Digital Communication Systems
ECS 452

Asst. Prof. Dr. Prapun Suksompong

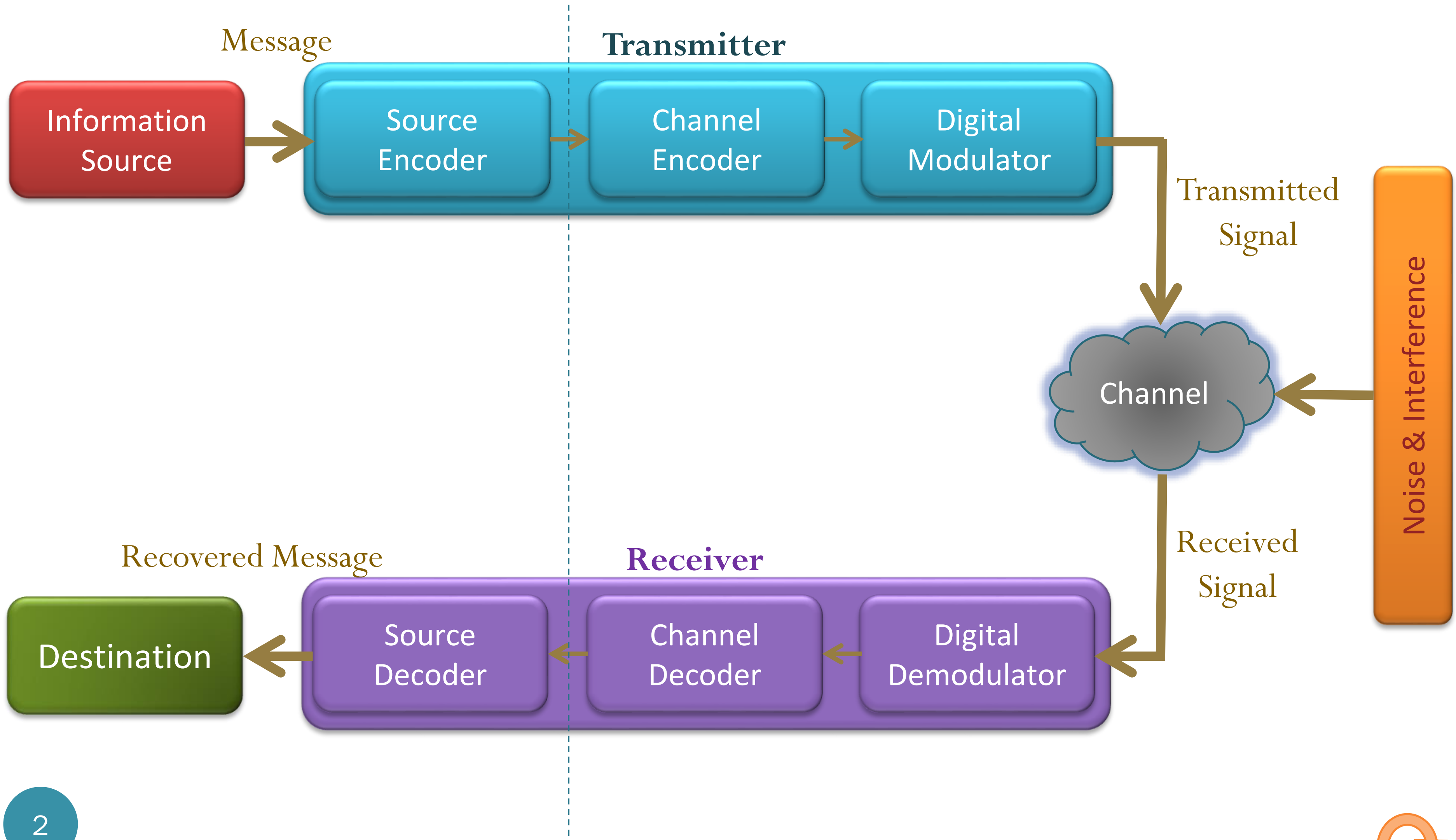
prapun@siit.tu.ac.th

Source Coding

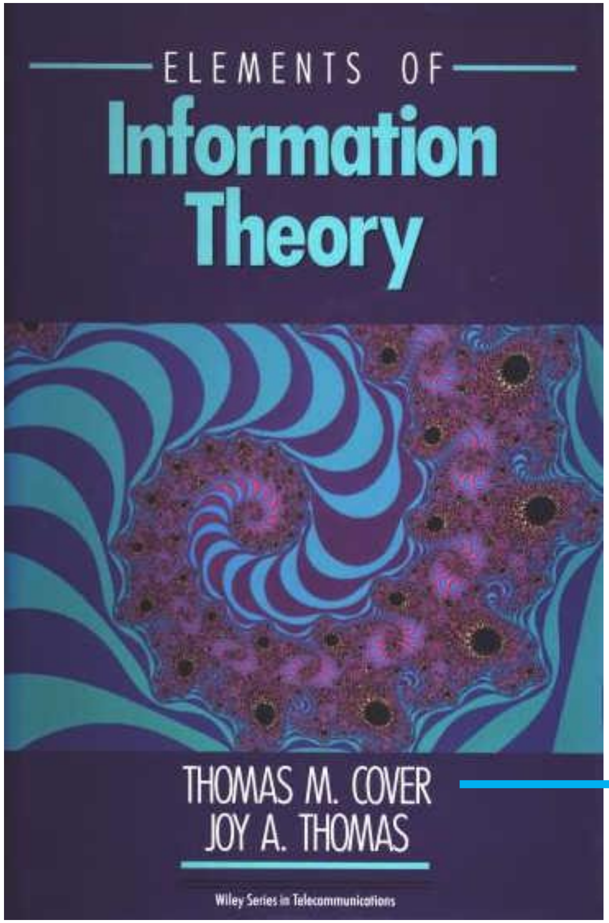
Office Hours:
BKD 3601-7

Monday	**14:00-16:00**
Wednesday	**14:40-16:00**

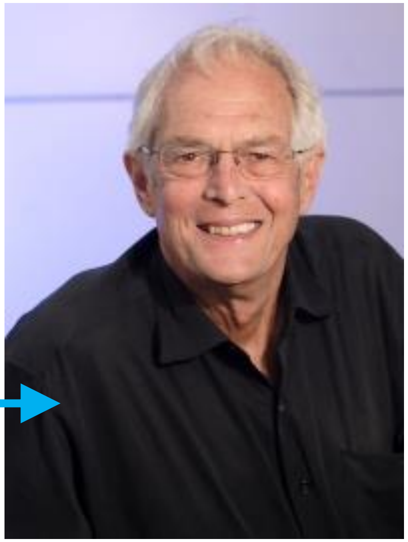
Elements of digital commu. sys.



