ACORN:
A Lightweight Authenticated Cipher
(v3)

Designer and Submitter: Hongjun Wu

Division of Mathematical Sciences
Nanyang Technological University
wuhongjun@gmail.com

2016.09.15

Contents

Chapter 1

Specification

The specifications of ACORN-128 are given in this chapter.

1.1 Recommended parameter sets

- Primary Recommendation: ACORN-128
  128-bit key, 128-bit nonce, 128-bit tag
  Use Case 1: Lightweight applications (resource constrained environments)
  Use Case 2: High-performance applications

1.2 Operations, Variables and Functions

The operations, variables and functions used in ACORN are defined below.

1.2.1 Operations

The following operations are used in ACORN:

$\oplus$	:	bit-wise exclusive OR
&	:	bit-wise AND
$\sim$	:	bit-wise NOT
$\|\|$	:	concatenation

1.2.2 Variables and constants

The following variables and constants are used in ACORN:

AD	:	associated data (this data will not be encrypted or decrypted).
ad_i	:	one bit of associated data block.
$adlen$	:	bit length of the associated data with $0 \leq adlen < 2^{64}$.
C	:	ciphertext.

c_i	:	the ith ciphertext bit.
ca_i	:	a control bit at the ith step. It is used to separate the processing of associated data, the processing of plaintext, and the generation of authentication tag.
cb_i	:	another control bit at the ith step. It is used to allow a keystream bit to affect a feedback bit during initialization, processing of associated data, and the tag generation.
IV_{128}	:	128-bit initialization vector of ACORN-128.
$IV_{128,i}$	:	the ith bit of IV_{128}.
K_{128}	:	128-bit key of ACORN-128.
$K_{128,i}$	:	the ith bit of K_{128}.
ks_i	:	The keystream bit generated at the ith step.
$pclen$	:	bit length of the plaintext/ciphertext with $0 \leq pclen < 2^{64}$.
m_i	:	one data bit.
P	:	plaintext.
p_i	:	the ith plaintext bit.
S_i	:	state at the beginning of the ith step.
$S_{i,j}$	:	jth bit of state S_i . For ACORN-128, $0 \leq j \leq 292$.
T	:	authentication tag.
t	:	bit length of the authentication tag with $64 \leq t \leq 128$.

1.2.3 Functions

Two Boolean functions are used in ACORN: maj and ch.

$$\begin{aligned} maj(x, y, z) &= (x\&y) \oplus (x\&z) \oplus (y\&z) \ ; \\ ch(x, y, z) &= (x\&y) \oplus ((\sim x)\&z) \ ; \end{aligned}$$

1.3 ACORN-128

ACORN-128 uses a 128-bit key and a 128-bit initialization vector. The associated data length and the plaintext length are less than 2^{64} bits. The authentication tag length is less than or equal to 128 bits. We recommend the use of a 128-bit tag.

1.3.1 The state of ACORN-128

The state size of ACORN-128 is 293 bits. There are six LFSRs being concatenated in ACORN-128. The state is shown in Fig.1.1.

1.3.2 The functions of ACORN-128

There are three functions in ACORN-128: the function to generate keystream bit from the state, the function to compute the overall feedback bit, and the

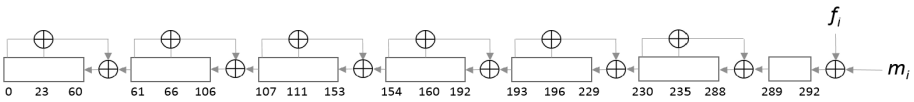

Figure 1.1: The concatenation of 6 LFSRs in ACORN-128. f_i indicates the overall feedback bit for the ith step; m_i indicates the message bit for the ith step.

function to update the state.

Generate the Keystream Bit. At each step, the keystream bit is computed using the function $ks_i = KSG128(S_i)$:

$$ks_i \quad = \quad S_{i,12} \oplus S_{i,154} \oplus maj(S_{i,235}, S_{i,61}, S_{i,193}) \oplus ch(S_{i,230}, S_{i,111}, S_{i,66});$$

Compute the Feedback Bit. At each step, the feedback bit is computed using the function $f_i = FBK128(S_i, ca_i, cb_i)$:

$$f_i \quad = \quad S_{i,0} \oplus (\sim S_{i,107}) \oplus maj(S_{i,244}, S_{i,23}, S_{i,160}) \oplus (ca_i \& S_{i,196}) \oplus (cb_i \& ks_i) \ ;$$

The State Update Function. At each step, the pseudo code for the state update function $S_{i+1} = \text{StateUpdate128}(S_i,\ m_i,\ ca_i,\ cb_i)$ is given as :

/*step 1. update using six LFSRs */
$S_{i,289} = S_{i,289} \oplus S_{i,235} \oplus S_{i,230}$;
$S_{i,230} = S_{i,230} \oplus S_{i,196} \oplus S_{i,193}$;
$S_{i,193} = S_{i,193} \oplus S_{i,160} \oplus S_{i,154}$;
$S_{i,154} = S_{i,154} \oplus S_{i,111} \oplus S_{i,107}$;
$S_{i,107} = S_{i,107} \oplus S_{i,66} \oplus S_{i,61}$;
$S_{i,61} = S_{i,61} \oplus S_{i,23} \oplus S_{i,0}$;

/*step 2. generate keystream bit */
$ks_i = KSG128(S_i)$;

/*step 3. generate the nonlinear feedback bit */
$f_i = FBK128(S_i, ca_i, cb_i)$;

/*step 4. shift the 293-bit register with the feedback bit f_i */
for $j := 0$ to 291 do $S_{i+1,j} = S_{i,j+1}$;
$S_{i+1,292} = f_i \oplus m_i$;

1.3.3 The initialization of ACORN-128

The initialization of ACORN-128 consists of loading the key and IV into the state, and running the cipher for 1792 steps.

1. Initialize the state S_{-1792} to 0.

2. Let $m_{-1792+i} = K_{128,i}$ for $i = 0$ to 127;
   Let $m_{-1792+128+i} = IV_{128,i}$ for $i = 0$ to 127;
   Let $m_{-1792+256} = K_{128,i \mod 128} \oplus 1$ for $i = 0$;
   Let $m_{-1792+256+i} = K_{128,i \mod 128}$ for $i = 1$ to 1535;

3. Let $ca_{-1792+i} = 1$ for $i = 0$ to 1791;
   Let $cb_{-1792+i} = 1$ for $i = 0$ to 1791;

4. for $i = -1792$ to -1, $S_{i+1} = \text{StateUpdate128}(S_i, m_i, ca_i, cb_i)$;

Note that in the initialization, the keystream bit is used to update the state since $cb_i = 1$.

1.3.4 Processing the associated data

After the initialization, the associated data AD is used to update the state.

1. Let $m_i = ad_i$ for $i = 0$ to $adlen - 1$;
   Let $m_{adlen} = 1$;
   Let $m_{adlen+i} = 0$ for $i = 1$ to 255;

2. Let $ca_i = 1$ for $i = 0$ to adlen+127;
   Let $ca_i = 0$ for $i =$ adlen+128 to adlen+255;
   Let $cb_i = 1$ for $i = 0$ to adlen+255;

3. for $i = 0$ to $adlen + 255$, $S_{i+1} = \text{StateUpdate128}(S_i, m_i, ca_i, cb_i)$;

Note that even when there is no associated data, we still need to run the cipher for 256 steps. When we process the associated data, the keystream bit is used to update the state since $cb_i = 1$. The cipher specification is changed for 128 steps (since the value of ca_i is set to 0 for 128 steps) so as to separate the associated data from the plaintext/ciphertext. Here separating the associated data from the plaintext/ciphertext means preventing using part of the associated data as ciphertext, or using part of the ciphertext as associated data.

1.3.5 The encryption

After processing the associated data, at each step of the encryption, one plaintext bit p_i is used to update the state, and p_i is encrypted to c_i.

