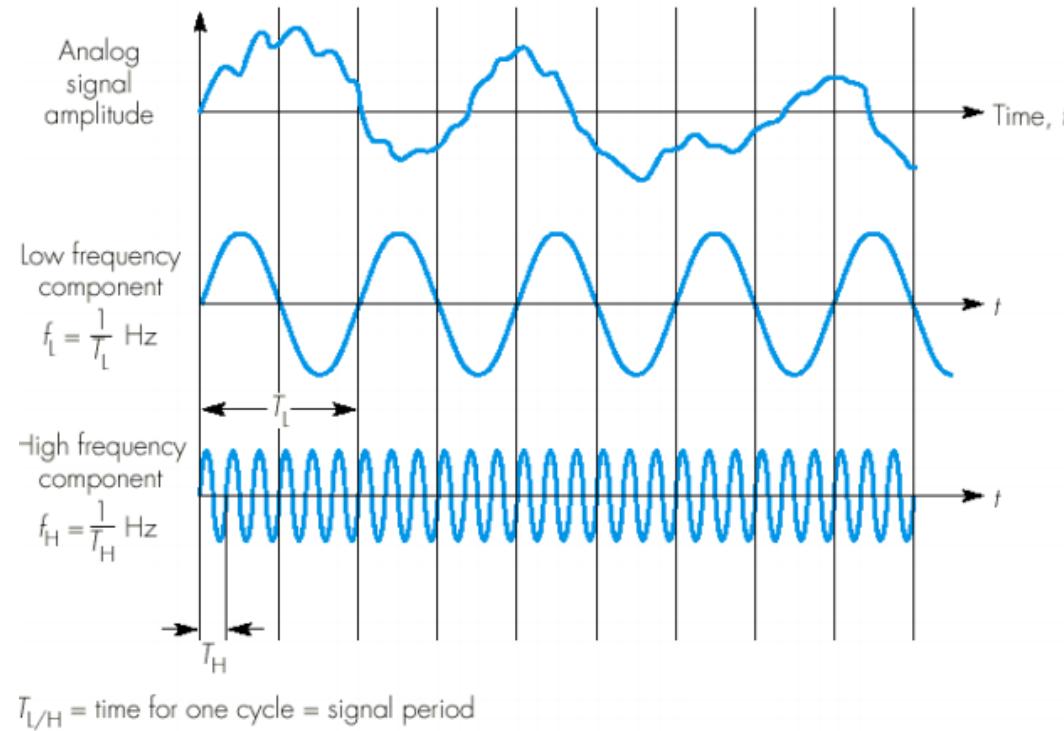


Pan flute



# Representation

- **Analog signal:** Sight and sound we perceive in our natural world are continuous.