Reference



- Elements of Information Theory
- 2006, 2nd Edition
- Chapters 2, 4 and 5



'the jewel in Stanford's crown'

One of the greatest information theorists since Claude Shannon (and the one most like Shannon in approach, clarity, and taste).

The ASCII Coded Character Set

					6	0	0	0	0	1	1	1	1
Bit					5	0	0	1	1	0	0	1	1
Number					4	0	1	0	1	0	1	0	1
				Hex	1st	0	1	2	3	4	5	6	7
3	**2**	**1**	**0**	**2nd**		0	16	32	48	64	80	96	112
0	0	0	0	0		NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1		SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2		STX	DC2	“	2	B	R	b	r
0	0	1	1	3		ETX	DC3	#	3	C	S	c	s
0	1	0	0	4		EOT	DC4	$	4	D	T	d	t
0	1	0	1	5		ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6		ACK	SYN	&	6	F	V	f	v
0	1	1	1	7		BEL	ETB	'	7	G	W	g	w
1	0	0	0	8		BS	CAN	(	8	H	X	h	x
1	0	0	1	9		HT	EM	)	9	I	Y	i	y
1	0	1	0	A		LF	SUB	*	:	J	Z	j	z
1	0	1	1	B		VT	ESC	+	;	K	[	k	{
1	1	0	0	C		FF	FS	,	<	L	\	l	|
1	1	0	1	D		CR	GS	-	=	M	]	m	}
1	1	1	0	E		SO	RS	.	>	N	^	n	~
1	1	1	1	F		SI	US	/	?	O	—	o	DEL

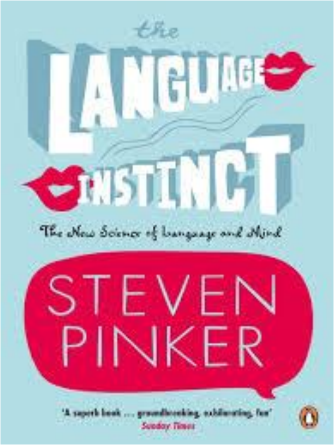
[The ARRL Handbook for Radio Communications 2013]

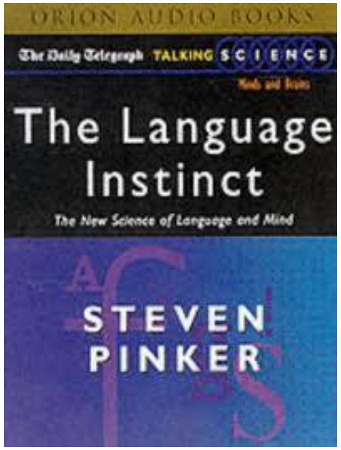
English Redundancy: Ex. 1

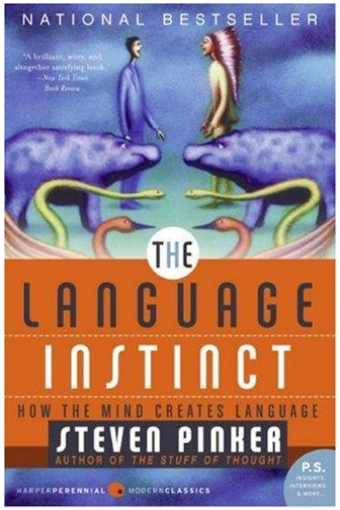
J-st tr- t- r--d th-s s-nt-nc-.

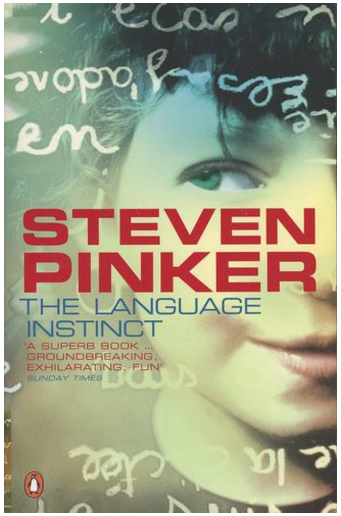
English Redundancy: Ex. 2

yxx cxn xndxrstxnd whxt x xm wrxtxng xvxn xf x rxplxcx xll thx vxwxls wxth xn 'x' (t gts lttl hrdr f y dn't vn kn whr th vwls r).