1. Let $m_{adlen+256+i} = p_i$ for $i = 0$ to $pclen - 1$;
   Let $m_{adlen+256+pclen} = 1$;
   Let $m_{adlen+256+pclen+i} = 0$ for $i = 1$ to 255;

2. Let $ca_i = 1$ for $i = adlen + 256$ to $adlen + pclen + 383$;
   Let $ca_i = 0$ for $i = adlen + pclen + 384$ to $adlen + pclen + 511$;
   Let $cb_i = 0$ for $i = adlen + 256$ to $adlen + pclen + 511$;

3. for $i = adlen + 256$ to $adlen + pclen + 511$,
   $S_{i+1} = \text{StateUpdate128}(S_i, m_i, ca_i, cb_i)$;
   $c_{i-adlen-256} = p_{i-adlen-256} \oplus ks_i$;
   (when performing encryption, $0 \leq i - adlen - 256 \leq pclen - 1$)
   end for;

Note that even when there is no plaintext, we still need to run the cipher for 256 steps. When we process the plaintext, the keystream bit is not used to update the state since $cb_i = 0$. The cipher specification is changed for 128 steps (since the value of ca_i is set to 0 for 128 steps) so as to separate the processing of plaintext/ciphertext and the finalization. Here separating the finalization from the plaintext/ciphertext means preventing using part of the keystream (obtained from plaintext and ciphertext) as the authentication tag.

1.3.6 The finalization

After processing all the plaintext bits, we generate the authentication tag T.

1. Let $m_{adlen+pclen+512+i} = 0$ for $i = 0$ to 767;
2. Let $ca_i = 1$ for $i = adlen + pclen + 512$ to $adlen + pclen + 1279$;
   Let $cb_i = 1$ for $i = adlen + pclen + 512$ to $adlen + pclen + 1279$;
3. for $i = adlen + pclen + 512$ to $adlen + pclen + 1279$,
   $S_{i+1} = \text{StateUpdate128}(S_i, m_i, ca_i, cb_i)$;
   end for;

The authentication tag T is the last t keystream bits, i.e.,
$T = ks_{adlen+pclen+1279-t+1} \parallel ks_{adlen+pclen+1279-t+2} \parallel \cdots \parallel ks_{adlen+pclen+1279}$.

1.3.7 The decryption and verification

The decryption and verification are very similar to the encryption and tag generation. The finalization in the decryption process is the same as that in the encryption process. We emphasize that if the verification fails, the ciphertext and the newly generated authentication tag should not be given as output; otherwise, the state of ACORN-128 is vulnerable to known-plaintext or chosen-ciphertext attacks (using a fixed IV).

Chapter 2

Security Goals

The security goals of ACORN are given in Table 2.1. In ACORN, each key, IV pair is used to protect only one message. If verification fails, the new tag and the decrypted ciphertext should not be given as output.

Note that the authentication security in Table 2.1 includes the integrity security of plaintext, associated data and nonce.

Table 2.1: Security Goals of ACORN-128 (128-bit tag)

	Encryption	Authentication
ACORN-128	128-bit	128-bit

Chapter 3

Security Analysis

The following requirements should be satisfied in order to use ACORN securely.

1. Each key should be generated in a secure and random way.
2. Each key and IV pair should not be used to protect more than one message; and each key and IV pair should not be used with two different tag sizes.
3. If verification fails, the decrypted plaintext and the wrong authentication tag should not be given as output.

If the above requirements are satisfied, we have the following security claims:

Claim 1. The success rate of a forgery attack is 2^{-t}, where t is the tag size. If the forgery attack is repeated n times, the success rate of a forgery attack is about $n \times 2^{-t}$.

Claim 2. The state and key cannot be recovered faster than exhaustive key search if the forgery attack is not successful. We recommend the use of a 128-bit tag size for ACORN in order to resist repeated forgery attacks.

If an IV is reused in encryption, or if the plaintext is leaked in the failed verification, the state can be recovered easily. In [4], it is shown that if the IV is reused seven times, the security of ACORN is lost. We point out here that in the new version of ACORN, the secret key of ACORN cannot be recovered easily from the state since the initialization is now non-invertible if the secret key is unknown, so now ACORN protects the secret key against the nonce reuse attack.

According to our analysis, ACORN is a strong cipher. Since the design approach of ACORN is very new, we encourage the researchers to conduct thorough security analysis of ACORN.

In this chapter, we focus on the security analysis of authentication, since it is very challenging to design and analyze the differential propagation in an authenticated cipher based on a bit-oriented sequential stream cipher.

3.1 The security of initialization

The initialization can be attacked by analyzing the relation between IV and keystream. In ACORN-128, the IV passes through at least 1792 steps before affecting ciphertext. The initialization is designed to prevent various attacks against stream cipher initialization: the linear attack (such as the attack in [22]), differential attacks (such as the attacks in [24] and [23]) and cube attacks [9, 10].

3.1.1 Differential analysis of initialization

In this section, we analyze the difference propagation in the state. In our analysis, we introduce difference into IV, then compute the differential probability in each step. The initialization of ACORN is invertible (for a given known key), so every difference in the IV passes through the complete initialization.

In the initialization, $f_i \oplus ks_i$ is used as the nonlinear feedback bit. In our first experiment, we assume that after the IV being loaded in the state, there is a difference at $state_{292}$ at the beginning of the ith step, and every state bit at this step is secret. We also assume that the difference in $f_j \oplus ks_j$ is always eliminated whenever possible. The differential probability is 2^{-227} after 400 steps.

In our second experiment, we tried to improve the first experiment so as to minimize the difference in the right most LFSR. For example, if there is difference in $S_{j,234} \oplus S_{j,239}$, and if there is chance to introduce difference in $f_j \oplus ks_j$, we always introduce difference in $f_j \oplus ks_j$ so as to eliminate a difference bit in the right most LFSR at the $(j+4)$th step (here S_j is the state at the jth step after the operation of those 6 LFSRs, but before the shift operation). The differential probability is 2^{-232} after 400 steps, not better than the first experiment.

In our third experiment, we tried to improve the first experiment so that there are more difference bits at the beginning of the ith step. For example, we tested the differences at $S_{i,230}$, $S_{i,235}$ and $S_{i,289}$. The differential probability is 2^{-293} after 400 steps, not better than the first experiment. Another example, we tested the differences at $S_{i,61}$, $S_{i,66}$ and $S_{i,107}$. The differential probability is 2^{-289} after 400 steps, not better than the first experiment. It indicates that introducing more initial difference bits may give even lower differential probability.

From the above experiments, we think that the ACORN initialization has large security margin against differential cryptanalysis.

3.1.2 Cube analysis of initialization

Cube attack is effective against the ciphers with low algebraic degree, or against the ciphers with high algebraic degree but the system of nonlinear equations of the cipher is very sparse. In the ACORN initialization, there are 6 linear feedback shift registers, and there are 14 taps being used in the feedback (keystream

is also used for feedback in the initialization), we expect that the system of equations of ACORN would be dense.

We performed experiments to estimate the algebraic degrees of state bits in terms of the input. In particular, we analyze whether the term $iv_{i_0} \cdot iv_{i_1} \cdot iv_{i_2} \cdot iv_{i_3} \cdots iv_{i_{n-1}}$ affects every state bits state bits. In our experiments, we focus on the term $iv_{i_0} \cdot iv_{i_1} \cdot iv_{i_2} \cdot iv_{i_3} \cdots iv_{i_{n-1}}$ which is the product of the last n bits of IV. Note that in the initialization, the ith IV bit is XORed to the state at the $(256+i)$th step, so the last n bits of the IV bits are expected to have the least effect on the state. In the experiments, we set the rest of the IV bits to zero, and use 16 random keys for confirmation.

Table 3.1 shows the minimum number of steps that are needed so that every state bit is affected by the term $iv_{128-n} \cdot iv_{128-n+1} \cdots iv_{127}$, which is the product of the last n bits of IV. Note that for $n = 1$, it takes 669 steps so that the last IV bit iv_{127} affects all the state bits. The reason is that iv_{127} is XORed to the state at the 255th step, then it takes at least 292 steps for this bit being shifted through the state, so at least 548 steps are needed for iv_{127} to affect every state bit.