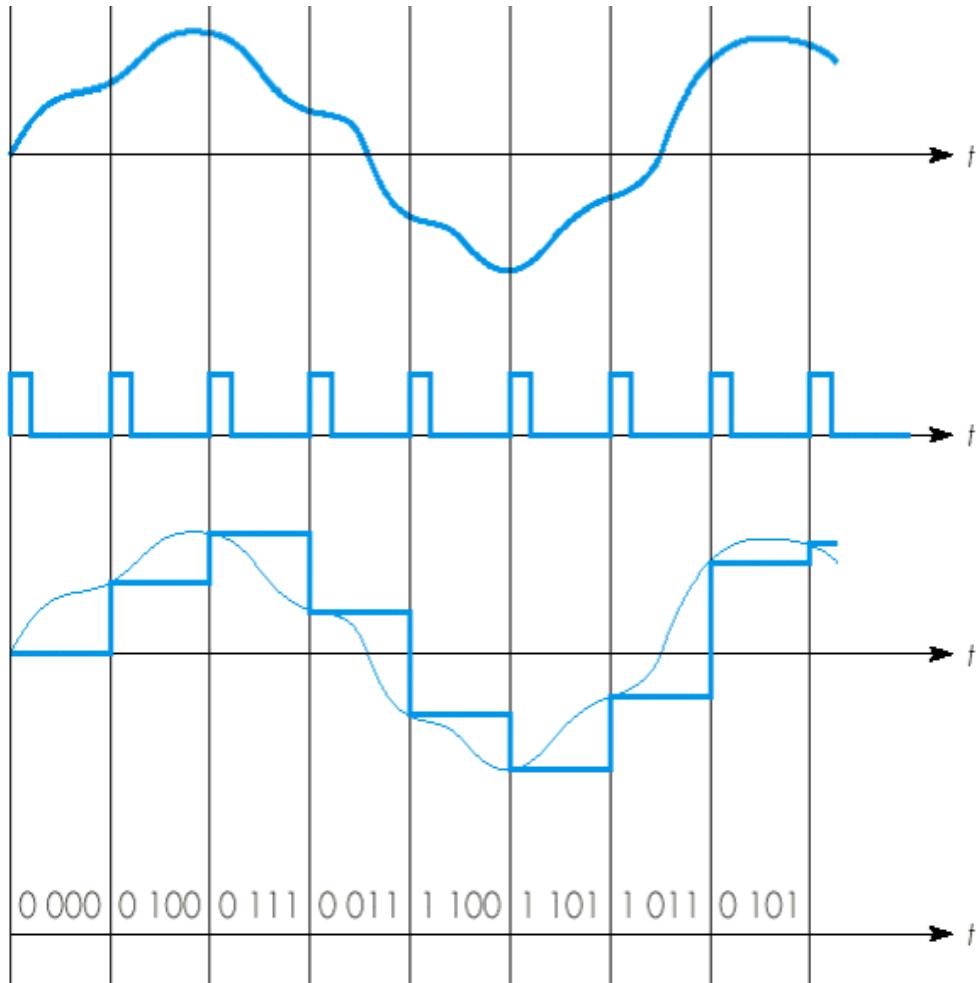


Continuous and infinite number of points between any two points.

# Representation

- **Issue:** natural medias are analog continuous signals, however, computers handle discrete digital data only, therefore, how to realize analog-to-digital conversion is a fundamental issue to represent multimedia information on computer.
- **Method:** Sampling and Quantization.

# Representation



Step1. Sampling:  
Obtain the corresponding  
amplitude value in fixed  
sampling time interval.

Step2. Quantization:  
Confines the amplitude of a signal  
into a finite number of values.

# Representation

## ➤ Capture and playback of digital audio

Air pressure variations



Captured via microphone



Analogue to Digital Converter

ADC

0101001101  
0110101111

Convert back into voltage

DAC



Air pressure variations

# Representation

- Image---A group of pixels.
- All images are displayed in the form of a two-dimensional matrix of individual picture elements called **pixels**.

# Representation

## ➤ Image---Pixel

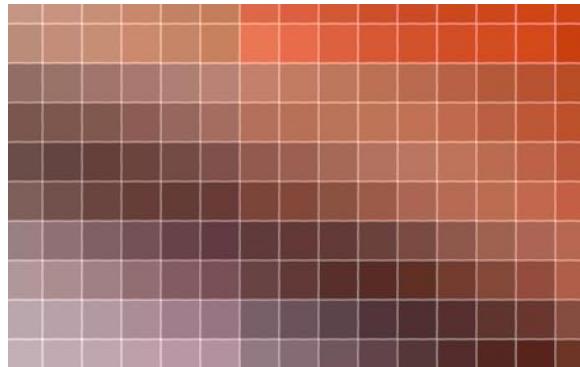
This photo looks like contains smooth variations in shading, but...



2



1



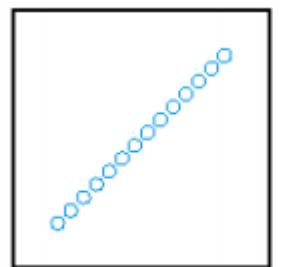
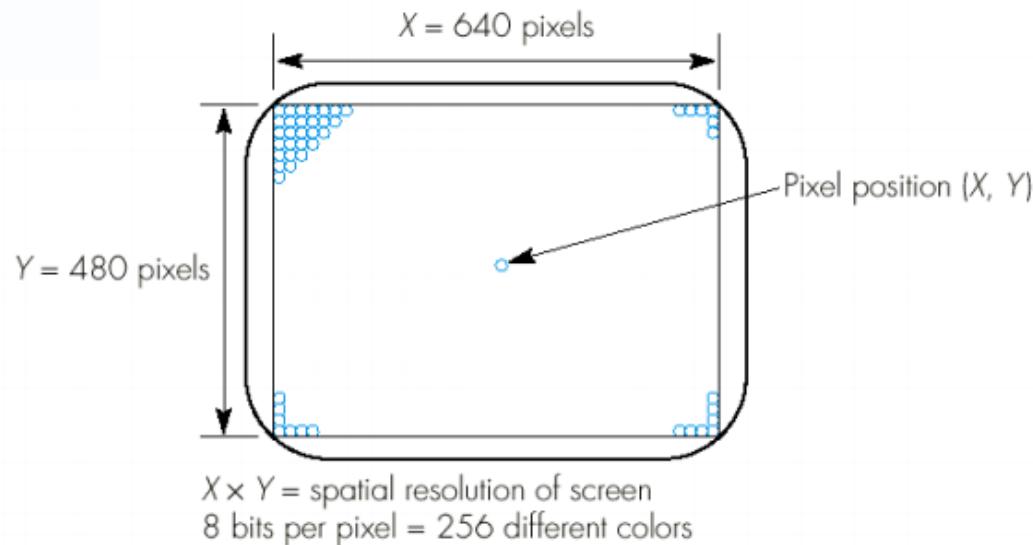
...if we zoom in on his eye, we can see that it is not completely smooth...