English Redundancy: Ex. 3

To be, or xxx xx xx,
xxxx xx xxx xxxxxxxxx

Ex. DMS (1)

$$\mathcal{S}_X = \{a,b,c,d,e\}$$

$$p_X(x) = \begin{cases} \frac{1}{5}, & x \in \{a,b,c,d,e\} \\ 0, & \text{otherwise} \end{cases}$$

Information Source $\rightarrow$

x_1 x_2 x_3

a	c	a	c	e	c	d	b	c	e
d	a	e	e	d	a	b	b	b	d
b	b	a	a	b	e	b	e	d	c
c	e	d	b	c	e	c	a	a	c
a	a	e	a	c	c	a	a	d	c
d	e	e	a	a	c	a	a	a	b
b	c	a	e	b	b	e	d	b	c
d	e	b	c	a	e	e	d	d	c
d	a	b	c	a	b	c	d	d	e
d	c	e	a	b	a	a	c	a	d

Approximately 20% are letter ‘a’s

[GenRV_Discrete_datasample_Ex1.m]

Ex. DMS (2)

$$\mathcal{S}_X = \{1,2,3,4\}$$

$$p_X(x) = \begin{cases} \frac{1}{2}, & x = 1, \\ \frac{1}{4}, & x = 2, \\ \frac{1}{8}, & x \in \{3,4\} \\ 0, & \text{otherwise} \end{cases}$$

Information Source

2	1	1	2	1	4	1	1	1	1
1	1	4	1	1	2	4	2	2	1
3	1	1	2	3	2	4	1	2	4
2	1	1	2	1	1	3	3	1	1
1	3	4	1	4	1	1	2	4	1
4	1	4	1	2	2	1	4	2	1
4	1	1	1	1	2	1	4	2	4
2	1	1	1	2	1	2	1	3	2
2	1	1	1	1	1	1	2	3	2
2	1	1	2	1	4	2	1	2	1

Approximately 50% are number ‘1’s

[GenRV_Discrete_datasample_Ex2.m]

Demo: DMS in MATLAB

```
clear all; close all;

S_X = [1 2 3 4]; p_X = [1/2 1/4 1/8 1/8]; n = 1e6;

SourceString = randsrc(1,n,[S_X;p_X]);
```

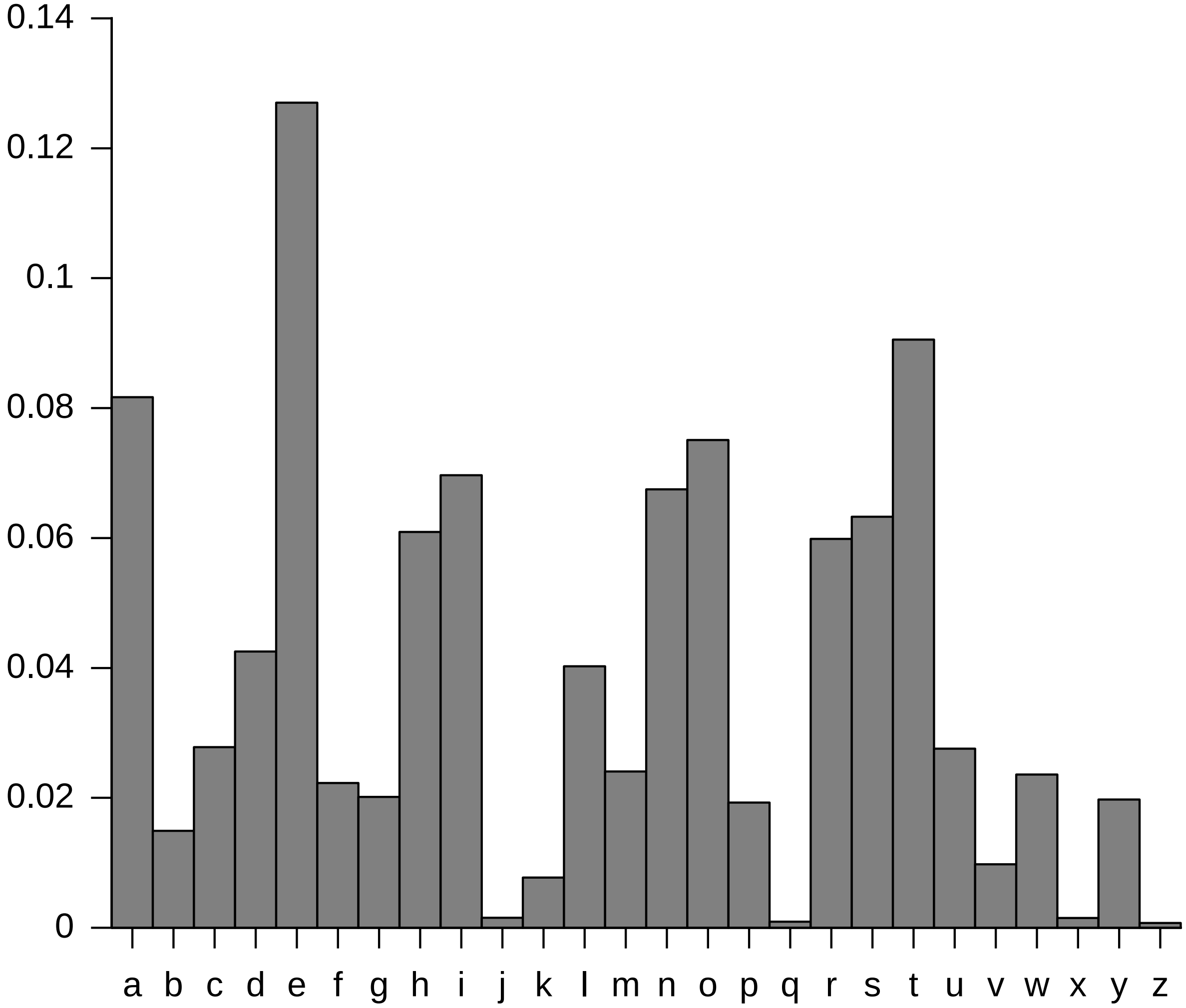
Alternatively, we can also use

```
SourceString = datasample(S_X,n,'Weights',p_X);
```

```
rf = hist(SourceString,S_X)/n; % Ref. Freq. calc.
stem(S_X,rf,'rx','LineWidth',2) % Plot Rel. Freq.
hold on
stem(S_X,p_X,'bo','LineWidth',2) % Plot pmf
xlim([min(S_X)-1,max(S_X)+1])
legend('Rel. freq. from sim.','pmf p_X(x)')
xlabel('x')
grid on
```