Table 3.1: The minimum number of steps for every state bit being affected by $iv_{128-n} \cdot iv_{128-n+1} \cdots iv_{127}$

n	steps	n	steps	n	steps	n	steps
1	669	9	891	17	931	25	955
2	736	10	894	18	933	26	957
3	781	11	894	19	933	27	961
4	796	12	896	20	938	28	964
5	806	13	906	21	942	29	965
6	843	14	912	22	949	30	966
7	852	15	923	23	950	31	972
8	869	16	931	24	954	32	974

From Table 3.1, our observation is that as the cube size increases, the minimum number of steps increases at a decelerated pace. From cube size 1 to 8, the minimum number of steps increases from 669 to 869 (increment: 200 steps); from cube size 9 to 16, the minimum number of steps increases from 891 to 931 steps (increment: 40 steps); from cube size 17 to 24, the minimum number of steps increases from 931 to 954 (increment: 33 steps); from cube size 25 to 32, the minimum number of steps increases from 955 to 974 (increment: 19 steps). Even if we assume that the increment of the minimum of steps is less than 4 as the cube size increases by 1, the minimum number of steps is at most $974 + 4 \times 96 = 1358$. We thus expect that ACORN has large security margin against the cube attack.

3.2 The security of encryption

We emphasize here that ACORN encryption is a stream cipher with a large state which is updated continuously. The attacks against a block cipher cannot be applied directly to ACORN.

3.2.1 Statistical attacks against encryption

If the IV is used only once for each key, it is impossible to apply a differential attack to the encryption process. It is extremely difficult to apply a linear attack to recover the secret state since the state of ACORN is updated in a nonlinear way. In general, it would be difficult to apply any statistical attack to recover the secret state due to the nonlinear state update function (the statistical correlation between any two states vanishes quickly as the distance between them increases).

3.2.2 Traditional attacks against encryption

The traditional attacks against stream ciphers (such as fast correlation attack, algebraic and fast algebraic attack) exploit the linear state update function in stream ciphers. The state of ACORN is updated in a nonlinear way, and every state bit affects the whole state, so ACORN is strong against those powerful attacks on stream ciphers (correlation and fast correlation attack [21, 20, 17, 18, 5], algebraic attack and fast algebraic attacks [6, 7]).

Time-memory-data trade-off attacks are powerful against stream ciphers with small states [2, 13, 3]. **In general, if the state size of a stream cipher is at least twice of the key size, the stream cipher is strong against the time-memory-data trade-off attacks.** For example, for a stream cipher with n-bit key and $2n$-bit state, the simple time-memory-data trade-off attack [2, 13] requires 2^n data and 2^n memory in order to break the cipher with time 2^n. In the advanced time-memory-data trade-off attack [3], under the condition that the pre-computing time does not exceed 2^n, the attack requires 2^n data, and the attack complexity is $T \times M^2 = 2^{2n}$, where T and M are time and memory, respectively (there is additional constraint on the time and data). Note that if we apply Hellman's time-memory trade-off attack [16] directly to this stream cipher, the attack complexity is 2^n pre-computation, negligible amount of data and $T \times M^2 = 2^{2n}$. Hellman's time-memory trade-off attack is a general attack against all the symmetric key ciphers, so **any attack which is more expensive than Hellman's time-memory trade-off attack is ineffective**. The results above show that for a stream cipher with n-bit key and $2n$-bit state, the time-memory-data trade-off attack performs much worse than Hellman's time-memory trade-off attack (especially if we conside the data), so **the time-memory-data trade-off attack is not a threat to a stream cipher with $2n$-bit state in practice**.

The state size of ACORN is more than twice of the key size, so ACORN is strong against the time-memory-trade-off attack.

3.2.3 Guess-and-determine attack on encryption

Guess-and-determine attack can be applied to analyze the stream ciphers based on linear or nonlinear state update functions [14, 19, 12]. In the guess-and-determine attack, part of the secret information is guessed so as to recover more state information.

In ACORN, each keystream bit gives a nonlinear equation involving the state bits:

$ks_i = S_{i,12} \oplus S_{i,154} \oplus maj(S_{i,235}, S_{i,61}, S_{i,193}) \oplus ch(S_{i,230}, S_{i,111}, S_{i,66})$;

If we guess that $maj(S_{i,235}, S_{i,61}, S_{i,193})$ always gives the output $S_{i,61}$ (with probability $\frac{3}{4}$), and the function $ch(S_{i,230}, S_{i,111}, S_{i,66})$ always gives the output $S_{i,66}$ (assume $S_{i,230} = 0$), we obtain a linear equation

$$ks_i = S_{i,12} \oplus S_{i,154} \oplus S_{i,61} \oplus S_{i,66};$$

The probability for this linear function to hold is $\frac{3}{4} \times \frac{1}{2} = \frac{3}{8}$. Thus with probability $\frac{3}{8}$, we can obtain 2 linear equations involving the state bits (another linear equation is $S_{i,230} = 0$). Even if we assume that the nonlinear feedback function does not affect the guess-and-determine attack, we need 293 linear equations to recover the state, and the probability to obtain 293 linear equations is at most $(\frac{3}{8})^{292/2} = 2^{-206}$. The analysis above shows that ACORN has large security margin against the guess-and-determine attack.

3.3 The security of message authentication before finalization

A common approach to attack ACORN authentication is to inject a difference into the state by modifying ciphertext or associated data. Ensuring the security of authentication is the most challenging part in the design and security analysis of ACORN.

A main feature of ACORN-128 is the concatenation of 6 small LFSRs, as shown in Fig. 1.1. The concatenation of six LFSRs ensures that once a difference bit is injected into the state (the first difference bit must be injected into the state through m_i), there are many difference bits in the state before the state difference gets eliminated.

To eliminate the difference in the right most LFSR, the input difference to that LFSR should have the following linear recurrence (in order to reduce the number of difference bits in the state, we consider only the shortest linear recurrence):

$$d_n = d_{n-59} \oplus d_{n-54}$$

Similarly, in order to eliminate the difference in each of the other five LFSRs, the input difference should have the following linear recurrences:

$$d_n = d_{n-37} \oplus d_{n-34}$$
$$d_n = d_{n-39} \oplus d_{n-33}$$
$$d_n = d_{n-47} \oplus d_{n-43}$$
$$d_n = d_{n-46} \oplus d_{n-41}$$
$$d_n = d_{n-61} \oplus d_{n-38}$$

Combining the above six linear occurrences, the input difference to those six registers is given below. There are other input differences, but this difference is likely the shortest one (290 bits).

```
10000000000000000000000000000000011001110101001100000010000101000
00010011111000100011101101010110100011001011101011001011101000010
11011111000101111101111101001011100111001000111011100100111010100
00100100111001100111111111111100100001011101000000110010110000000
01011011000001101101000010110110011
```

Once the input difference to those six registers is known, we are able to compute the probability that the difference in the state can be eliminated. **Note that the difference in the overall feedback bit f_i can always get eliminated with certain probability by modifying the ciphertext or associated data**. The details of the computation are given in Appendix A and B.

The analysis of the difference in associated data and the analysis of the difference in ciphertext are similar (except that the ciphertext leaks the state information which is useful for the attack). The reason is that when the associated data gets processed, the keystream bit is used as part of the feedback bit; when the ciphertext gets processed, the keystream is generated and xored to the ciphertext, then the decrypted plaintext bit is used as part of the feedback bit (i.e., the keystream bit also affects the input difference to the LFSRs).

We note that at each step, there are three nonlinear functions being involved: two maj functions and one ch function. The differential property of function maj is that if there are one or two input difference bits, then the output difference is 1 with probability 0.5; if there are three input difference bits, then the output difference is 1 with probability 1. The differential property of function $ch(x, y, z)$ is that if there are differences in both y and z, then the output difference is one with probability 1; otherwise, if there is any difference in the input, the output difference is 1 with probability 0.5.

According to the above input difference to the LFSRs and the differential properties of the nonlinear functions, the probability to eliminate the difference in the state is 2^{-181} (in a successful attack, proper difference should be injected into associated data or ciphertext so that the input difference to LFSRs is not affected by the nonlinear functions). With this differential probability, ACORN is able to provide 128-bit MAC security.

Note that in the forgery attack, an attacker can only modify the ciphertext or the associated data. If the IV is reused, an attacker can modify the plaintext so that two different plaintexts can have the same authentication tag. For ACORN, the success rate of the forgery attack under IV reuse is 2^{-120}. However, if IV

is reused in ACORN, there is more serious state recovery attack, and messages can be forged once the state is recovered. So this type of forgery attack is not a concern in the design of ACORN.

