

Licenciatura em Engenharia Informática

# Sistemas Multimédia

## Codificação de Informação e Entropia

Telmo Reis Cunha

Departamento de Eletrónica, Telecomunicações e Informática

Universidade de Aveiro – 2018/2019

# 1. Codificação de Informação

---

- A parte relevante de sinais, imagens, textos, etc. é a **Informação** que estes contêm.
- Essa informação deve ser representada num determinado suporte que lhe permita ser armazenada, transmitida, processada, sem que a referida informação se perca.
- Ao processo de representação da informação por **códigos** denomina-se **Codificação**.
- Por exemplo, o guião de “Sistemas Multimédia”:
  - guardado num ficheiro de texto, ocupa 3259 bytes;
  - comprimido para um ficheiro .ZIP ocupa 1676 bytes.
- A mesma informação foi armazenada com codificações distintas.

## 2. Codificação Binária

---

- Hoje em dia, a informação é processada, armazenada e transmitida usando **codificação binária**.
- Esta codificação admite que o elemento básico de codificação (o **bit**) pode obter dois valores possíveis: 0 ou 1.
- Códigos mais complexos são obtidos agrupando um conjunto de bits.
- Uma codificação de  $N$  bits gera, assim, um conjunto de  $2^N$  códigos possíveis.
- Por exemplo, uma codificação de 8 bits (i.e., **1 byte**) dá origem a 256 códigos.

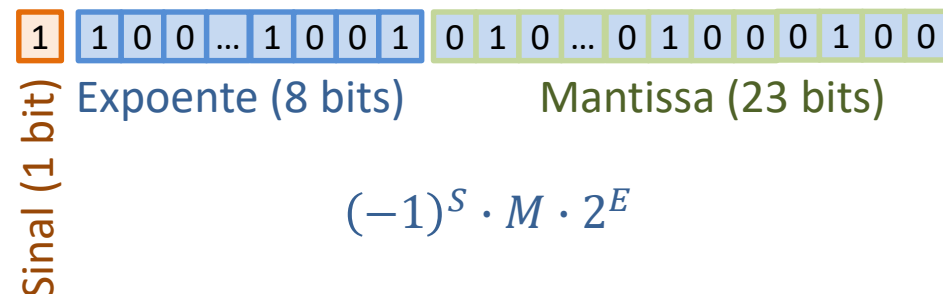
## 2. Codificação Binária

Binário								Decimal
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	1	0	2
0	0	0	0	0	0	1	1	3
⋮								⋮
1	1	1	1	1	1	0	0	252
1	1	1	1	1	1	0	1	253
1	1	1	1	1	1	1	0	254
1	1	1	1	1	1	1	1	255

Com 2 bytes (1 **word**) podem-se representar 65536 códigos distintos.

Com 2 words (1 **double word**, ou **dword**) são 4294967296 códigos.

Números reais também podem ser representados por códigos. Exemplo (IEEE 754 *single precision*):



$$(-1)^S \cdot M \cdot 2^E$$

Exemplo:

$$10010101_2 = 1 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 149_{10}$$

Base 2 Base 10

## 2. Codificação Binária

- Cada código pode ser associado a um **símbolo** do **alfabeto** que compõe as mensagens que são possíveis de realizar.

- Por exemplo, o código ASCII representa os caracteres num código de (originalmente) 7 bits.

ASCII – American Standard Code for Information Interchange (1963).

- Com a introdução de mais caracteres, o código ASCII usa atualmente 8 bits.