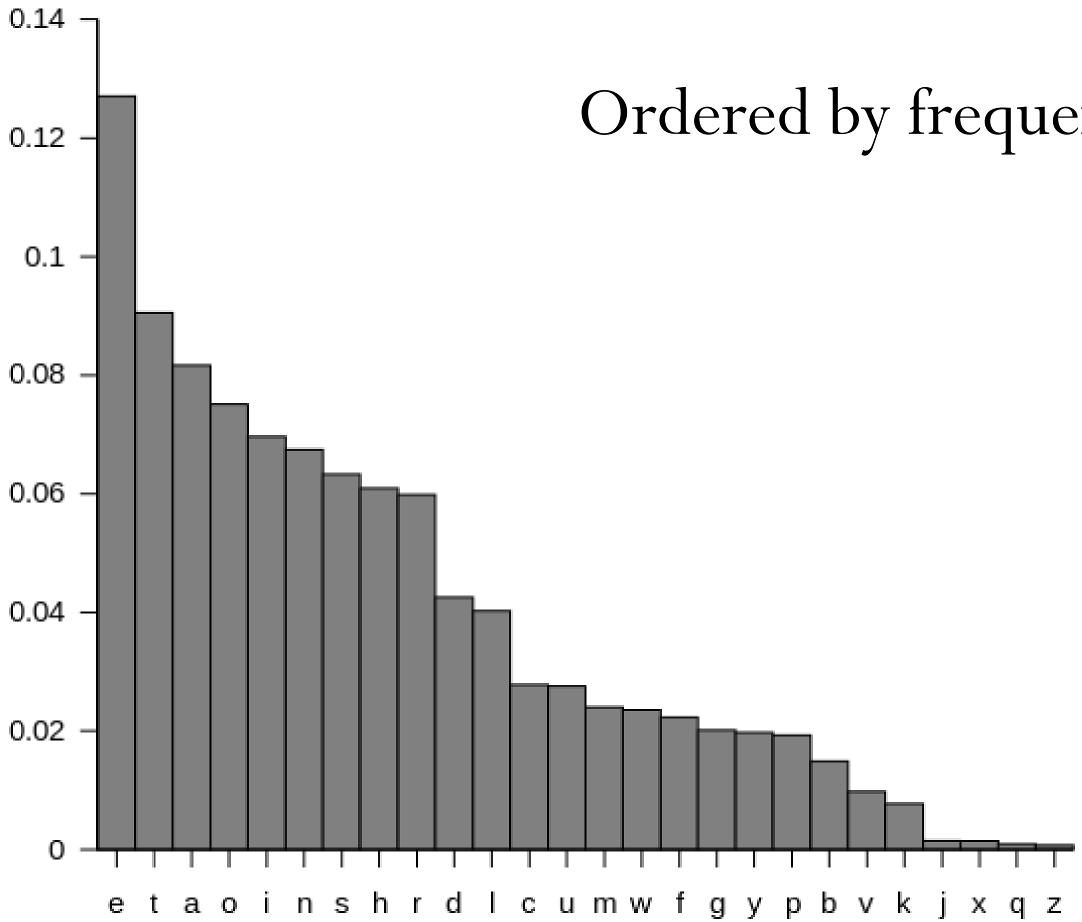
[GenRV_Discrete_datasample_Ex.m]

Relative freq. of letters in the English language



[http://en.wikipedia.org/wiki/Letter_frequency]

Relative freq. of letters in the English language



Ordered by frequency

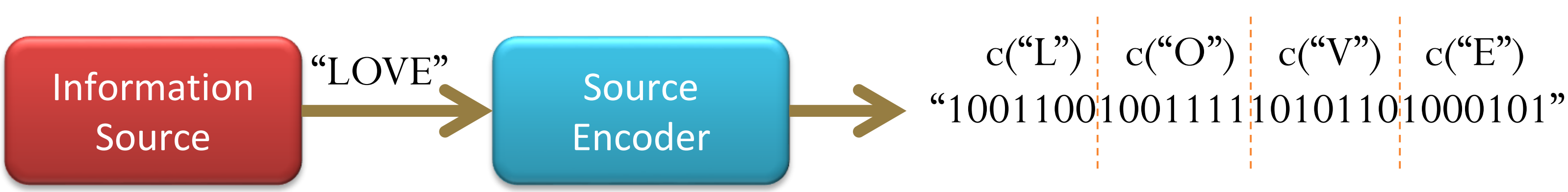
[http://en.wikipedia.org/wiki/Letter_frequency]

Example: ASCII Encoder

Character	Codeword
⋮	
E	1000101
⋮	
L	1001100
⋮	
O	1001111
⋮	
V	1010110
⋮	

MATLAB:

```
>> M = 'LOVE';
>> X = dec2bin(M,7);
>> X = reshape(X',1,numel(X))
X =
1001100100111110101101000101
```



Morse code

(wired and wireless)