3.4 The security of finalization

The finalization stage is to resist the forgery attack. In our analysis in Section 3.1.1, the differential probability is less than 2^{-200} after 400 steps. We thus expect that the finalization of ACORN has large security margin against the forgery attack.

Chapter 4

Features

- Novel design. ACORN is a bit-based sequential authenticated cipher in which the difference is injected into the state for authentication for better performance. It is the first time that this type of authenticated cipher is designed based on bit-based sequential stream cipher. The challenge in this design approach is to achieve high authentication security, and we solved this problem by using six concatenated linear feedback shift registers to ensure that it is expensive to eliminate the difference in the state, and it is easy to analyze the authentication security.
- One message bit bit is processed in each step. This feature benefits lightweight hardware implementation, and the control circuit in the hardware implementation can be greatly simplified.
- ACORN allows parallel computation. In ACORN, 32 steps can be computed in parallel. This parallel feature benefits high speed hardware and software implementation.
- Length information of associated data and plaintext/ciphertext is not needed in ACORN, i.e., ACORN does not need to check the length of message, and ACORN does not need to pad the message to a multiple of block size (the length of the bits being padded is fixed in ACORN). This feature reduces further the cost of hardware implementation.
- Efficient in Hardware.
  According to our estimation, the hardware cost of ACORN-128 is slightly higher than that of TRIVIUM [8], which is very efficient in hardware.
- Efficient in Software
  In ACORN, 32 steps can be computed in parallel, so its software speed is reasonably fast.
- ACORN has several advantages over AES-GCM: ACORN is more hardware efficient than AES-GCM (especially for hardware resource and energy consumption). On the general computing devices (no AES-NI and no

polynomial computing circuits), ACORN is more efficient than AES-GCM in software. The code size of ACORN is very small.

Chapter 5

The Performance of ACORN

Hardware Performance. Tao Huang implemented ACORN in VHDL. In one implementation, 8 steps are computed in parallel, so the cipher processes 8 bits at the same time. This version is fast and small in hardware. On the Xilinx Virtex 7, 499 LUTs are used, and the speed is 3.4 Gbits/s. Currently this area is much smaller than the other CAESAR candidates [11]. It can be expected that the area can be reduced further if we compute only one or two steps at the same time.

In another implementation, 32 steps are computed in parallel, so the cipher processes 32 bits at the same time. This version is quite fast in hardware. On Xilinx Virtex 7, 979 LUTs are used, and the speed is 11.3 Gbits/s.

Software Performance. We implemented ACORN in C code. We tested the speed on Intel Core i7-6550U 2.2GHz processor (Skylake) running 64-bit Windows 10 and turning off the Turbo Boost. The compiler being used is gcc 5.3.0 of Cygwin64 2.874, and the optimization option "-O3" is used. In our test, associated data is not considered and 128-bit tag is used. The test is performed by processing a message repeatedly and printing out the final message.

Table 5.1: The speed (cpb) of ACORN for different message length on Intel Skylake processor

	64B	128B	256B	512B	1024B	2048B	4096B
encryption	38.2	23.2	15.4	11.8	11.2	8.8	8.2
decryption	37.8	22.1	14.3	10.5	8.4	7.5	7.1

Chapter 6

Design Rationale

ACORN is designed to be efficient in hardware (focus), and also efficient in software.

In order to be efficient in hardware, we use a bit-based stream cipher for its well-known hardware efficiency (such as A5/1 [1], Grain [15] and Trivium [8]). In order to resist the traditional attacks (correlation attacks [21, 20, 17, 18, 5] and algebraic attacks [6, 7]) on stream cipher, the state is updated in a nonlinear way, and every state bit affects the whole state.

We inject message into the state so that we could obtain authentication security almost for free. The challenge is that in a bit-based sequential stream cipher based on nonlinear feedback registers, it is tremendously difficult to trace the differential propagation in the state, especially if we want to achieve high authentication security (such as 128-bit). Our design focus is to solve this problem so that the authentication security could be easily analyzed. Our solution is to use the concatenation of several linear feedback shift registers to ensure that once there is difference in the state, the number of difference bits in the state would be sufficiently large before the difference gets eliminated. When there are difference bits in the state, the nonlinear function FBK introduces difference noise to the feedback bit f_i so as to reduce the success rate of forgery attack. If an attacker intends to modify the ciphertext, the difference in the keystream bits would also affect the state through the decrypted plaintext bits.

In order to further reduce the hardware complexity, ACORN does not check the message length in decryption and verification, and the padding bits (both length and values) are always fixed. In order to separate the processing of associated data and plaintext/ciphertext, the cipher feedback is modified for 128 steps (through modifying ca_i) before the plaintext/ciphertext gets processed. Similarly, in order to separate the processing of plaintext/ciphertext and the finalization, the cipher feedback is modified (through modifying ca_i) for 256 bits before the finalization. Separating the plaintext from the associated data means that an attacker cannot use part of the plaintext bits as associated data, and vice versa. Separating the encryption/decryption from the finalization means that an attacker cannot use part of the keystream as authentication tag.

In order to be fast in hardware and software, 32 steps of ACORN can be computed in parallel.

In order to have high security for stream cipher, when we select the tap positions, we try to have tap distances which are prime or contain some large prime factor.

In order to resist the differential attack against ACORN for fixed IV (such as the differential attack against Phelix [25]), we require that each key/IV pair is used to protect only one message, and the decrypted plaintext should not be disclosed if the verification fails.

6.1 Response to the review comments

The anonymous CAESAR committee members gave helpful comments on ACORN. We provide below explanation to some comments.

1. The purpose of introducing a small (four-bit) register at the input end of NLFSR is to ensure that the state update function is invertible (for the known key); otherwise, we have an non-invertible state update function which is not good for security (such as the differential attacks against the non-invertible initialization of Py [24] and ZUC [26]), and not good for the period of keystream.

   The reason that the size of the small register is 4 bits is that we get a 293-bit state. Here, 293 is prime, so this overall tap distance 293 is coprime to all the tap distances in the NLFSR. In the design of stream ciphers, the coprime tap distances are important to security.

2. Separating the associated data from plaintext/ciphertext means preventing an attacker from using part of the associated data as plaintext/cihertext, and vice versa. Separating the encryption/decryption from the finalization means preventing an attacker from reducing the size of plaintext/ciphertext, and use part of the keystream as authentication tag. The reason is that in ACORN, we generate the tag in a similar way as generating keystream (so as to reduce hardware cost).

3. In Appendix A, we provide all the differential components in the computation of the forgery success rate of 2^{-181}.

Chapter 7

Changes

7.1 Changes in the third round submission

Tweak. We tweaked ACORN in the third round submission. The tweak is to move the function $ch(S_{i,230}, S_{i,111}, S_{i,66})$ from the feedback function to the output filtering function. The tweak is to provide large security margin against the guess-and-determine attack. The tweak does not affect the performance of ACORN. The tweak does not affect the security analysis of the initialization and finalization, and the tweak does not affect the integrity analysis of ACORN.

The new output filtering function and nonlinear feedback function are given below:

$ks_i = S_{i,12} \oplus S_{i,154} \oplus maj(S_{i,235}, S_{i,61}, S_{i,193}) \oplus ch(S_{i,230}, S_{i,111}, S_{i,66})$;
$f_i = S_{i,0} \oplus (\sim S_{i,107}) \oplus maj(S_{i,244}, S_{i,23}, S_{i,160}) \oplus (ca_i \& S_{i,196}) \oplus (cb_i \& ks_i)$;

Security Analysis. We added the security analysis against differential cryptanalysis and cube attack in this document. And we provided the details of authentication security analysis in Appendix A and B. We explained that ACORN is strong against time-memory-data trade-off attack.

We corrected the linear recurrence of the right most LFSR in Section 3.3. It is $d_n = d_{n-59} \oplus d_{n-54}$, instead of $d_n = d_{n-59} \oplus d_{n-53}$. We updated the analysis accordingly, and the authentication security is 2^{-181} (instead of 2^{-189}).