Dec	Char	Dec	Char	Dec	Char	Dec	Char
0	NUL (null)	32	SPACE	64	@	96	`
1	SOH (start of heading)	33	!	65	A	97	a
2	STX (start of text)	34	"	66	B	98	b
3	ETX (end of text)	35	#	67	C	99	c
4	EOT (end of transmission)	36	\$	68	D	100	d
5	ENQ (enquiry)	37	%	69	E	101	e
6	ACK (acknowledge)	38	&	70	F	102	f
7	BEL (bell)	39	'	71	G	103	g
8	BS (backspace)	40	(	72	H	104	h
9	TAB (horizontal tab)	41	)	73	I	105	i
10	LF (NL line feed, new line)	42	*	74	J	106	j
11	VT (vertical tab)	43	+	75	K	107	k
12	FF (NP form feed, new page)	44	,	76	L	108	l
13	CR (carriage return)	45	-	77	M	109	m
14	SO (shift out)	46	.	78	N	110	n
15	SI (shift in)	47	/	79	O	111	o
16	DLE (data link escape)	48	0	80	P	112	p
17	DC1 (device control 1)	49	1	81	Q	113	q
18	DC2 (device control 2)	50	2	82	R	114	r
19	DC3 (device control 3)	51	3	83	S	115	s
20	DC4 (device control 4)	52	4	84	T	116	t
21	NAK (negative acknowledge)	53	5	85	U	117	u
22	SYN (synchronous idle)	54	6	86	V	118	v
23	ETB (end of trans. block)	55	7	87	W	119	w
24	CAN (cancel)	56	8	88	X	120	x
25	EM (end of medium)	57	9	89	Y	121	y
26	SUB (substitute)	58	:	90	Z	122	z
27	ESC (escape)	59	;	91	[	123	{
28	FS (file separator)	60	<	92	\	124	
29	GS (group separator)	61	=	93	]	125	}
30	RS (record separator)	62	>	94	^	126	~
31	US (unit separator)	63	?	95	_	127	DEL

## 2. Codificação Binária

- Por exemplo, mensagem “Sistemas Multimédia” requer 19 bytes de informação (para armazenamento ou transmissão), se se usar o código ASCII.
- A questão é se esta será a forma mais eficiente de representar esta informação (já foi visto que não é).
- Torna-se necessário, então, analisar a **Eficiência da Codificação**.

0	1	0	1	0	0	1	1	S							
0	1	1	0	1	0	0	1	i							
0	1	1	1	0	0	1	1	s							
0	1	1	1	0	1	0	0	t							
1	0	0	0	0	0	1	0	é							
0	1	1	0	0	1	0	0	d							
0	1	1	0	1	0	0	0	i							
0	1	1	0	0	0	0	1	a							

### 3. Codificação Probabilística

---

- Nos codificadores binários vistos anteriormente, o número de bits usado para cada símbolo é constante.
- Mas pode ser considerada uma codificação que não impõe essa condição.
- Por exemplo:

Símbolo	Código
A	0
B	00
C	1

Este código é **ambíguo**.

A sequência 001 tanto pode representar BC como AAC.

Símbolo	Código
A	0
B	01
C	001

Este código é **não instantâneo**.

Sempre que surge um novo bit 0, é necessário esperar pelos bits seguintes (à partida, não se sabe quantos) para se identificar a chegada de um novo símbolo.

### 3. Codificação Probabilística

---

- Nos codificadores binários vistos anteriormente, o número de bits usado para cada símbolo é constante.
- Mas pode ser considerada uma codificação que não impõe essa condição.
- Por exemplo:

Símbolo	Código
A	00
B	01
C	1

Este código é **não ambíguo** e **instantâneo**.

Sempre que surge um 0, já se sabe que é necessário aguardar mais um bit para se identificar o símbolo.



### 3. Codificação Probabilística

---

- Como, num conjunto grande de informação (i.e., num conjunto grande de mensagens), alguns símbolos são mais frequentes que outros, então pode-se atribuir um menor número de bits aos símbolos mais frequentes.
- Este conceito deu origem à **Codificação Probabilística**, que entra em conta com a probabilidade de ocorrência de cada símbolo para gerar codificações eficientes.
- Como iremos ver, a probabilidade de ocorrência de um símbolo é uma característica fundamental no conceito de Informação.
- Por exemplo: a informação meteorológica de “chuva” num aeroporto situado num deserto é muito mais informática do que a informação “sol”.

### 3. Codificação Probabilística

---

- Considere-se a seguinte mensagem:

**AABCABABAAABBCABCAAC**

- Esta mensagem é composta por uma sequência de 20 símbolos.
- O alfabeto associado contém 3 símbolos: {A,B,C}
- Nesta mensagem, a frequência de cada símbolo é:

Símbolo	Número de ocorrências	Frequência
A	10	0.5
B	6	0.3
C	4	0.2

### 3. Codificação Probabilística

---

- Considere-se a seguinte mensagem:

AABCABABAAABBCABCAAC

- Considere-se, então, a seguinte codificação:

Símbolo	Código
A	0
B	10
C	11

Código não ambíguo e instantâneo.

Número médio de **bits por símbolo**:

$$\frac{5}{3} \approx 1.67$$

- Então, a mensagem pode ser codificada em 30 bits:

001011010010000101011010110011

- A codificação que consideraria um número igual de bits para cada símbolo requeria 40 bits para representar esta mensagem.