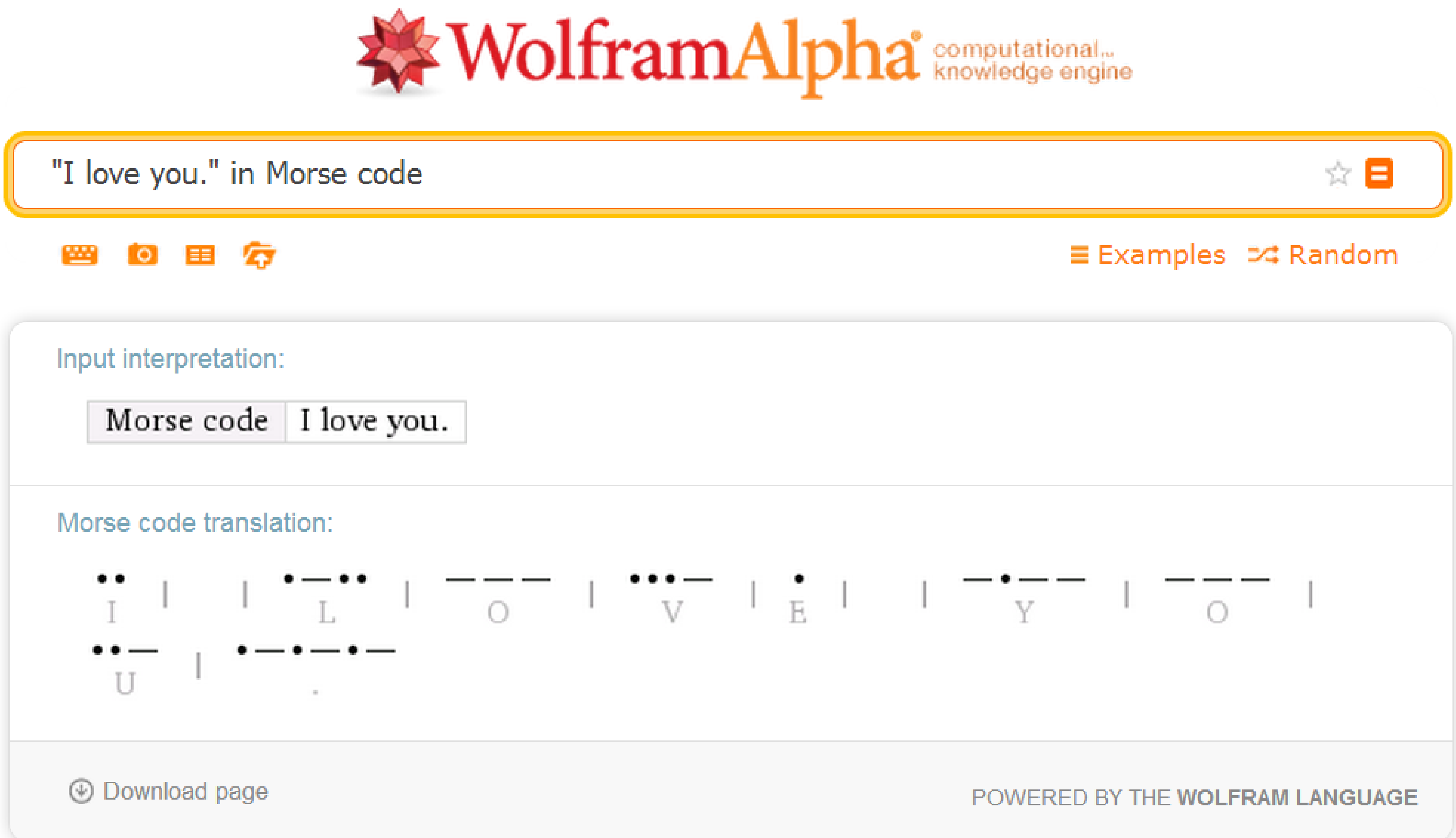
- **Telegraph network**
- Samuel **Morse**, 1838

- A sequence of on-off tones (or , lights, or clicks)

A	• ▬	U	• • ▬
B	▬ • • •	V	• • • ▬
C	▬ • ▬ •	W	• ▬ ▬
D	▬ • •	X	▬ • • ▬
E	•	Y	▬ • ▬ ▬
F	• • ▬ •	Z	▬ ▬ • •
G	▬ ▬ •		
H	• • • •		
I	• •		
J	• ▬ ▬ ▬		
K	▬ • ▬	1	• ▬ ▬ ▬ ▬
L	• ▬ • •	2	• • ▬ ▬ ▬
M	▬ ▬	3	• • • ▬ ▬
N	▬ •	4	• • • • ▬
O	▬ ▬ ▬	5	• • • • •
P	• ▬ ▬ •	6	▬ • • • •
Q	▬ ▬ • ▬	7	▬ ▬ • • •
R	• ▬ •	8	▬ ▬ ▬ • •
S	• • •	9	▬ ▬ ▬ ▬ •
T	▬	0	▬ ▬ ▬ ▬ ▬

SAMUEL MORSE DICTATES A LETTER TO HIS SECRETARY.

Example



[http://www.wolframalpha.com/input/?i=%22I+love+you.%22+in+Morse+code]

Morse code: Key Idea

Frequently-used characters (e,t) are mapped to short codewords.

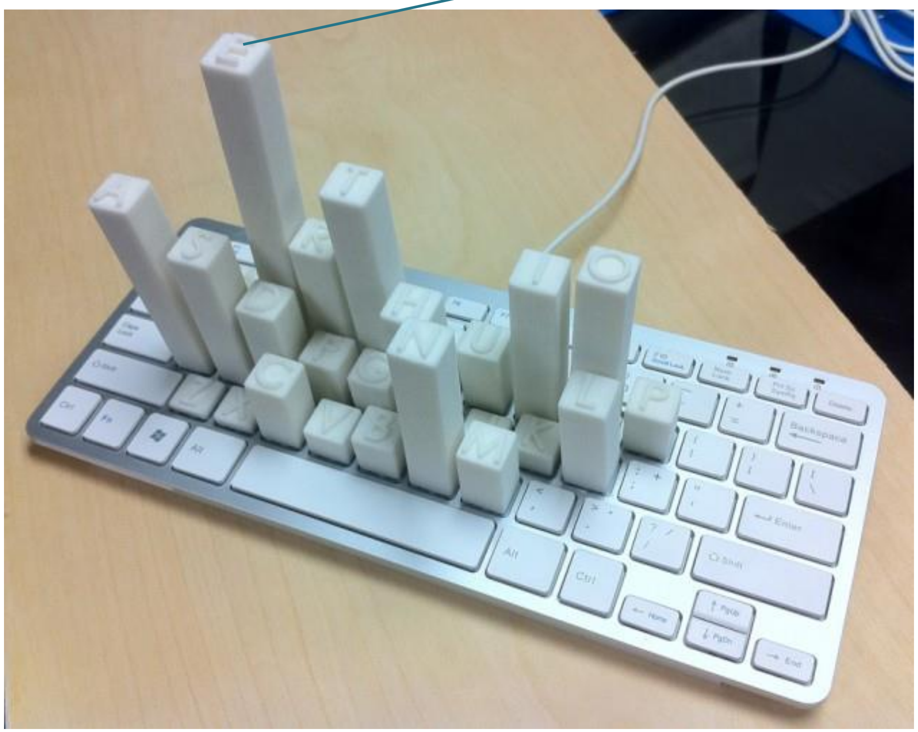
A	● ▬	U	● ● ▬
B	▬ ● ● ●	V	● ● ● ▬
C	▬ ● ▬ ●	W	● ▬ ▬
D	▬ ● ●	X	▬ ● ● ▬
E	●	Y	▬ ● ▬ ▬
F	● ● ▬ ●	Z	▬ ▬ ● ●
G	▬ ▬ ●		
H	● ● ● ●		
I	● ●		
J	● ▬ ▬ ▬		
K	▬ ● ▬	1	● ▬ ▬ ▬ ▬
L	● ▬ ● ●	2	● ● ▬ ▬ ▬
M	▬ ▬	3	● ● ● ▬ ▬
N	▬ ●	4	● ● ● ● ▬
O	▬ ▬ ▬	5	● ● ● ● ●
P	● ▬ ▬ ●	6	▬ ● ● ● ●
Q	▬ ▬ ● ▬	7	▬ ▬ ● ● ●
R	● ▬ ●	8	▬ ▬ ▬ ● ●
S	● ● ●	9	▬ ▬ ▬ ▬ ●
T	▬	0	▬ ▬ ▬ ▬ ▬