7.2 Changes in the second round submission

We made the following tweaks in the second round submission:

1. The number of steps in the initialization, padding of associated data, padding of plaintext, and finalization are changed from 1536, 512, 512, 512 to 1792, 256, 256, 768, respectively.

The main reason from the change is to increase the steps in the initialization, so as to provide better protection of the secret key when nonce is reused.

2. In the initialization stage, the key bits are now used as inputs in 1664 steps. (In version 1, the key bits are used only in 128 steps.) The reason for this tweak is to strengthen the cipher against the nonce reuse attack (in encryption/decryption) so that the secret key cannot be easily recovered in the nonce reuse attack.

Chapter 8

No Hidden Weakness

We state here that the designer/designers have not hidden any weaknesses in this cipher.

Chapter 9

Intellectual property

We state that ACORN is not patented and it is freely available for all applications.

If any of this information changes, the submitter will promptly (and within at most one month) announce these changes on the `crypto-competitions` mailing list.

Chapter 10

Consent

The submitter hereby consents to all decisions of the CAESAR selection committee regarding the selection or non-selection of this submission as a second-round candidate, a third-round candidate, a finalist, a member of the final portfolio, or any other designation provided by the committee. The submitter understands that the committee will not comment on the algorithms, except that for each selected algorithm the committee will simply cite the previously published analyses that led to the selection of the algorithm. The submitter understands that the selection of some algorithms is not a negative comment regarding other algorithms, and that an excellent algorithm might fail to be selected simply because not enough analysis was available at the time of the committee decision. The submitter acknowledges that the committee decisions reflect the collective expert judgments of the committee members and are not subject to appeal. The submitter understands that if he disagrees with published analyses then he is expected to promptly and publicly respond to those analyses, not to wait for subsequent committee decisions. The submitter understands that this statement is required as a condition of consideration of this submission by the CAESAR selection committee.

Bibliography

[1] A5/1. Available at http://en.wikipedia.org/wiki/A5/1

[2] S. Babbage. "A Space/Time Tradeo in Exhaustive Search Attacks on Stream Ciphers." *European Convention on Security and Detection*, IEE Conference Publication No. 408, May 1995.

[3] A. Biryukov and A. Shamir. "Cryptanalytic Time/Memory/Data Tradeos for Stream Ciphers". In *Advances in Cryptology – Asiacrypt 2000*, pp. 1–13.

[4] C. Chaigneau, T. Fuhr and H. Gilbert. " Full key-recovery on ACORN in nonce-reuse and decryption-misuse settings." In CAESAR mailing list.

[5] V.V. Chepyzhov, T. Johansson and B. Smeets. "A Simple Algorithm for Fast Correlation Attacks on Stream Ciphers." In *Fast Software Encryption – FSE 2000*, pp. 181-195.

[6] N. T. Courtois. "Fast Algebraic Attacks on Stream Ciphers with Linear Feedback." In *Advances in Cryptology – CRYPTO 2003*, LNCS 2729, pp. 176-194.

[7] N. T. Courtois and W. Meier. "Algebraic Attacks on Stream Ciphers with Linear Feedback." In *Advances in Cryptology – EUROCRYPT 2003*, pp. 345-359.

[8] C. De Canniere, Bart Preneel. Trivium. In *New Stream Cipher Designs – The eSTREAM Finalists*, Springer-Verlag, 2008.

[9] I. Dinur, A. Shamir. Cube Attacks on Tweakable Black Box Polynomials. In *Advances in Cryptology – EUROCRYPT 2009*, pp. 278–299.

[10] I. Dinur, A. Shamir. Breaking Grain-128 with Dynamic Cube Attacks. In *Fast Software Encryption – FSE 2011*, pp. 167–187.

[11] eBAEAD: ECRYPT Benchmarking of Authenticated Ciphers. Available at `https://bench.cr.yp.to/results-caesar.html`.

[12] X. Feng, J. Liu, Z. Zhou, C. Wu and D. Feng. A Byte-Based Guess and Determine Attack on SOSEMANUK. In *Advances in Cryptology – Asiacrypt 2010*, pp. 146–157.

[13] J. Golic. "Cryptanalysis of Alleged A5 Stream Cipher". In *Advances in Cryptology – Eurocrypt'97*, pp. 239-255.

[14] P. Hawkes and G. G. Rose. Guess-and-Determine Attacks on SNOW. In *Selected Area in Cryptography – SAC 2002*, pp. 37 – 46.

[15] M. Hell, T. Johansson, A. Maximov and W. Meier. "The Grain Family of Stream Ciphers." In *New Stream Cipher Designs – The eSTREAM Finalists*, Springer-Verlag, 2008.

[16] M. E. Hellman. "A Cryptanalytic Time-Memory Trade-O", IEEE Transactions on Information Theory, Vol. IT–26, N 4, pp.401406, July 1980.

[17] T. Johansson and F. Jönsson. "Improved Fast Correlation Attacks on Stream Ciphers via Convolutional Codes." In *Advances in Cryptology – EUROCRYPT'99*, LNCS 1592, pp. 347-362, Springer-Verlag, 1999.

[18] T. Johansson and F. Jönsson. "Fast Correlation Attacks Based on Turbo Code Techniques." In *Advances in Cryptology – CRYPTO'99*, LNCS 1666, pp. 181-197, Springer-Verlag, 1999.

[19] A. Maximov and A. Biryukov. Two Trivial Attacks on Trivium. In *Selected Area in Cryptography – SAC 2002*, pp. 36 – 55.

[20] W. Meier and O. Staffelbach. "Fast Correlation Attacks on Certain Stream Ciphers." *Journal of Cryptography*, 1(3):159-176, 1989.

[21] T. Siegenthaler. "Correlation-Immunity of Nonlinear Combining Functions for Cryptographic Applications." *IEEE Transactions on Information Theory*, IT-30:776-780,1984.

[22] H. Wu and B. Preneel. "Cryptanalysis of the Stream Cipher DECIM." In *Fast Software Encryption – FSE 2006*, LNCS 4047, pp. 30-40, Springer-Verlag, 2006.

[23] H. Wu and B. Preneel. "Resynchronization Attacks on WG and LEX." In *Fast Software Encryption – FSE 2006*, LNCS 4047, pp. 422-432, Springer-Verlag, 2006.

[24] H. Wu and B. Preneel. "Differential attacks on Stream Ciphers Py, Py6 and Pypy." In *Advances in Cryptology – Eurocrypt 2007*, pp. 276-290, Springer-Verlag.

[25] H. Wu and B. Preneel. "Differential attacks on Stream Ciphers Phelix." In *Fast Software Encryption – FSE 2007*, pp. 87-100.

[26] H. Wu, T. Huang, P. H. Nguyen, H. Wang, S. Ling. "Differential Attacks against Stream Cipher ZUC." In *Advances in Cryptology – ASIACRYPT 2012*, pp. 262-277.

Appendix A

Computing the Probability of Eliminating the State Difference

In Section 3.3, we analyzed the security of message authentication. Our analysis shows that the probability to eliminate the difference in the state is 2^{-181} when there is difference in ciphertext or associated data. Table A.1 shows how the differential probability is calculated in each step (the complete differences in the state are given in Appendix B). The first column gives the step number, the following 9 columns indicates whether there is difference in a state bit $S_{i,j}$ in the ith step, and 1 indicates that there is difference. **Here the S_i is the state of the ith step after being updated using the LFSRs, and before the shift operation**. P_i is the probability at the i step that an attacker can eliminate the difference introduced in the feedback.

We illustrate Table A.1 using several examples. At the 50th step, the function $maj(S_{i,244}, S_{i,23}, S_{i,160}$ is active since there is difference in $S_{i,244}$, and the chance is half that an attacker can eliminate the difference of this function. At the 58th step, the function $maj(S_{i,235}, S_{i,61}, S_{i,193}$ is active since there is difference in $S_{i,235}$, and the chance is half that an attacker can eliminate the difference of this function. At the 63th step, the function $ch(S_{i,230}, S_{i,111}, S_{i,66})$ is active since there is difference in $S_{i,230}$, and the chance is half that an attacker can eliminate the difference of this function. At the 92th step, $maj(S_{i,235}, S_{i,61}, S_{i,193}$ and $maj(S_{i,244}, S_{i,23}, S_{i,160}$ are active since there are differences in $S_{i,235}$ and $maj(S_{i,244}$, the chance is still half that an attacker can eliminate the difference of $maj(S_{i,235}, S_{i,61}, S_{i,193} \oplus maj(S_{i,244}, S_{i,23}, S_{i,160})$.

