

ENCODE (*uncompressed_data*, *i*, *claimed*, **var** *newly_claimed*, **var** *encoded*, **var** *length_stream*)

1	<i>byte_index</i> ← 0
2	<i>bit_index</i> ← 0
3	<i>j</i> ← <i>i</i> + 1
4	<i>skipped</i> ← 0
5	<i>encoded</i> ← 0
6	<i>stream</i> [0] ← 00000000 ₂
7	while <i>j</i> - <i>i</i> ≤ 2040 and <i>j</i> + <i>skipped</i> < length(<i>uncompressed_data</i>) do
8	if not <i>claimed</i> [<i>j</i> + <i>skipped</i>] then
9	if <i>uncompressed_data</i> [<i>j</i> + <i>skipped</i>] = <i>uncompressed_data</i> [<i>i</i>] then
10	Set 2 ^{<i>bit_index</i>} bit of <i>stream</i> [<i>byte_index</i>] to binary 1
11	<i>newly_claimed</i> [<i>j</i> + <i>skipped</i>] ← <i>true</i>
12	<i>encoded</i> ← <i>encoded</i> + 1
13	else Set 2 ^{<i>bit_index</i>} bit of <i>stream</i> [<i>byte_index</i>] to binary 0
14	<i>bit_index</i> ← <i>bit_index</i> + 1
15	if <i>bit_index</i> > 7 then
16	<i>bit_index</i> ← 0
17	<i>byte_index</i> ← <i>byte_index</i> + 1
18	<i>stream</i> [<i>byte_index</i>] ← 00000000 ₂
19	<i>j</i> ← <i>j</i> + 1
20	else <i>skipped</i> ← <i>skipped</i> + 1
21	while <i>byte_index</i> ≥ 0 and <i>stream</i> [<i>byte_index</i>] = 00000000 ₂ do
22	<i>byte_index</i> ← <i>byte_index</i> - 1
23	<i>length_stream</i> ← <i>byte_index</i> + 1
24	return <i>stream</i>