Basic form of compression.

Morse code: Key Idea

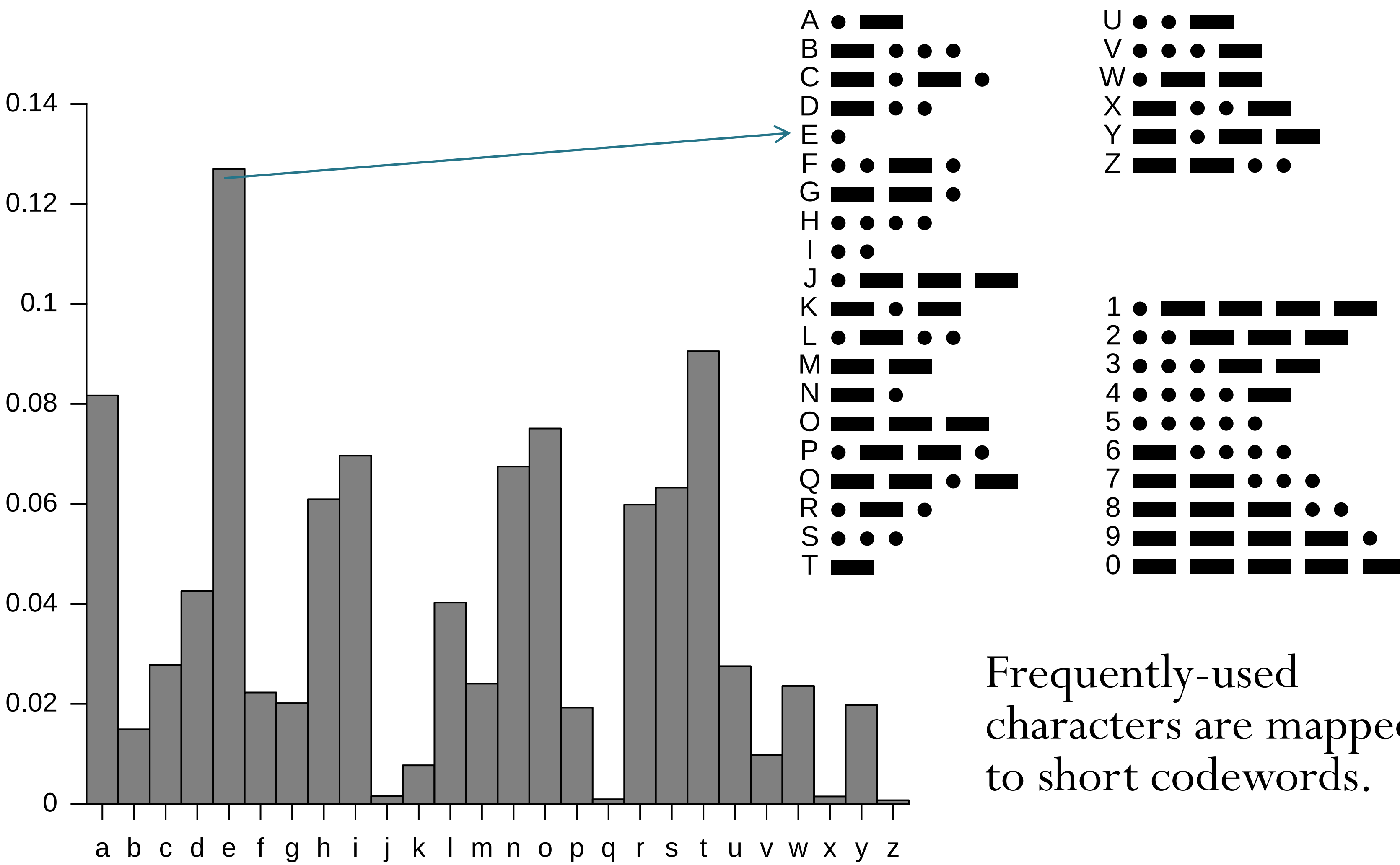
Frequently-used characters are mapped to short codewords.