3

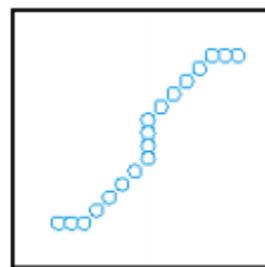
...and we zoom in even more, we can see that the image is made up of lots of small squares, so called **pixels**.

# Representation

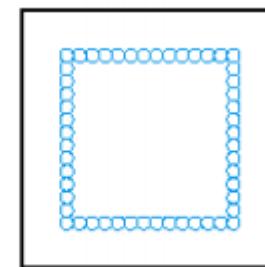
## ➤ Image---Pixel



Line



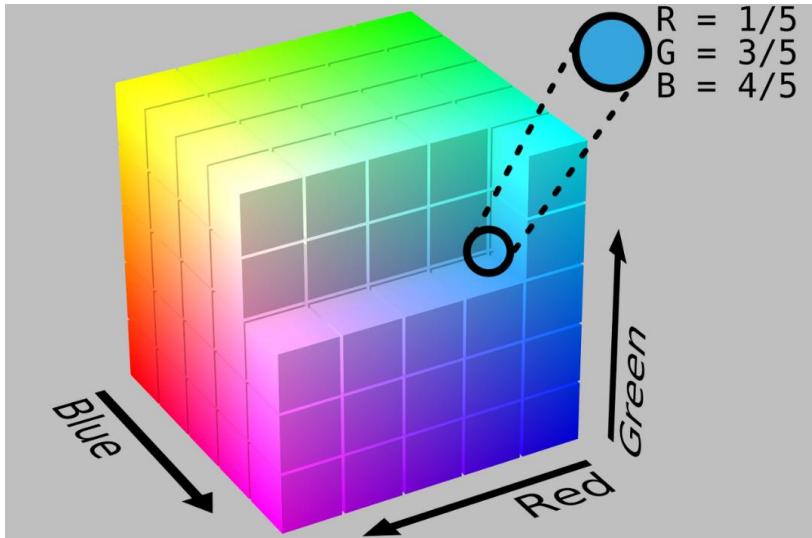
Curve



Square

# Representation

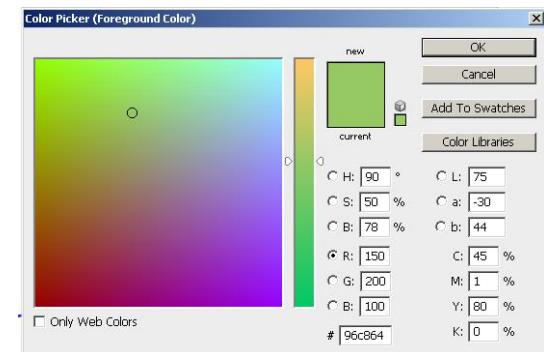
- Each pixel has color. RGB、YUV、CMYK、HSB、CIE and their variants are common color models.