Table A.1 Differential probability in each step of eliminating the state difference

step	$S_{i,235}$	$S_{i,61}$	$S_{i,193}$	$S_{i,230}$	$S_{i,111}$	$S_{i,66}$	$S_{i,244}$	$S_{i,23}$	$S_{i,160}$	P_i
0	0	0	0	0	0	0	0	0	0	

1	0	0	0	0	0	0	0	0	0	
2	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	0	0	0	
7	0	0	0	0	0	0	0	0	0	
8	0	0	0	0	0	0	0	0	0	
9	0	0	0	0	0	0	0	0	0	
10	0	0	0	0	0	0	0	0	0	
11	0	0	0	0	0	0	0	0	0	
12	0	0	0	0	0	0	0	0	0	
13	0	0	0	0	0	0	0	0	0	
14	0	0	0	0	0	0	0	0	0	
15	0	0	0	0	0	0	0	0	0	
16	0	0	0	0	0	0	0	0	0	
17	0	0	0	0	0	0	0	0	0	
18	0	0	0	0	0	0	0	0	0	
19	0	0	0	0	0	0	0	0	0	
20	0	0	0	0	0	0	0	0	0	
21	0	0	0	0	0	0	0	0	0	
22	0	0	0	0	0	0	0	0	0	
23	0	0	0	0	0	0	0	0	0	
24	0	0	0	0	0	0	0	0	0	
25	0	0	0	0	0	0	0	0	0	
26	0	0	0	0	0	0	0	0	0	
27	0	0	0	0	0	0	0	0	0	
28	0	0	0	0	0	0	0	0	0	
29	0	0	0	0	0	0	0	0	0	
30	0	0	0	0	0	0	0	0	0	
31	0	0	0	0	0	0	0	0	0	
32	0	0	0	0	0	0	0	0	0	
33	0	0	0	0	0	0	0	0	0	
34	0	0	0	0	0	0	0	0	0	
35	0	0	0	0	0	0	0	0	0	
36	0	0	0	0	0	0	0	0	0	
37	0	0	0	0	0	0	0	0	0	
38	0	0	0	0	0	0	0	0	0	
39	0	0	0	0	0	0	0	0	0	
40	0	0	0	0	0	0	0	0	0	
41	0	0	0	0	0	0	0	0	0	
42	0	0	0	0	0	0	0	0	0	
43	0	0	0	0	0	0	0	0	0	
44	0	0	0	0	0	0	0	0	0	
45	0	0	0	0	0	0	0	0	0	
46	0	0	0	0	0	0	0	0	0	
47	0	0	0	0	0	0	0	0	0	

48	0	0	0	0	0	0	0	0	0	
49	0	0	0	0	0	0	1	0	0	0.5
50	0	0	0	0	0	0	0	0	0	
51	0	0	0	0	0	0	0	0	0	
52	0	0	0	0	0	0	0	0	0	
53	0	0	0	0	0	0	0	0	0	
54	0	0	0	0	0	0	0	0	0	
55	0	0	0	0	0	0	0	0	0	
56	0	0	0	0	0	0	0	0	0	
57	0	0	0	0	0	0	0	0	0	
58	1	0	0	0	0	0	0	0	0	0.5
59	0	0	0	0	0	0	0	0	0	
60	0	0	0	0	0	0	0	0	0	
61	0	0	0	0	0	0	0	0	0	
62	0	0	0	0	0	0	0	0	0	
63	0	0	0	1	0	0	0	0	0	0.5
64	0	0	0	0	0	0	0	0	0	
65	0	0	0	0	0	0	0	0	0	
66	0	0	0	0	0	0	0	0	0	
67	0	0	0	0	0	0	0	0	0	
68	0	0	0	0	0	0	0	0	0	
69	0	0	0	0	0	0	0	0	0	
70	0	0	0	0	0	0	0	0	0	
71	0	0	0	0	0	0	0	0	0	
72	0	0	0	0	0	0	0	0	0	
73	0	0	0	0	0	0	0	0	0	
74	0	0	0	0	0	0	0	0	0	
75	0	0	0	0	0	0	0	0	0	
76	0	0	0	0	0	0	0	0	0	
77	0	0	0	0	0	0	0	0	0	
78	0	0	0	0	0	0	0	0	0	
79	0	0	0	0	0	0	0	0	0	
80	0	0	0	0	0	0	0	0	0	
81	0	0	0	0	0	0	0	0	0	
82	0	0	0	0	0	0	1	0	0	0.5
83	0	0	0	0	0	0	1	0	0	0.5
84	0	0	0	0	0	0	0	0	0	
85	0	0	0	0	0	0	0	0	0	
86	0	0	0	0	0	0	1	0	0	0.5
87	0	0	0	0	0	0	1	0	0	0.5
88	0	0	0	0	0	0	1	0	0	0.5
89	0	0	0	0	0	0	0	0	0	
90	0	0	0	0	0	0	1	0	0	0.5
91	1	0	0	0	0	0	0	0	0	0.5
92	1	0	0	0	0	0	1	0	0	0.5
93	0	0	0	0	0	0	0	0	0	
94	0	0	0	0	0	0	0	0	0	

95	1	0	0	0	0	0	1	0	0	0.5
96	1	0	0	1	0	0	1	0	0	0.5
97	1	0	0	0	0	0	0	0	0	0.5
98	0	0	0	0	0	0	0	0	0	
99	1	0	0	0	0	0	0	0	0	0.5
100	0	0	1	0	0	0	0	0	0	0.5
101	1	0	0	1	0	0	0	0	0	0.5
102	0	0	0	1	0	0	0	0	0	0.5
103	0	0	0	0	0	0	0	0	0	
104	1	0	0	1	0	0	0	0	0	0.5
105	1	0	0	0	0	0	0	0	0	0.5
106	0	0	0	1	0	0	0	0	0	0.5
107	0	0	0	0	0	0	0	0	0	
108	0	0	0	0	0	0	0	0	0	
109	0	0	0	1	0	0	0	0	0	0.5
110	0	0	0	1	0	0	1	0	0	0.5
111	0	0	0	0	0	0	0	0	0	
112	0	0	0	0	0	0	0	0	0	
113	0	0	0	0	0	0	0	0	0	
114	0	0	0	0	0	0	0	0	0	
115	0	0	0	0	0	0	0	0	0	
116	0	0	0	0	0	0	1	0	0	0.5
117	0	0	0	0	0	0	0	0	0	
118	0	0	0	0	0	0	0	0	0	
119	1	0	0	0	0	0	1	0	0	0.5
120	0	0	0	0	0	0	1	0	0	0.5
121	0	0	0	0	0	0	1	0	0	0.5
122	0	0	0	0	0	0	1	0	0	0.5
123	0	0	0	0	0	0	1	0	0	0.5
124	0	0	0	1	0	0	0	0	0	0.5
125	1	0	0	0	0	0	0	0	0	0.5
126	0	0	0	0	0	0	0	0	0	
127	0	0	0	0	0	0	1	0	0	0.5
128	1	0	0	0	0	0	0	0	0	0.5
129	1	0	0	0	0	0	0	0	0	0.5
130	1	0	0	0	0	0	0	0	0	0.5
131	1	0	0	0	0	0	1	0	0	0.5
132	1	0	0	0	0	0	1	0	0	0.5
133	0	0	0	0	0	0	1	0	1	0.5
134	0	0	0	1	0	0	0	0	0	0.5
135	0	0	0	0	0	0	1	0	0	0.5
136	1	0	0	0	0	0	0	0	0	0.5
137	0	0	0	1	0	0	1	0	0	0.5
138	0	0	1	0	0	0	1	0	0	0.5
139	0	0	0	1	0	0	0	0	0	0.5
140	1	0	0	1	0	0	0	0	0	0.5
141	1	0	1	0	0	0	0	0	0	0.5