### 3. Codificação Probabilística

- O código Morse é um exemplo onde códigos mais frequentes são representados com códigos mais curtos – o objetivo essencial era aumentar a **eficiência** na transmissão/receção de mensagens.
- A frequência dos símbolos é a que surge em textos de língua inglesa.

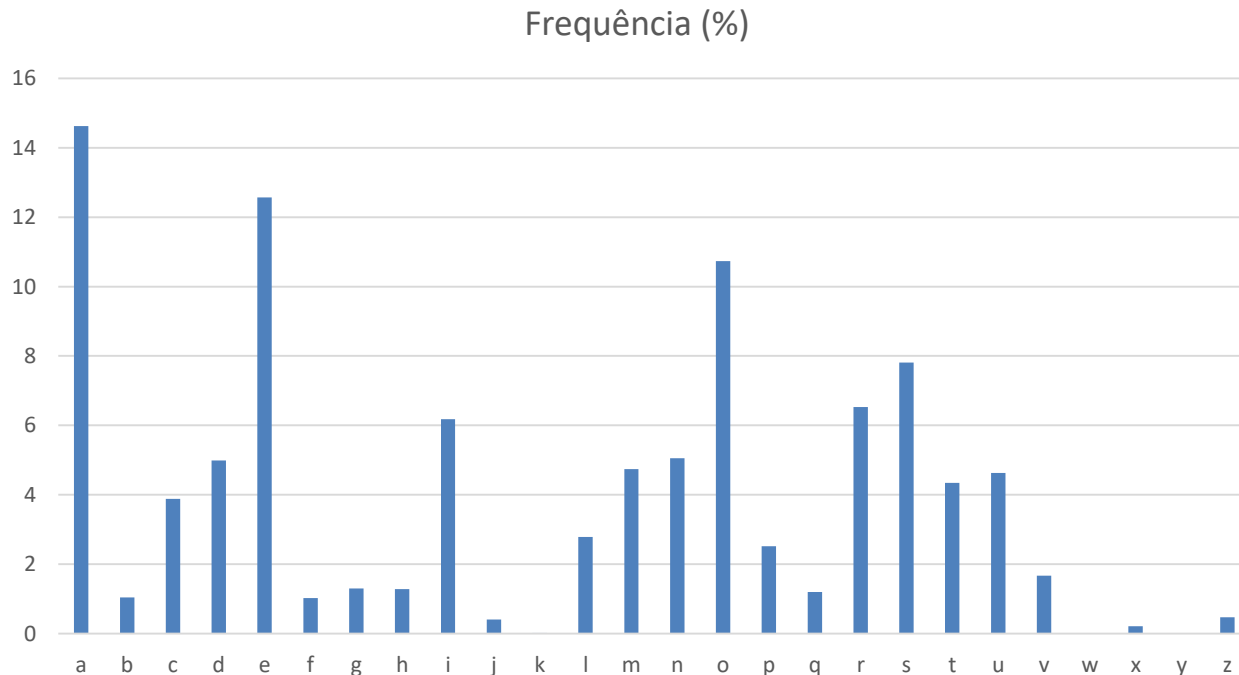
A • ■  
B ■ • •  
C ■ • ■  
D ■ •  
E •  
F • • ■  
G ■ ■  
H • • •  
I • •  
J • ■ ■ ■  
K ■ • ■  
L • ■ •  
M ■ ■  
N ■ •  
O ■ ■ ■  
P • ■ ■ •  
Q ■ ■ • ■  
R • ■ •  
S • • •  
T ■

U • • ■  
V • • ■  
W • ■ ■  
X ■ • • ■  
Y ■ • ■ ■  
Z ■ ■ • •

1 • ■ ■ ■ ■  
2 • • ■ ■ ■  
3 • • • ■ ■  
4 • • • ■  
5 • • • •  
6 ■ • • •  
7 ■ ■ • •  
8 ■ ■ ■ •  
9 ■ ■ ■ ■  
0 ■ ■ ■ ■ ■

### 3. Codificação Probabilística

- Naturalmente, na escrita de textos e mensagens em Português, alguns caracteres têm maior probabilidade de ocorrência do que outros.
- Essa **probabilidade** associada a cada símbolo (carater) é fundamental para a obtenção de **codificações eficientes**.