RGB Color

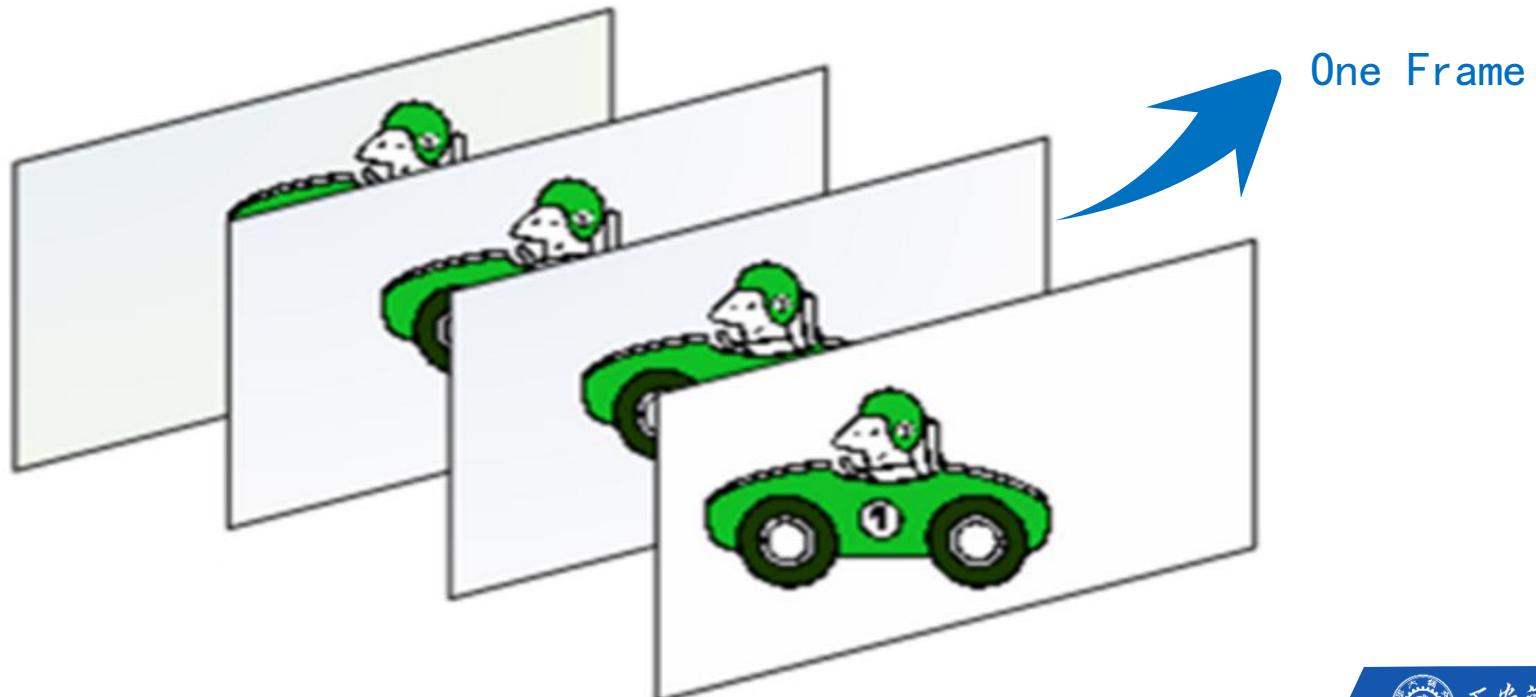
RGB color space is defined by the three chromaticity of the red, green, and blue, and can produce any chromaticity that is the triangle defined by those primary colors.

Color Picker



# Representation

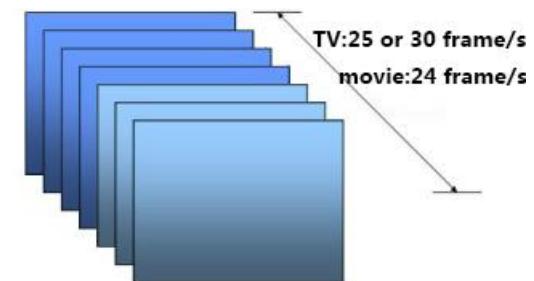
- Video: a group of images. Video combines a sequence of images to form a moving picture. It usually has audio components that correspond with the picture being shown on the screen.



# Representation

- Video--- Frame rate
  - Frame rate is the **frequency** at which an imaging device displays consecutive images called **frames**.
  - Frame rate should be higher than **15frame/s** . Why ???

Persistence of vision works because the human eye and brain can only process 10 to 12 separate images per second, retaining an image for up to 15 of a second. If a subsequent image replaces it in this period of time it will create the illusion of continuity.



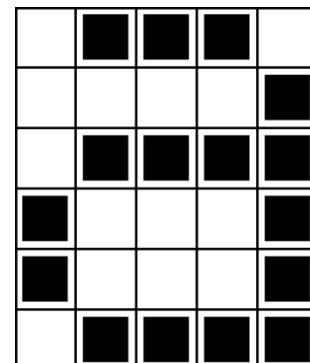
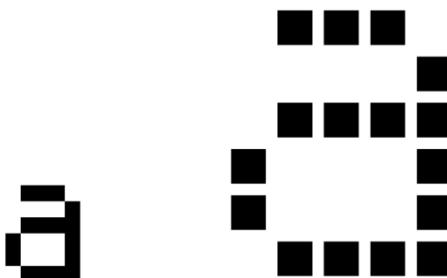
TV: 30 or 25 frames/s  
Movie: 24 frames/s

- Compression is necessary.

