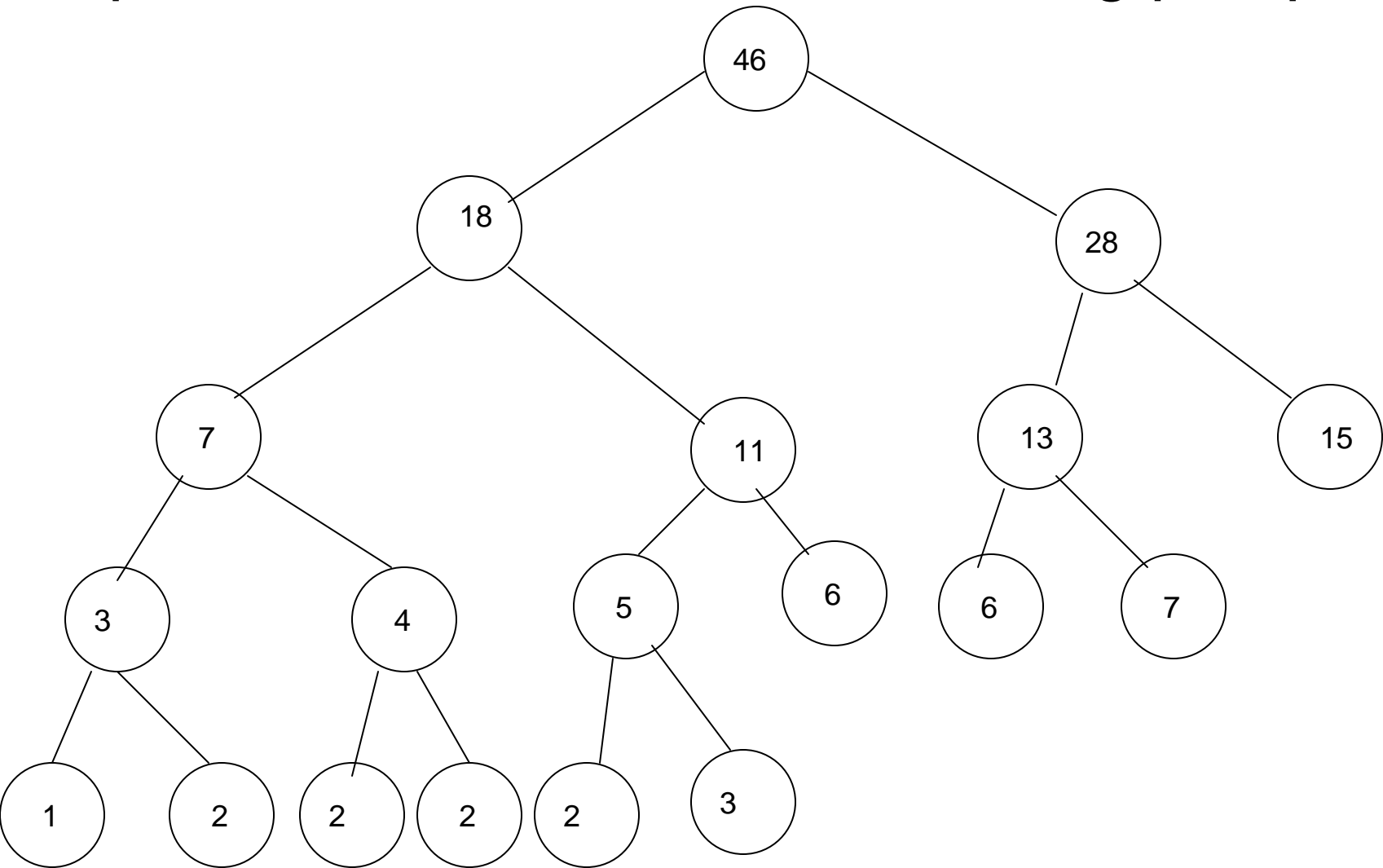


# Distribution free data compression algorithms

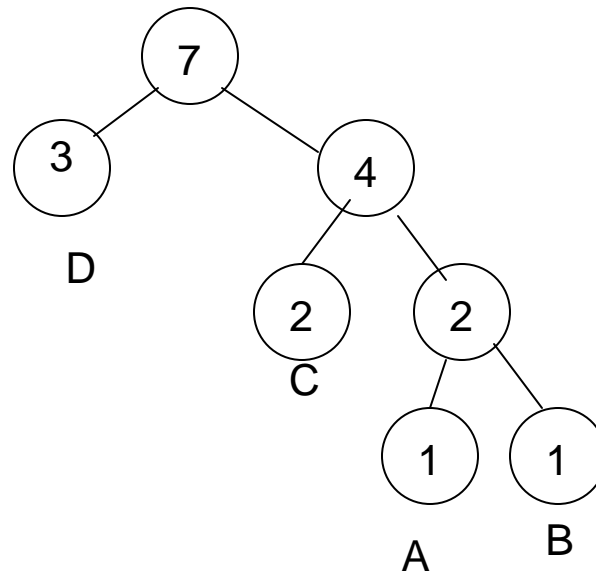
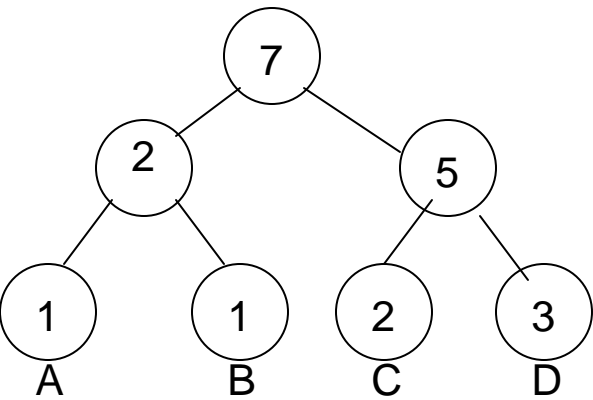