## 4. Quantidade de Informação e Entropia

---

- Analisemos, então, uma forma de medir a quantidade de informação.
- Seja um símbolo,  $s_k$ , de um determinado alfabeto.
- Sabe-se que esse símbolo tem a probabilidade  $p_k$  de ocorrer em mensagens geradas por esse alfabeto.
- Que **Quantidade de Informação** está associada à ocorrência (**evento**) desse símbolo numa mensagem?
- Por exemplo, num alfabeto de 256 símbolos, todos com igual probabilidade de ocorrência, é necessário usar um byte para conseguir distinguir símbolos distintos.
- Neste caso, a quantidade de informação associada à ocorrência de um símbolo é igual a 1 byte.

## 4. Quantidade de Informação e Entropia

---

- Define-se, então, a seguinte **métrica para a quantidade de informação associada à ocorrência de um símbolo**  $s_k$ :

$$I(s_k) = \log_2 \left( \frac{1}{p_k} \right) = -\log_2(p_k)$$

- Sabe-se que esse símbolo tem a probabilidade  $p_k$  de ocorrer em mensagens geradas por esse alfabeto.
- No exemplo anterior, onde  $p_k = 1/256$ , obtém-se:

$$I(s_k) = \log_2(2^8) = 8 \text{ bits (1 byte)}$$

## 4. Quantidade de Informação e Entropia

- E qual será, então, a quantidade de informação média associada à ocorrência de uma mensagem, gerada por um alfabeto de  $N$  símbolos?
- Essa quantidade de informação média designa-se por **Entropia**, e define-se pela média da quantidade de informação associada aos símbolos do alfabeto:

$$H(M) = \sum_{k=1}^N p_k \log_2 \left( \frac{1}{p_k} \right) = - \sum_{k=1}^N p_k \log_2(p_k)$$

- A Entropia tem como unidades:

bits por símbolo (bps)



## 4. Quantidade de Informação e Entropia

- Revisitando o exemplo visto num slide anterior:

AABCABABAAABBCABCAAC

Símbolo	Número de ocorrências	Frequência	Código
A	10	0.5	0
B	6	0.3	10
C	4	0.2	11

- Entropia:

$$H(M) = 0.5 \log_2 \left( \frac{1}{0.5} \right) + 0.3 \log_2 \left( \frac{1}{0.3} \right) + 0.2 \log_2 \left( \frac{1}{0.2} \right) \approx 1.49 \text{ bps}$$

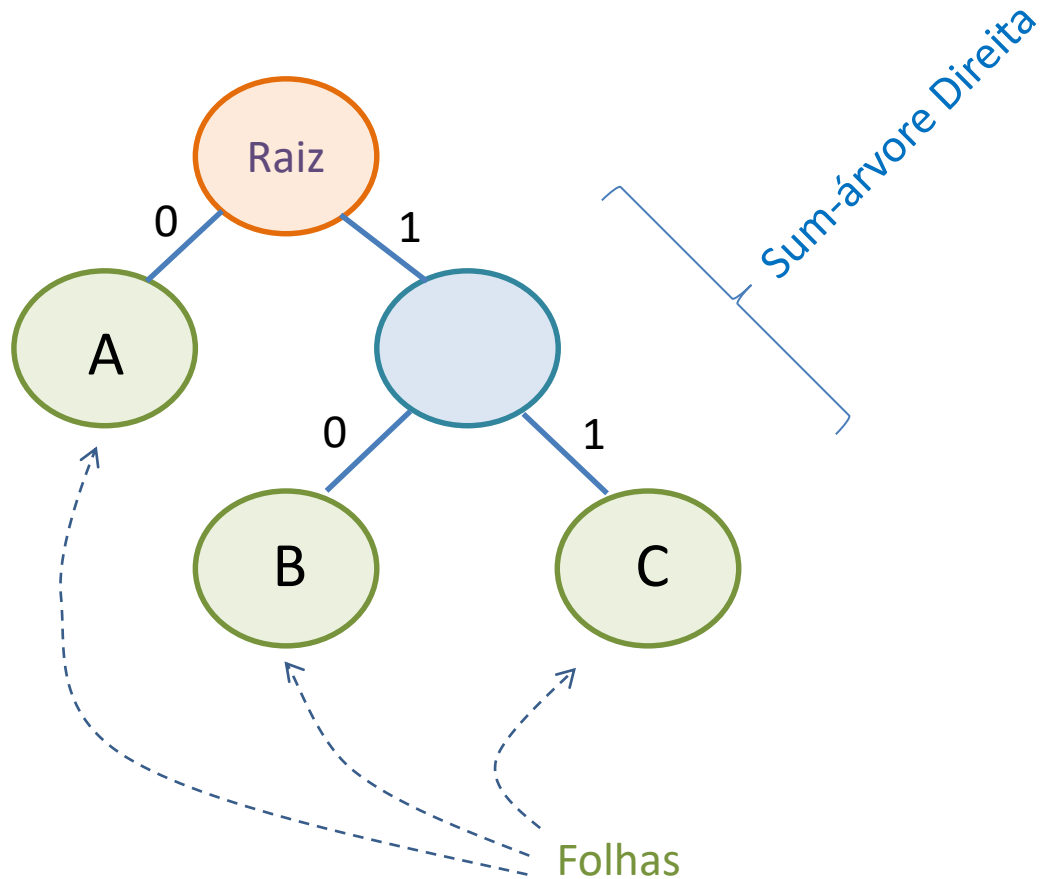
- Note-se que, neste exemplo, a probabilidade de cada símbolo é dada apenas pela amostra da mensagem considerada.

