本講義の概要

- **情報通信の数理**
  - ⋆ **情報理論**
  - ⋆ **符号理論**
  - ⋆ **暗号理論**
- **それを支える数学**
  - ⋆ **有限体とその上の線型代数・Galois 理論**
  - ⋆ **計算量の理論**

情報通信を行なう際の要請

- **効率的に** $\longrightarrow$ **情報理論**
- **確実に** $\longrightarrow$ **符号理論**
- **安全に** $\longrightarrow$ **暗号理論**

情報通信を行なう際の要請

- **効率的に** $\longrightarrow$ **情報理論**
- **確実に** $\longrightarrow$ **符号理論**
- **安全に** $\longrightarrow$ **暗号理論**

情報理論 (符号化・情報量の理論)

- 伝えるべき情報をより効率良く伝えるには

- 「効率の良さ」を計る
  - ⋆ 伝えるべき「情報の量」を計る
  - ⋆ 伝える為の「手間」を計る

$\longrightarrow$ Shannon
「情報の量は伝えるのに必要な手間と一致」

ASCII code

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	(control characters)															
1	(control characters)															
2		!	"	#	$	%	&	’	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	‘	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	

7 bit で 1 文字を表す (実装上は 8 bit にする)

ASCII code

A	01000001
B	01000010
C	01000011
D	01000100
E	01000101
F	01000110
G	01000111
H	01001000
I	01001001

J	01001010
K	01001011
L	01001100
M	01001101
N	01001110
O	01001111
P	01010000
Q	01010001
R	01010010

S	01010011
T	01010100
U	01010101
V	01010110
W	01010111
X	01011000
Y	01011001
Z	01011010

モールス符号 (Morse code)

A	・ ー
B	ー ・・・
C	ー ・ ー ・
D	ー ・・
E	・
F	・・ ー ・
G	ー ー ・
H	・・・・
I	・・

J	・ ー ー ー
K	ー ・ ー
L	・ ー ・・
M	ー ー
N	ー ・
O	ー ー ー
P	・ ー ー ・
Q	ー ー ・ ー
R	・ ー ・

S	・・・
T	ー
U	・・ ー
V	・・・ ー
W	・ ー ー
X	ー ・・ ー
Y	ー ・ ー ー
Z	ー ー ・・

モールス符号 (Morse code)

1 **文字のための符号長が区々**

$\longleftarrow$ **頻度の高い文字は短く、低い文字は長く**

$\longrightarrow$ **頻度まで考慮して符号長の期待値を短く**

$\cdots$ **頻度 (出現確率) を考慮して**
符号効率の定式化を考える

モールス符号 (Morse code)

1 **文字のための符号長が区々**

$\longleftarrow$ **頻度の高い文字は短く、低い文字は長く**

$\longrightarrow$ **頻度まで考慮して符号長の期待値を短く**

$\cdots$ **頻度 (出現確率) を考慮して**
符号効率の定式化を考える

モールス符号 (Morse code)

1 **文字のための符号長が区々**

$\longleftarrow$ **頻度の高い文字は短く、低い文字は長く**

$\longrightarrow$ **頻度まで考慮して符号長の期待値を短く**

$\cdots$ **頻度 (出現確率) を考慮して**
符号効率の定式化を考える

モールス符号 (Morse code)

1 **文字のための符号長が区々**

$\longleftarrow$ **頻度の高い文字は短く、低い文字は長く**

$\longrightarrow$ **頻度まで考慮して符号長の期待値を短く**

$\cdots$ **頻度 (出現確率) を考慮して**
符号効率の定式化を考える

各符号語の長さが異なると問題も生ずる

$\longrightarrow$ 一意復号可能か？

$\longrightarrow$ 一意復号可能としても瞬時復号可能か？

各符号語の長さが異なると問題も生ずる

$\longrightarrow$ **一意復号可能か？**

$\longrightarrow$ **一意復号可能としても瞬時復号可能か？**

各符号語の長さが異なると問題も生ずる

$\longrightarrow$ **一意復号可能か？**

$\longrightarrow$ **一意復号可能としても瞬時復号可能か？**

例: $S = \{a, b, c\}, T = \{0, 1\}$

$$a \longmapsto 0$$

$$b \longmapsto 01$$

$$c \longmapsto 001$$

「001」が ab か c か判らない

$\longrightarrow$ 一意復号可能でない!!

例: $S = \{a, b, c\}, T = \{0, 1\}$

$$a \longmapsto 0$$

$$b \longmapsto 01$$

$$c \longmapsto 001$$

「001」が ab か c か判らない

$\longrightarrow$ **一意復号可能でない!!**

例: $S = \{a, b, c\}, T = \{0, 1\}$

$$a \longmapsto 0$$

$$b \longmapsto 01$$

$$c \longmapsto 001$$

「001」が ab か c か判らない

$\longrightarrow$ **一意復号可能でない!!**

例: $S = \{a, b, c\}, T = \{0, 1\}$

$$a \longmapsto 0$$

$$b \longmapsto 01$$

$$c \longmapsto 11$$

一意復号可能ではあるが、

「011$\cdots$」まで見ただけでは

ac$\cdots$ か bc$\cdots$ か判らない

(「0111」なら bc、「01111」なら acc)

$\longrightarrow$ 瞬時復号可能でない

例: $S = \{a, b, c\}, T = \{0, 1\}$

$$a \longmapsto 0$$
$$b \longmapsto 01$$
$$c \longmapsto 11$$

一意復号可能ではあるが、

「011$\cdots$」まで見ただけでは
ac$\cdots$ か bc$\cdots$ か判らない
(「0111」なら bc、「01111」なら acc)

$\longrightarrow$ **瞬時復号可能でない**

例: $S = \{a, b, c\}, T = \{0, 1\}$

$$a \longmapsto 0$$
$$b \longmapsto 01$$
$$c \longmapsto 11$$

一意復号可能ではあるが、

「011$\cdots$」まで見ただけでは
ac$\cdots$ か bc$\cdots$ か判らない
(「0111」なら bc、「01111」なら acc)

$\longrightarrow$ **瞬時復号可能でない**

例: $S = \{a, b, c\}, T = \{0, 1\}$

$$a \longmapsto 0$$

$$b \longmapsto 01$$

$$c \longmapsto 11$$

一意復号可能ではあるが、

「011$\cdots$」まで見ただけでは
ac$\cdots$ か bc$\cdots$ か判らない
(「0111」なら bc、「01111」なら acc)

$\longrightarrow$ **瞬時復号可能でない**