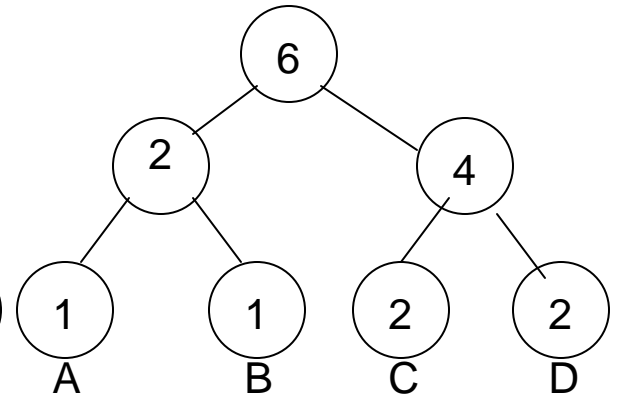
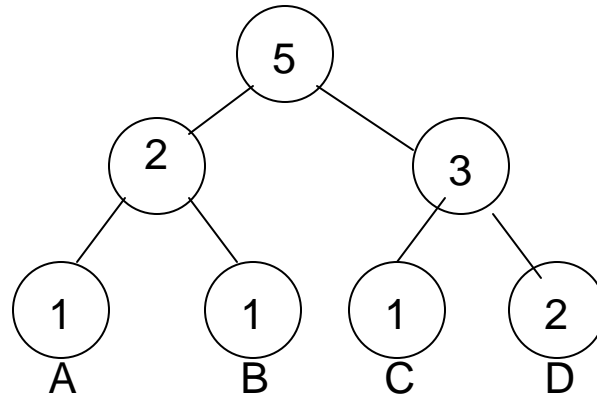
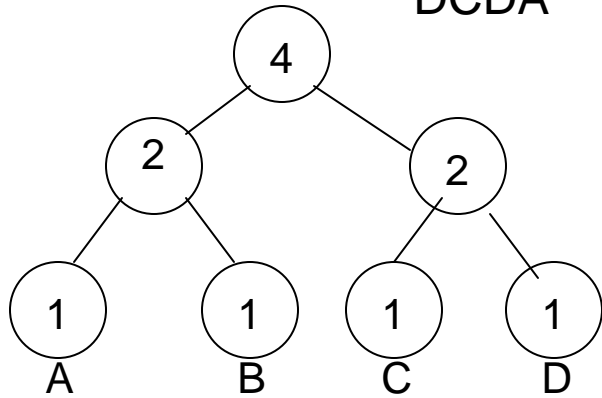
# Adaptive Huffman codes – the sibling pair property



