

Comprensión de archivos

Códigos de Huffman

Ejemplos: Suponga un archivo con:

1) Abacdababcbabcdeabaac

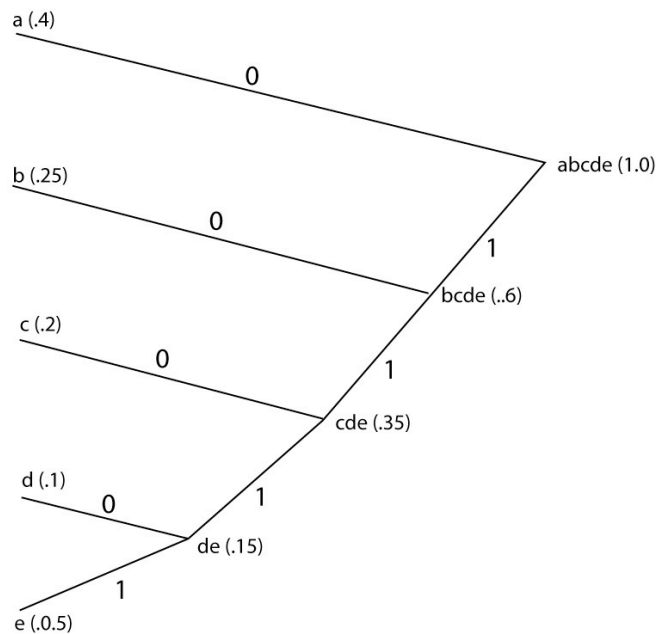
Pasos:

- Realizar “conteo” por cada símbolo
- Calcular probabilidades (por cada símbolo)

Simbol	a	b	c	d	e	
Cantidad	8	5	4	2	1	= 20 Bytes
Prob.	8/20	5/20	4/20	2/20	1/20	
Prob.	0.4	0.25	0.2	0.1	0.05	

Formar el Árbol Binario de “Re-Codificación”

- Tome las dos menores probabilidades y construya un nuevo “padre” con su Sumatoria (las 2 prob’s).
- Con la nueva prob. siga construyendo “padres” como el paso anterior.
- El algoritmo termina cuando se crea el “padre total”...(cuya probabilidad es igual a 1!)
- Codifique cada “rama”...o si está a la izquierda...1 si está a la derecha.
- El código de cada símbolo se obtiene al “coleccionar” los 1’s y 0’s desde la “raíz” o padre hasta el símbolo



Recodificar símbolos

Simbol	a	b	c	d	e	
Cantidad	8	5	4	2	1	20 Bytes
Cód.	0	10	110	1110	1111	
Bits.	1x8	2x5	3x4	4x2	4x1	5 bytes + 2 bits

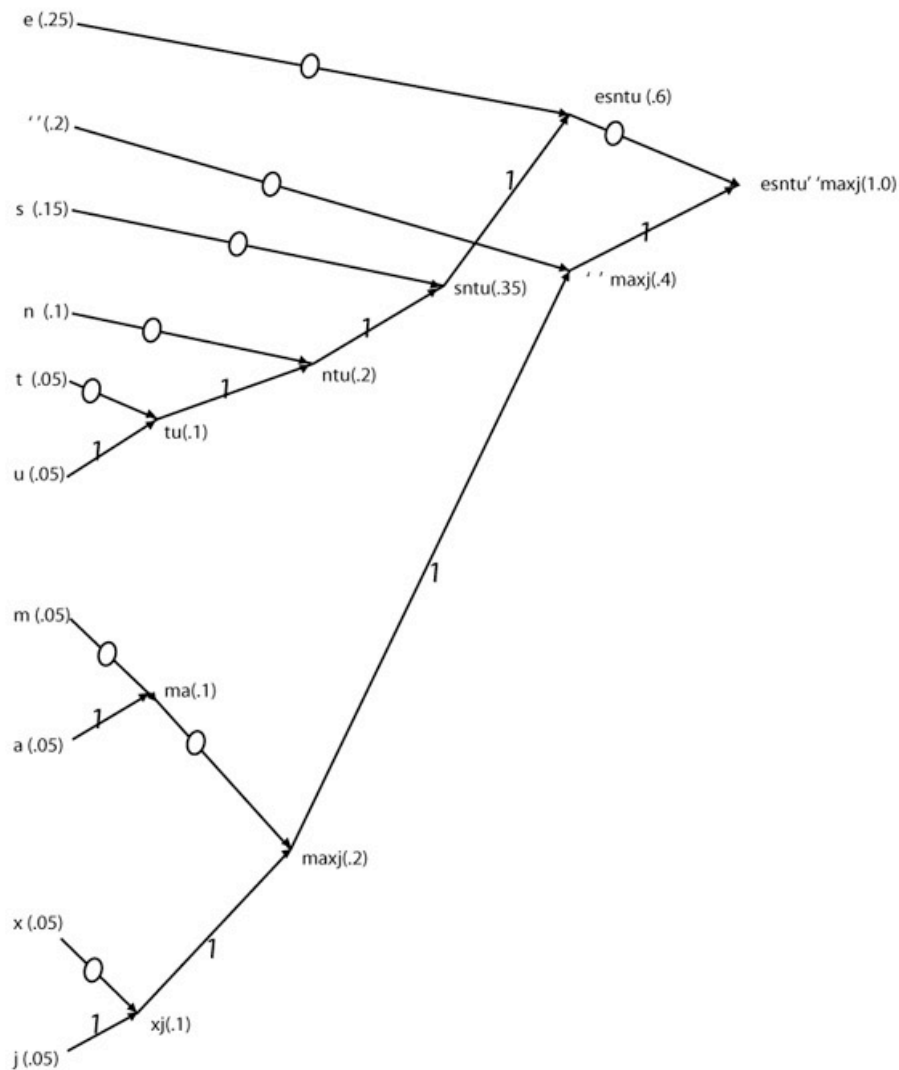
Luego escriba en el archivo el “código nuevo” de cada símbolo...de una forma “continua” (es conveniente guardar un “encabezado con las equivalencias”)

Ventajas

- Se logra máxima compactación / Es muy eficiente cuando hay muchas repeticiones.
- Ningún código nuevo (por cada símbolo) es PREFIJO de otro.

2) este es un mensaje x

Simbol	e	s	t	' '	u	m	n	a	x	j
Cantidad	5	3	1	4	1	1	2	1	1	1
Prob.	.25	.15	.05	.20	.05	.05	.10	.05	.05	.05



Práctica

- 1) Haga la tabla de “códigos” por símbolo...
- 2) Haga los “algoritmos necesarios” para “descomprimir” o sea, leer los códigos binarios del archivo “comprimido”... interpretarlos... compararlos contra una “tabla de equivalencias”... y generar los códigos originales.
- 3) Realice bien los puntos anteriores..ya que esto es parte de la tarea programada #2.