142	1	0	0	0	0	0	1	0	0	0.5
143	0	0	1	0	0	0	1	0	0	0.5
144	1	0	0	1	0	0	1	0	0	0.5
145	0	0	0	1	0	0	0	0	0	0.5
146	1	0	1	0	0	0	0	0	0	0.5
147	1	0	1	0	0	0	1	0	0	0.5
148	0	0	0	0	0	0	0	0	0	
149	0	0	0	1	0	0	1	0	0	0.5
150	0	0	0	0	0	0	0	0	0	
151	1	0	0	1	0	0	1	0	0	0.5
152	1	0	0	1	0	0	0	0	0	0.5
153	1	0	0	0	0	0	1	0	0	0.5
154	0	0	0	0	0	0	1	0	0	0.5
155	0	0	0	0	0	0	0	0	0	
156	1	0	0	1	0	0	1	0	0	0.5
157	0	0	0	1	0	0	1	0	0	0.5
158	1	0	0	0	0	0	0	0	0	0.5
159	0	0	0	0	0	0	1	0	0	0.5
160	1	0	0	0	0	0	0	0	0	0.5
161	0	0	1	0	0	0	1	0	0	0.5
162	1	0	0	0	0	0	1	0	0	0.5
163	1	0	0	1	0	0	0	0	0	0.5
164	0	0	0	0	0	0	1	0	0	0.5
165	1	0	0	1	0	0	1	0	0	0.5
166	1	0	0	0	0	0	0	0	0	0.5
167	0	0	0	1	0	0	1	0	0	0.5
168	1	0	0	0	0	0	1	0	0	0.5
169	0	0	0	0	0	0	0	0	0	
170	1	0	0	1	0	0	1	0	0	0.5
171	1	0	0	1	0	0	1	0	1	0.5
172	0	0	0	0	0	0	0	0	0	
173	1	0	0	0	0	0	1	0	0	0.5
174	1	0	0	0	0	0	1	0	1	0.5
175	0	0	0	1	0	0	1	0	0	0.5
176	1	0	0	0	0	0	1	0	1	0.5
177	1	0	0	1	0	0	0	0	0	0.5
178	0	0	0	0	0	0	0	0	0	
179	1	0	1	0	0	0	1	0	1	0.5
180	1	0	0	0	0	0	0	0	1	0.5
181	0	0	1	0	0	0	1	0	0	0.5
182	1	0	0	0	1	0	0	0	0	0.5
183	1	0	0	1	0	0	1	0	0	0.5
184	1	0	0	1	0	0	1	0	0	0.5
185	1	0	1	0	0	0	1	0	0	0.5
186	0	0	0	0	0	0	0	0	0	
187	0	0	0	1	0	0	1	0	0	0.5
188	1	0	1	0	0	0	1	0	0	0.5

189	0	0	1	0	0	0	1	0	0	0.5
190	1	0	0	0	0	0	0	0	0	0.5
191	0	0	0	1	0	0	1	0	0	0.5
192	1	0	0	0	0	0	1	0	0	0.5
193	1	0	1	0	0	0	1	0	0	0.5
194	1	0	0	1	0	0	0	0	1	0.5
195	0	0	0	1	0	0	0	0	0	0.5
196	1	0	0	0	0	0	1	0	0	0.5
197	1	0	0	0	0	0	1	0	0	0.5
198	1	0	0	1	0	0	0	0	0	0.5
199	0	0	0	0	0	0	1	0	0	0.5
200	1	0	0	1	0	0	1	0	0	0.5
201	1	0	0	0	0	0	0	0	0	0.5
202	1	0	1	0	0	0	0	0	0	0.5
203	0	0	0	1	0	0	0	0	0	0.5
204	0	0	1	0	0	0	0	0	0	0.5
205	1	0	0	0	0	0	0	0	0	0.5
206	1	0	0	1	0	0	1	0	0	0.5
207	0	0	1	0	0	0	0	0	0	0.5
208	1	0	1	1	0	0	1	0	0	0.5
209	1	0	0	1	0	0	1	0	0	0.5
210	0	0	0	1	0	0	0	0	0	0.5
211	0	0	0	0	0	0	1	0	0	0.5
212	0	0	0	1	0	0	0	0	1	0.5
213	0	0	0	1	0	0	1	0	0	0.5
214	0	0	0	0	0	0	1	0	1	0.5
215	1	0	0	0	0	0	0	0	0	0.5
216	0	0	0	0	0	0	0	0	0	
217	1	0	0	1	0	0	1	0	0	0.5
218	1	0	0	1	0	0	0	0	1	0.5
219	0	0	0	0	0	0	1	0	0	0.5
220	1	0	0	0	1	0	1	0	0	0.5
221	0	0	0	0	0	0	1	0	1	0.5
222	1	0	1	1	0	0	0	0	1	0.5
223	1	0	0	1	1	0	0	0	0	0.5
224	0	0	0	1	0	0	0	0	0	0.5
225	0	0	0	0	0	0	0	0	0	
226	1	0	1	0	0	0	0	0	1	0.5
227	0	0	1	1	0	1	1	0	0	0.5
228	1	0	0	1	1	0	1	0	0	0.5
229	1	0	0	1	0	0	0	0	0	0.5
230	1	0	0	0	0	0	1	0	0	0.5
231	0	0	1	0	0	0	0	0	0	0.5
232	0	1	0	0	0	0	1	0	0	0.5
233	0	0	0	1	0	0	1	0	0	0.5
234	0	0	0	0	0	0	0	0	0	
235	0	0	0	0	0	0	1	0	1	0.5

236	1	0	0	0	0	0	0	0	0	0.5
237	1	0	0	0	0	0	0	0	1	0.5
238	0	0	0	0	0	0	0	0	0	
239	1	0	0	0	0	0	0	0	0	0.5
240	0	0	0	0	0	0	1	0	1	0.5
241	1	0	0	1	0	0	1	0	1	0.5
242	1	0	0	0	0	0	0	0	0	0.5
243	0	0	0	0	1	0	1	0	0	0.5
244	1	0	0	0	0	0	1	0	0	0.5
245	0	0	1	1	0	0	1	0	0	0.5
246	0	0	0	1	0	0	1	0	0	0.5
247	0	0	0	1	0	0	0	0	0	0.5
248	0	0	0	0	0	0	1	0	0	0.5
249	1	0	1	0	0	0	1	0	0	0.5
250	1	0	1	1	0	0	1	0	0	0.5
251	0	0	0	1	0	0	1	0	0	0.5
252	1	0	0	1	0	0	1	0	0	0.5
253	1	0	0	0	0	0	1	0	0	0.5
254	1	0	1	0	0	0	0	0	0	0.5
255	1	0	0	0	0	0	0	0	1	0.5
256	0	0	0	1	0	0	1	0	0	0.5
257	1	0	0	0	0	0	0	0	0	0.5
258	1	0	0	0	0	0	0	0	0	0.5
259	1	0	0	0	0	0	0	0	1	0.5
260	1	0	0	0	0	0	0	0	1	0.5
261	1	0	0	0	1	0	1	0	0	0.5
262	1	0	0	0	0	0	0	0	0	0.5
263	0	0	0	0	0	0	0	0	0	
264	0	0	0	0	0	0	1	0	1	0.5
265	1	0	0	0	0	1	1	0	0	0.5
266	0	0	0	0	1	0	1	0	0	0.5
267	0	0	0	0	0	0	1	0	0	0.5
268	0	0	0	0	0	0	1	0	0	0.5
269	0	0	0	0	0	0	1	0	0	0.5
270	1	0	0	0	0	0	0	1	0	0.5
271	0	0	0	0	0	0	1	0	0	0.5
272	0	0	0	0	0	0	1	0	0	0.5
273	1	0	0	0	0	0	1	0	0	0.5
274	1	0	0	0	0	0	1	0	0	0.5
275	1	0	0	0	0	0	1	0	0	0.5
276	1	0	0	0	0	0	1	0	0	0.5
277	1	0	0	0	0	0	0	0	0	0.5
278	1	0	0	0	0	0	0	0	1	0.5
279	0	0	0	0	0	0	1	0	0	0.5
280	1	0	0	0	0	0	0	0	0	0.5
281	1	0	0	0	0	0	0	0	0	0.5
282	1	0	0	0	0	0	0	0	1	0.5