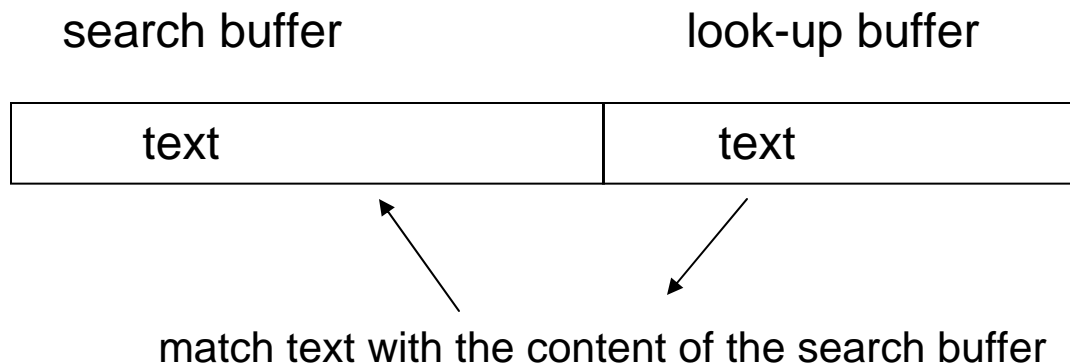
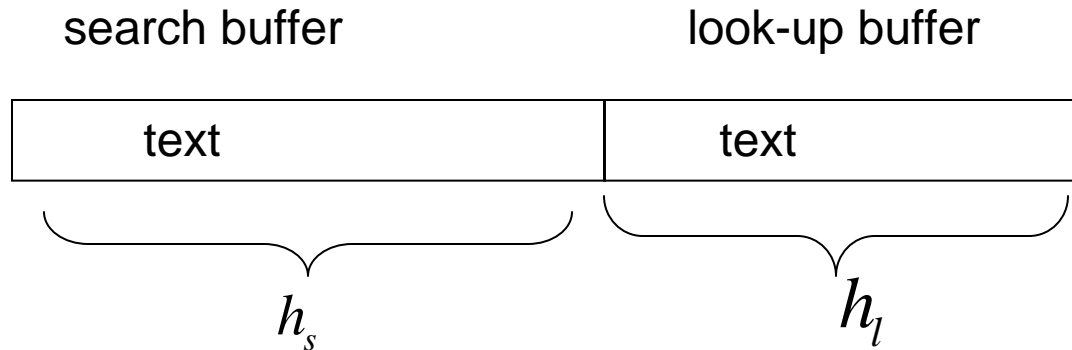
1,2,2,2,2,3; 3,4,5,6,6,7; 7,11,13,15; 18,28; 46 monotone increasing sequence from bottom layer to top layer

# Construction of adaptive Huffman codes

DCDA



# LZ77 algorithm



**Output:** starting position in the search buffer, length of coinciding symbols,  
code of the next symbol       $\langle p, l, n \rangle \longrightarrow \lceil \log h_s \rceil + \lceil \log h_l \rceil + \lceil \log \chi \rceil$

# Example

$h_s = 7, h_l = 6$  Text: ...cabracadabrarrard...

Initial state: 

c	a	b	r	a	c	a	d	a	b	r	r	a	r	a	d	...
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----

 Output:  $\langle 0, 0, f(d) \rangle$

↔  
7  
↔

Next state: c 

a	b	r	a	c	a	d	a	b	r	r	a	r	a	d	...
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	-----