Size of <b>1min</b> video file				
Frame Rate	Frame Size	Bits per Pixel	Bit Rate (bps)	File Size (Byte)
25	1920*1080	12		



- Compression is possible.
- For an image, adjacent pixels are correlated.



0, 1, 1, 1, 0  
0, 0, 0, 0, 1  
0, 1, 1, 1, 1  
1, 0, 0, 0, 1  
1, 0, 0, 0, 1  
0, 1, 1, 1, 1

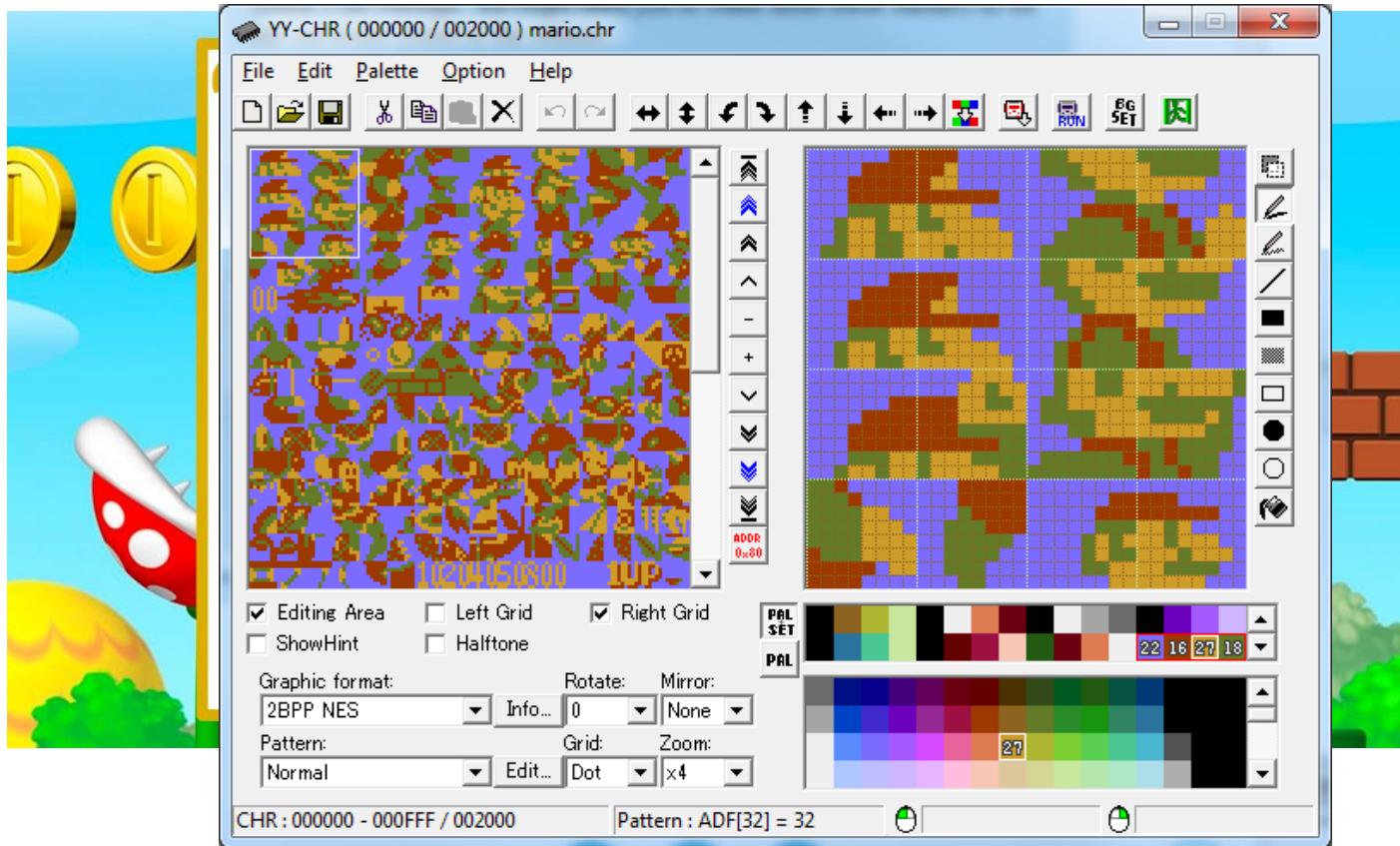
No Compression  
(0=white 1=black)



1, 3, 1  
4, 1  
1, 4  
0, 1, 3, 1  
0, 1, 3, 1  
1, 4

Compressed  
(Start with white)

- Compression is possible.
- In an image, some patterns are the same .