A	● ▬
B	▬ ● ● ●
C	▬ ● ▬ ●
D	▬ ● ●
E	●
F	● ● ▬ ●
G	▬ ▬ ●
H	● ● ● ●
I	● ●
J	● ▬ ▬ ▬
K	▬ ● ▬
L	● ▬ ● ●
M	▬ ▬
N	▬ ●
O	▬ ▬ ▬
P	● ▬ ▬ ●
Q	▬ ▬ ● ▬
R	● ▬ ●
S	● ● ●
T	▬
U	● ● ▬
V	● ● ● ▬
W	● ▬ ▬
X	▬ ● ● ▬
Y	▬ ● ▬ ▬
Z	▬ ▬ ● ●
1	● ▬ ▬ ▬ ▬
2	● ● ▬ ▬ ▬
3	● ● ● ▬ ▬
4	● ● ● ● ▬
5	● ● ● ● ●
6	▬ ● ● ● ●
7	▬ ▬ ● ● ●
8	▬ ▬ ▬ ● ●
9	▬ ▬ ▬ ▬ ●
0	▬ ▬ ▬ ▬ ▬



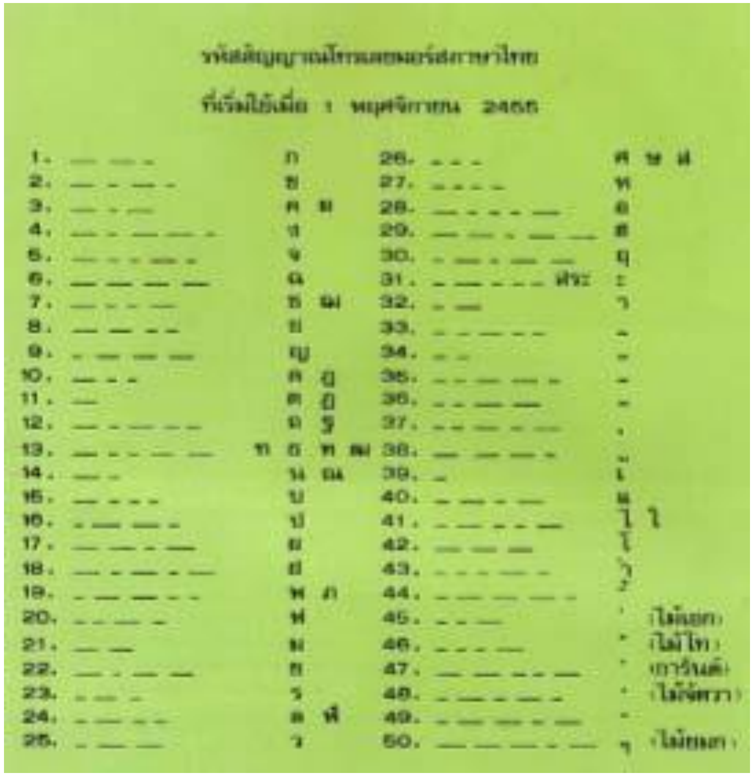
Relative frequencies of letters in the English language

Morse code: Key Idea



Frequently-used characters are mapped to short codewords.