 Output:  $\langle 7, 4, f(r) \rangle$

↔  
4  
↔

Next state: c a b r a c a 

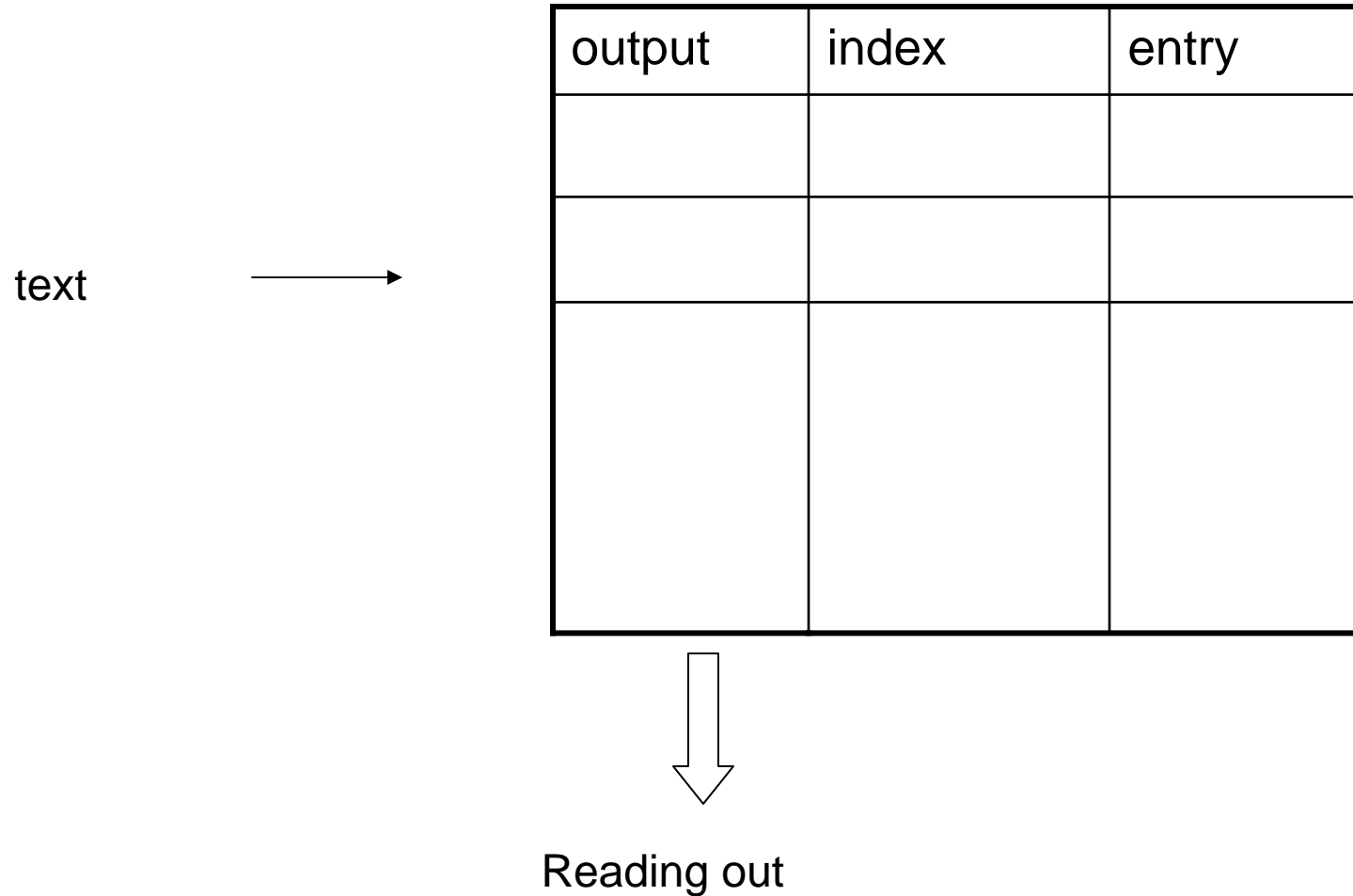
d	a	b	r	r	a	r	r	a	d	...
---	---	---	---	---	---	---	---	---	---	-----

 Output:  $\langle 3, 5, f(d) \rangle$

↔  
3  
↔

↔  
5  
↔

# LZ78 algorithm



# Example

Text: d|a|b|b|a|c|d|a|b|b|a|c|d|a|b|b|a|c|d|e|e|c|d|e|e|c|d|e|e|

Output	index	entry
$(0, f(d))$	1	<i>d</i>
$(0, f(a))$	2	<i>a</i>
$(0, f(b))$	3	<i>b</i>
$(3, f(a))$	4	<i>ba</i>
$(0, f(c))$	5	<i>c</i>
$(1, f(a))$	6	<i>da</i>
$(3, f(b))$	7	<i>bb</i>
$(2, f(c))$	8	<i>ac</i>
$(6, f(b))$	9	<i>dab</i>
$(4, f(c))$	10	<i>bac</i>
$(9, f(b))$	11	<i>dabb</i>
$(8, f(d))$	12	<i>acd</i>
$(0, f(e))$	13	<i>e</i>
$(13, f(c))$	14	<i>ec</i>
$(1, f(e))$	15	<i>de</i>
$(14, f(d))$	16	<i>ecd</i>
$(13, f(e))$	17	<i>ee</i>