# Super Mario examples

Truth or Fiction?

Seeking truth and exposing fiction since 1999

MARIO  
005400

• x22

WORLD  
1-1

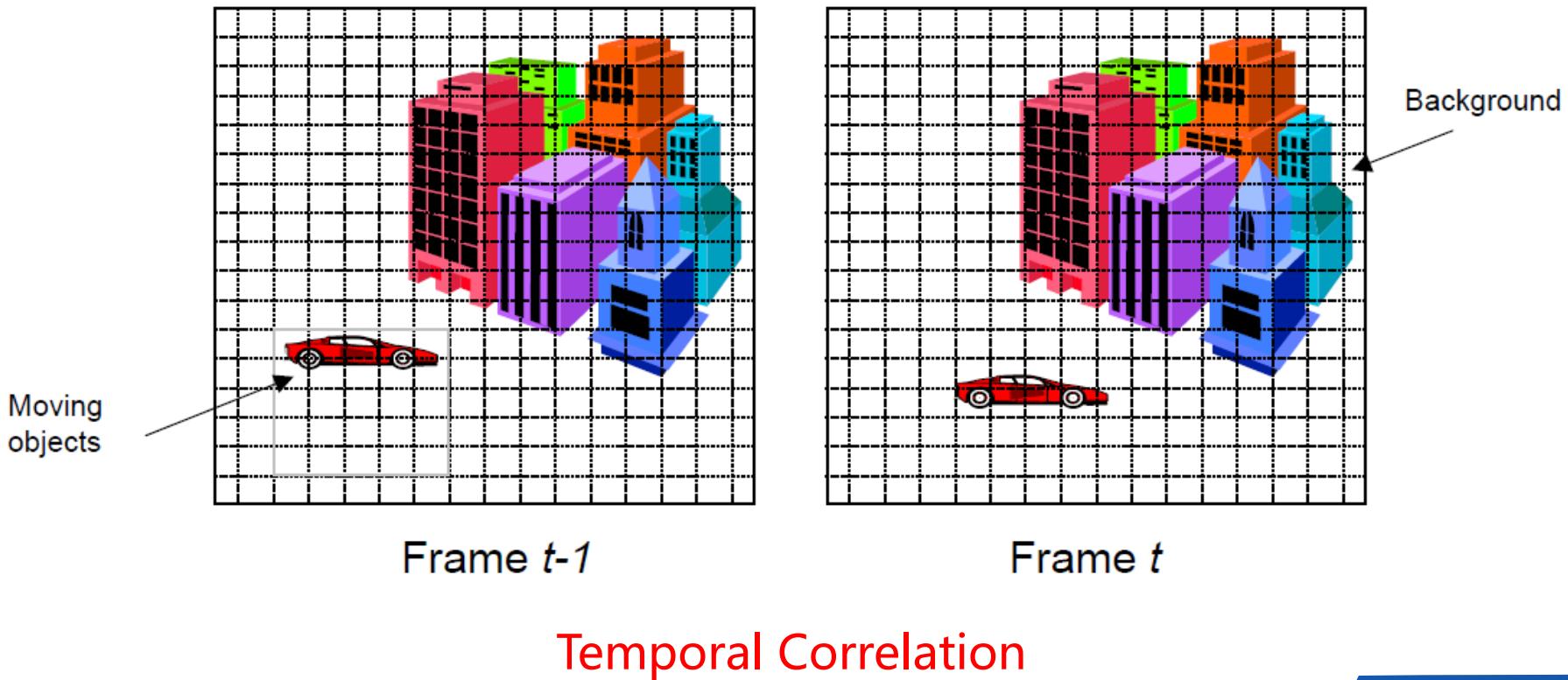
TIME  
367

The entire Super Mario Brothers game was only 40 kilobytes.

This image is 283 kilobytes.



- Compression is possible.
- For a video, adjacent frames are correlated.