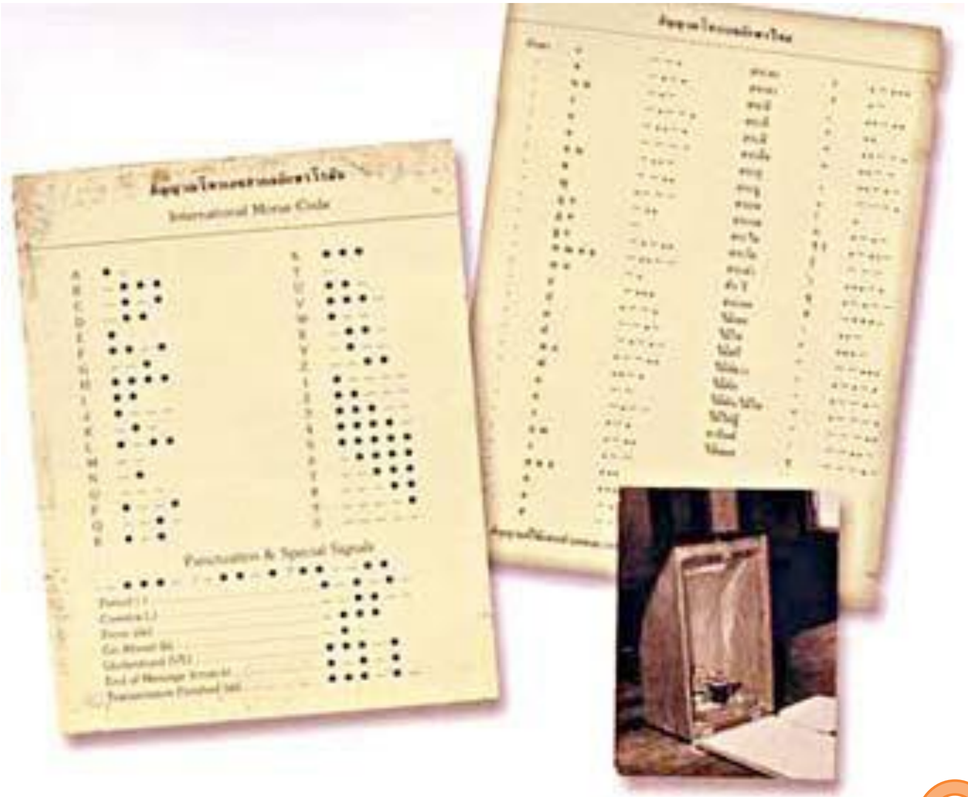
รหัสมอร์สภาษาไทย



รหัสสัญญาณโทรเลขมอร์สภาษาไทย

ที่เริ่มใช้เมื่อ 1 พฤศจิกายน 2455

1.	- - .	ก	26.	. . .	ศ ษ ส
2.	- . - .	ข	27.	. . . .	ห
3.	- . -	ค ฆ	28.	- . . . -	อ
4.	- . - - .	ง	29.	- - . - -	ฮ
5.	- . . - .	จ	30.	. - . - -	ฤ
6.	- - - -	ฉ	31.	. - . . . สระ	ะ
7.	- . . -	ช ฌ	32.	. -	า
8.	- - . .	ซ	33.	. . - . .	ิ
9.	. - - -	ญ	34.	. .	ี
10.	- . .	ด ฎ	35.	. . - - .	ึ
11.	-	ต ฏ	36.	. . - -	ื
12.	- . - . .	ถ ฐ	37.	. . - . -	ุ
13.	- . . - -	ฑ ธ ท ฒ	38.	- - - .	ู
14.	- .	น ณ	39.	.	เ
15.	- . . .	บ	40.	. - . -	แ
16.	. - - .	ป	41.	. - . . -	ไ ใ
17.	- - . -	ผ	42.	- - -	โ
18.	- . - . -	ฝ	43.	. . . - .	ำ
19.	. - - . .	พ ภ	44.	. - - - .	ั
20.	. . - .	ฟ	45.	. . -	่ (ไม้เอก)
21.	- -	ม	46.	. . . -	้ (ไม้โท)
22.	- . - -	ย	47.	- - . . -	์ (การันต์)
23.	. - .	ร	48.	. - . - .	๋ (ไม้จัตวา)
24.	. - . .	ล ฬ	49.	. - - . -	็
25.	. - -	ว	50.	- - - . -	ๆ (ไม้ยมก)

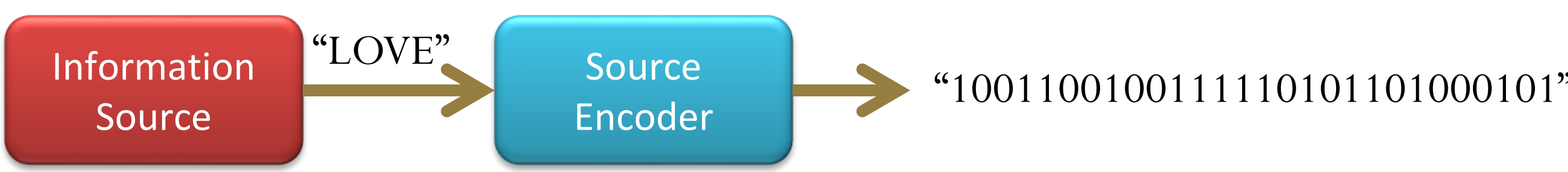


Example: ASCII Encoder

Character	Codeword
⋮	
E	1000101
⋮	
L	1001100
⋮	
O	1001111
⋮	
V	1010110
⋮	

MATLAB:

```
>> M = 'LOVE';
>> X = dec2bin(M,7);
>> X = reshape(X',1,numel(X))
X =
1001100100111110101101000101
```

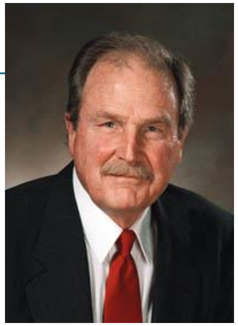


Prof. Robert Mario Fano (MIT)
Shannon Award (1976)

Shannon–Fano coding

- Proposed in Shannon's "A Mathematical Theory of Communication" in 1948
- The method was attributed to Fano, who later published it as a technical report.
- Should not be confused with
  - Shannon coding, the coding method used to prove Shannon's noiseless coding theorem, or with
  - Shannon–Fano–Elias coding (also known as Elias coding), the precursor to arithmetic coding.

Huffman Code


David Huffman (1925–1999)

- MIT, 1951
- Information theory class taught by Professor Fano.
- Huffman and his classmates were given the choice of
  - a term paper on the problem of finding the most efficient binary code.

  or
  - a final exam.
- Huffman, unable to prove any codes were the most efficient, was about to give up and start studying for the final when he hit upon the idea of using a frequency-sorted binary tree and quickly proved this method the most efficient.
- Huffman avoided the major flaw of the suboptimal Shannon-Fano coding by building the tree from the bottom up instead of from the top down.

Ex. Huffman Coding in MATLAB

Observe that MATLAB automatically give the **expected length** of the codewords

```
pX = [0.5 0.25 0.125 0.125];          % pmf of X
SX = [1:length(pX)];                  % Source Alphabet
[dict,EL] = huffmandict(SX,pX);       % Create codebook

%% Pretty print the codebook.
codebook = dict;
for i = 1:length(codebook)
    codebook{i,2} = num2str(codebook{i,2});
end
codebook

%% Try to encode some random source string
n = 5; % Number of source symbols to be generated
sourceString = randsrc(1,10,[SX; pX]) % Create data using pX
encodedString = huffmanenco(sourceString,dict) % Encode the data
```

[Huffman_Demo_Ex1]

Ex. Huffman Coding in MATLAB

```
codebook =

    [1]    '0'
    [2]    '1  0'
    [3]    '1  1  1'
    [4]    '1  1  0'


sourceString =

     1     4     4     1     3     1     1     4     3     4


encodedString =

     0     1     1     0     1     1     0     0     1     1     1     0     0     1     1     0     1     1     1     1     1     0
```


[Huffman_Demo_Ex1]

Ex. Huffman Coding in MATLAB

```
pX = [0.4 0.3 0.1 0.1 0.06 0.04]; % pmf of X
SX = [1:length(pX)];              % Source Alphabet
[dict,EL] = huffmandict(SX,pX);   % Create codebook

%% Pretty print the codebook.
codebook = dict;
for i = 1:length(codebook)
    codebook{i,2} = num2str(codebook{i,2});
end
codebook

EL
```

```
>> Huffman_Demo_Ex2

codebook =

    [1]    '1'
    [2]    '0  1'
    [3]    '0  0  0  0'
    [4]    '0  0  1'
    [5]    '0  0  0  1  0'
    [6]    '0  0  0  1  1'


EL =

   2.2000
```

The codewords can be different from our answers found earlier.
The expected length is the same.

[Huffman_Demo_Ex2]

Huffman Coding: Source Extension

$$X_k \overset{\text{i.i.d.}}{\sim} \text{Bernoulli}(p)$$

$$p = 0.1$$