## 5. Árvores Binárias

---

- É frequente apresentar-se um determinado esquema de codificação binária através de uma **Árvore Binária**.
- A árvore binária permite a implementação simples de algoritmos que definem esquemas de codificação adequados – de elevada **eficiência**.
- Este diagrama permite, ainda, uma perceção visual imediata sobre a abordagem considerada na criação de um esquema de codificação, facilitando, ainda, possíveis otimizações.
- Uma árvore binária é composta por uma sequência de ramificações duplas (ao ramo da esquerda associa-se o bit 0, e ao da direita o bit 1), e que culminam nos símbolos a codificar (as folhas da árvores).
- O código de cada símbolo é o percurso seguido até esse símbolo.

# 5. Árvores Binárias

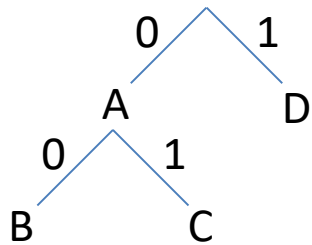


Símbolo	Código
A	0
B	10
C	11

O código de um símbolo é o caminho desde a raiz da árvore até ao elemento correspondente.

# 5. Árvores Binárias

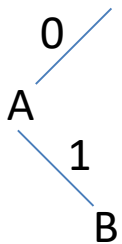
- Ambiguidade e Instantaneidade em árvores binárias:



Símbolo	Código
A	0
B	00
C	01
D	1

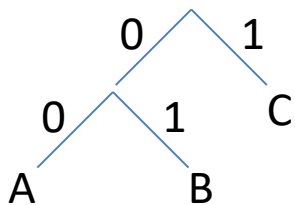
Código ambíguo e não instantâneo.

001 – pode ser AAD, BD ou AC.



Símbolo	Código
A	0
B	01

Código não ambíguo e não instantâneo.



Símbolo	Código
A	00
B	01
C	1

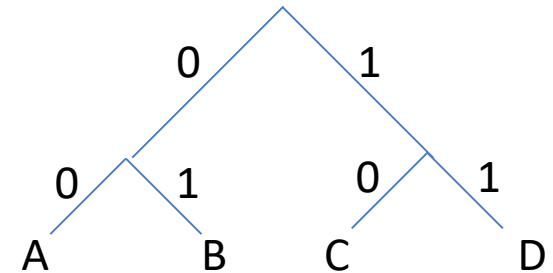
Código não ambíguo e instantâneo.

Verifica-se quando todos os símbolos se encontram em nós terminais (folhas).

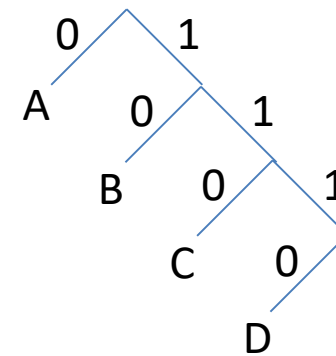
# 5. Árvores Binárias

- Exemplo:

Símbolo	Código
A	00
B	01
C	10
D	11



Símbolo	Código
A	0
B	10
C	110
D	1110



## 6. Código de Huffman

---

- O **código de Huffman** é o resultado de um algoritmo que define a forma de construção da respetiva árvore binária, e cuja eficiência de codificação é elevada.
- Este código considera a lista de símbolos presentes numa mensagem e o respetivo número de ocorrências nessa mensagem.
- Considere-se, então, a título de exemplo ilustrativo, a seguinte mensagem:

AABCADBABADABAABCABCAACACAAA

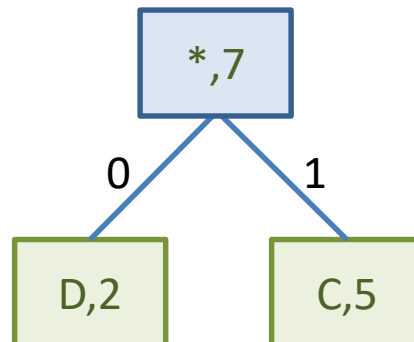
Símbolo	Ocorrências
A	15
B	6
C	5
D	2

## 6. Código de Huffman

AABCADBABADABAABCABCAACACAAA

Símbolo	Ocorrências
A	15
B	6
C	5
D	2

- Agrupa-se os dois símbolos de menor ocorrência e cria-se o elemento do nível imediatamente superior da árvore, com o valor da soma das ocorrências dos dois símbolos.
- O símbolo de menor ocorrência situa-se sempre à esquerda.

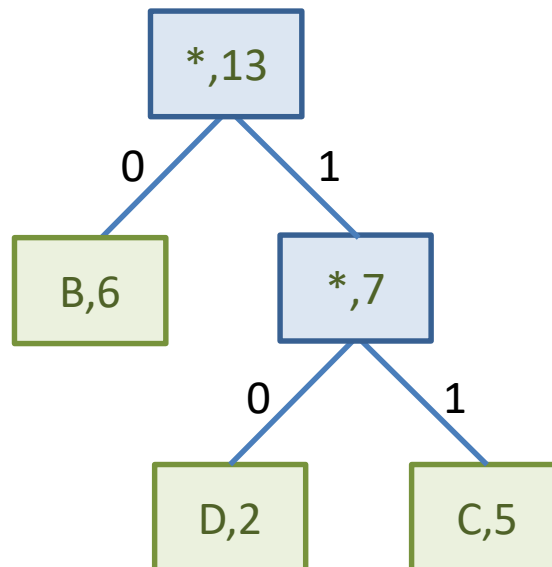


## 6. Código de Huffman

AABCADBABADABAABCABCAACACAAA

Símbolo	Ocorrências
A	15
B	6
C	5
D	2

- Adiciona-se o próximo símbolo (de menor ocorrência) no nível anteriormente criado.
- O símbolo de menor ocorrência situa-se sempre à esquerda.





## 6. Código de Huffman

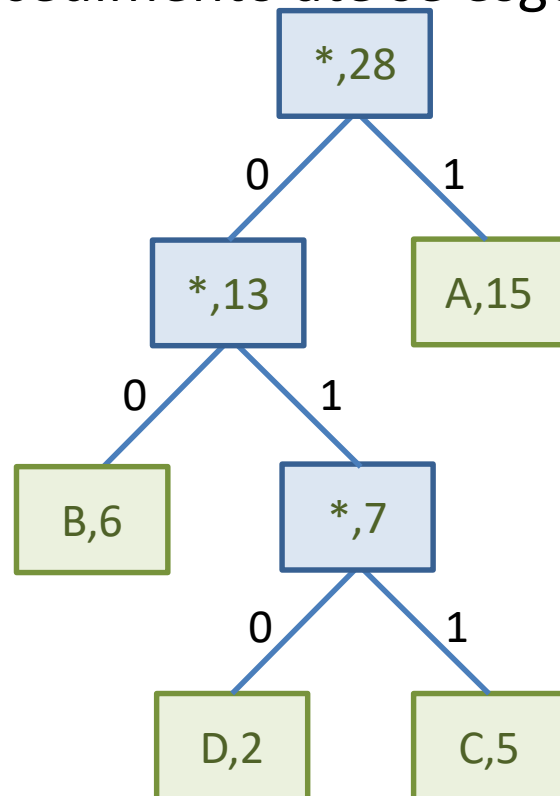
AABCADBABADABAABCABCAACACAAA

Símbolo	Ocorrências
A	15
B	6
C	5
D	2

- Repete-se o procedimento até se esgotar os símbolos.

Resultado:

Símbolo	Código
A	1
B	00
C	011
D	010



Número de bits necessários para representar a mensagem:

$$15 \cdot 1 + 6 \cdot 2 + 5 \cdot 3 + 2 \cdot 3 = 48$$

Número médio de bits por símbolo:

$$48/28 = 1.71 \text{ bps}$$

Entropia da mensagem: 1.67 bps