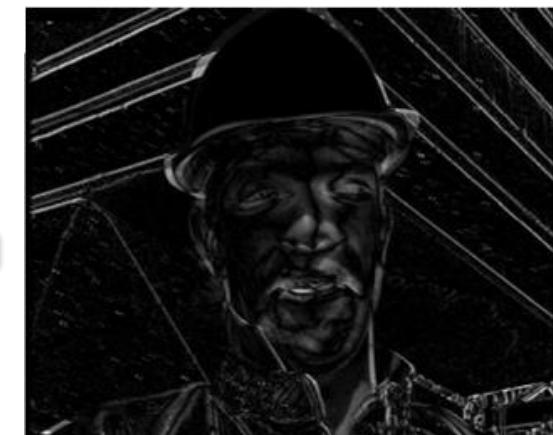
- Compression is possible.
- For a video, adjacent frames are correlated.



Frame t



Frame t-1



Differences of  
Frame t and Frame t-1

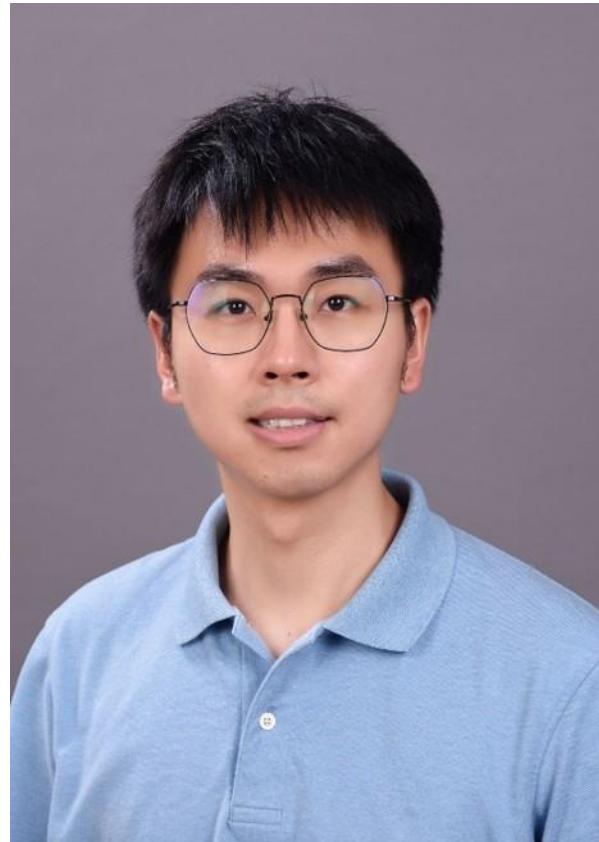
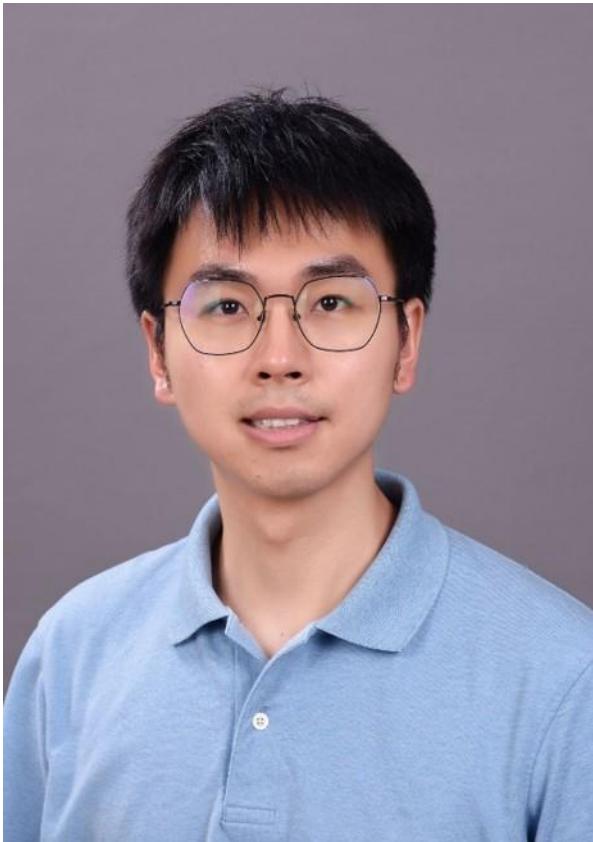


西安交通大学  
XI'AN JIAOTONG UNIVERSITY

# Thank you!

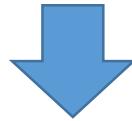


# »» 这两幅图有区别吗?



右边的图中有我给大家传递的秘密信息

# >> 那这两幅图呢？



R: 255 (0b1111 1111)

R: 254 (0b1111 1110)

G: 49 (0b0011 0001)

G: 48 (0b0011 0000)

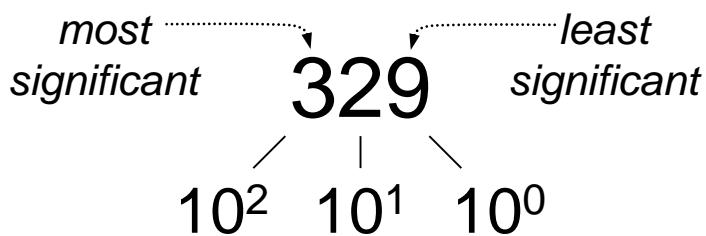
B: 49 (0b0011 0001)

B: 48 (0b0011 0000)

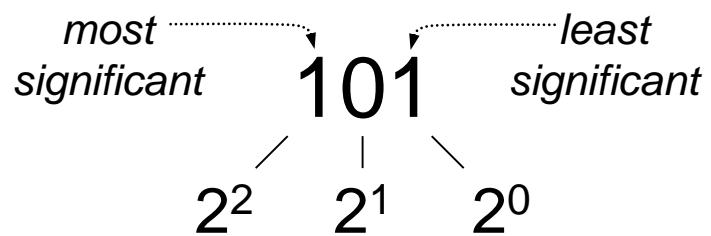
你得到了什么启示吗？

# >> 最低有效位 (least significant bit)

- 最低有效位是指一个二进制数字中的第0位（即最低位）。通常，最低有效位上的变化对整体数值影响最小。



$$3 \times 100 + 2 \times 10 + 9 \times 1 = 329$$

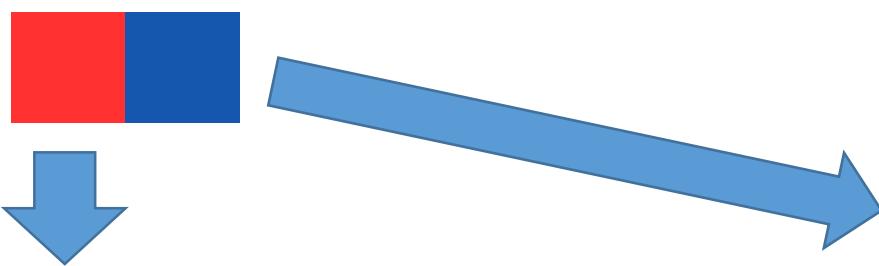


$$1 \times 4 + 0 \times 2 + 1 \times 1 = 5$$

- 例如，图像像素中最低有效位的变化肉眼很难察觉。

# >> 如何将信息隐藏于图像中?

- 1. 将所有像素点的R、G、B的最低有效位设为0
- 2. 将要隐藏的信息转化为二进制编码
- 3. 将二进制编码按顺序依次写入每个像素点R、G、B的最低有效位



R: 255 (0b1111 1111)

G: 48 (0b0011 0000)

B: 49 (0b0011 0001)

R: 21 (0b00010101)

G: 87 (0b01010111)

B: 174 (0b10101110)

嵌入的隐藏内容为：

101110

# »» 思考作业 (3.11中午12:00截止)

---

1. 解密我照片中给大家的密语
  
2. 编写一个程序，具备以下功能：
  - 输入：一张png格式图片+需要隐藏的信息  
输出：嵌入隐藏信息后的图片
  
  - 输入：嵌入隐藏信息后的图片  
输出：解密的隐藏信息

可使用读取png格式的库函数

# >> 作业

---

1. 请给出894、145的二进制表示，并写出步骤；请说出二进制100010111表示的是什么数？为什么？
2. 怎样理解“所有计算都可转化为逻辑运算实现”？例如，乘法运算“6\*5”如何用逻辑运算实现（搜索相关资料）？
3. 已知一个8字节的双精度浮点数十六进制表达为C0-5E-28-00-00-00-00-00，请问该数据代表的实数是多少？
4. 有一种说法“音频是对时间的采样，图片是对空间和时间的采样”你如何理解这种说法？
5. 请查阅资料叙述1种典型的格式，如bmp, mp3, jpg等，描述他们如何被编码程0, 1。隐藏在这些格式背后的往往是标准，这个标准是什么？这个标准有什么用？
6. 调研并回答计算机的指令如何编码的？什么是定长指令，什么是非定长指令集，各有什么好处？

## >> 扩展阅读

---

1. Latex教程（从第4次课开始，作业用Latex生成pdf提交）
2. 霍夫曼编码是如何实现压缩的