283	1	0	0	0	0	0	0	0	1	0.5
284	1	0	0	0	1	0	0	0	0	0.5
285	1	0	0	0	0	0	0	0	0	0.5
286	0	0	0	0	0	0	0	0	0	
287	0	0	0	0	0	0	0	0	1	0.5
288	1	0	0	0	0	1	0	0	0	0.5
289	0	0	0	0	1	0	0	0	0	0.5
290	0	0	0	0	0	0	0	0	0	
291	0	0	0	0	0	0	0	0	0	
292	0	0	0	0	0	0	0	0	0	
293	0	0	0	0	0	0	0	0	0	
294	0	0	0	0	0	0	0	0	0	

Appendix B

Differential Propagation in the State

We list below the differences in the state in each step when we compute the authentication security of ACORN. The state at each step is the state after applying the linear feedback shift registers (before the shift operation). The state bits are from bit 0 to bit 292. A difference is indicated using 1.

step 0:
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000
step 1:
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000001
step 2:
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000010
step 3:
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000100
step 4:

```
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000001000
step 5:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000010000
step 6:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000100000
step 7:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000001000000
step 8:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000010000000
step 9:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000100000000
step 10:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000001000000000
step 11:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000010000000000
step 12:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000100000000000
step 13:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000001000000000000
step 14:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000010000000000000
step 15:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000100000000000000
step 16:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000001000000000000000
step 17:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000
step 18:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000100000000000000000
step 19:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000001000000000000000000
step 20:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
```

```
00000000000000000000000000000000010000000000000000000
step 21:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000001000000000000000000000
step 22:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000
step 23:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000100000000000000000000000
step 24:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000100000000000000000000000
step 25:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000100000000000000000000000
step 26:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000100000000000000000000000
step 27:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000100000000000000000000000
step 28:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000100000000000000000000000
step 29:
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000001000000000000000000000000000
step 30:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000010000000000000000000000000000
step 31:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000010000000000000000000000000000
step 32:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000010000000000000000000000000000
step 33:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000010000000000000000000000000000
step 34:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000010000000000000000000000000001
step 35:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000010000000000000000000000000011
step 36:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000010000000000000000000000000110
step 37:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
```

```
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000001100
step 38:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000011001
step 39:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000110011
step 40:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000001100111
step 41:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000001000000000000000000000000000000011001110
step 42:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000100000000000000000000000000000000110011101
step 43:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000001000000000000000000000000000000001100111010
step 44:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000010000000000000000000000000000011001110101
step 45:
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000
```

```
00000000100000000000000000000000000000000110011101010
step 46:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000001000000000000000000000000000000001100111010100
step 47:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000010000000000000000000000000000000011001110101001
step 48:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000100000000000000000000000000000000110011101010011
step 49:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00001000000000000000000000000000000001100111010100110
step 50:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00010000000000000000000000000000000011001110101001100
step 51:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00100000000000000000000000000000000110011101010011000
step 52:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
01000000000000000000000000000000000110011101010011000
step 53:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
10000000000000000000000000000000000110011101010011000
step 54:
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000001
00000000000000000000000000000000011001110101001100000
step 55:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000010
00000000000000000000000000000000011001110101001100000001
step 56:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000100
00000000000000000000000000000000011001110101001100000010
step 57:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000001000
00000000000000000000000000000000011001110101001100000100
step 58:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000010000
00000000000000000000000000000000011001110101001100000000000
step 59:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000100000
00000000000000000000000000000000011001110101001100000000000
step 60:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000001000000
00000000000000000000000000000000011001110101001100000000001
step 61:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000010000000
00000000000000000000000000000000011001110101001100000000010
step 62:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000100000000
00000000000000000000000110011101010011000000000000101
step 63:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000001000000000
00000000000000000000000110011101010011000000000000010
step 64:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000010000000000
00000000000000000000000110011101010011000000000000100
step 65:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000100000000000
00000000000000000000000110011101010011000000000000001000
step 66:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000001000000000000
00000000000000000000000110011101010011000000000000010000
step 67:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000001000000000000
00000000000000000000000110011101010011000000000000100000
step 68:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000001000000000000
00000000000000000000000110011101010011000000000000001000001
step 69:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000001000000000000
00000000000000000000000110011101010011000000000000010000010
step 70:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000001000000000000
```

```
00000000000000001100111010100110000000000000100000100
step 71:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000000011001110101001100000000000001000001001
step 72:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000000110011101010011000000000000010000010011
step 73:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000001100111010100110000000000000100000100111
step 74:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000011001110101001100000000000001000001001111
step 75:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000110011101010011000000000000010000010011111
step 76:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000011001110101001100000000000001000001001111110
step 77:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000110011101010011000000000000010000010011111100
step 78:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000110011101010011000000000000010000010011111000
step 79:
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000000000
00000001100111010100110000000000000100000100111110001
step 80:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000000000
000000110011101010011000000000000010000010011111100010
step 81:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000000000
000001100111010100110000000000000100000100111110001000
step 82:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000000000
0000110011101010011000000000000010000010011111100010000
step 83:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000000000
000110011101010011000000000000010000010011111100010001
step 84:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000000000
001100111010100110000000000000100000100111110001000011
step 85:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000000000
011001110101001100000000000001000001001111100010001111
step 86:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000000000
110011101010011000000000000010000010011111001000111 0
step 87:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000000000001
10011101010011000000000000010000010011111000100011101
step 88:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000000000011
00111010100110000000000000100000100111110001000111011
step 89:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000000000110
01110101001100000000000001000001001111100010001110110
step 90:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000000001100
11101010011000000000000010000010011111000100011101101
step 91:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000000011001
11010100110000000000000100000100111110001000111010010
step 92:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000000110011
10101001100000000000001000001001111100010001110101101
step 93:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000001100111
01010011000000000000010000010011111000100011101011010
step 94:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000011001110
10100110000000000000010000010011111000100011101011010101
step 95:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000100000000000000000000000000000110011101
```

```
01001100000000000001000001001111100010001110101100011
step 96:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000001100111010
10011000000000000100000100111110001000111010110001l0
step 97:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000010001110101
00110000000000001000001001111100010001110101100011101
step 98:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000100011101010
01100000000000001000001001111100010001110101100011010
step 99:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000001000111010100
11000000000000010000100111110001000111010110001111100
step 100:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000001000000000000000000000000000000010000110101001
10000000000001000001001111100010001110101100011100000
step 101:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000010000000000000000000000000000001000011101010011
00000000000001000001001111100010001110101100011100001
step 102:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000100000000000000000000000000000001000011010100110
00000000000010000010011111000100011101011000111001011
step 103:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000001000000000000000000000000000000010000110101001100
00000000000100000100111110001000111010110001110010110
step 104:
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000010000000000000000000000000000010000110101001100
00000000010000010011111000100011101011000111001011000
step 105:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000100000000000000000000000000001000011010010011000
00000000100000100111110001000111010110001110010101010
step 106:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000001000000000000000000000000000010000110101001100000
00000000100000100111110001000111010110001110010101010
step 107:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000100000000000000000000000000000010000110101001100000
00000001000001001111100010001110101100011100101010101
step 108:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000100000000000000000000000000000010000110101001100000
00000010000010011111000100011101011000111001010101011
step 109:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00001000000000000000000000000000000010000110101001100000
00000100000100111110001000111010110001110010101011111
step 110:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00010000000000000000000000000000000010000110101001100000
00001000001001111100010001110101100011100101010110110
step 111:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00100000000000000000000000000000000010000110101001100000
00010000010011111000100011101011000111001010101101101
step 112:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
```

00000000000000000000000000000000000000000000000000000000000000000
01000000000000000000000000000000010000110101001100000000000
00100000100111110001000111010110001110010101011011010
step 113:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
10000000000000000000000000000000010000110101001100000000000
01000001001111100010001110101100011100101010110110101
step 114:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000001
00000000000000000000000000000000010000110101001100000000000
10000010011111000100011101011000111001010101101101011
step 115:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000010
00000000000000000000000000000000010000110101001100000000000001
00000100111110001000111010110001110010101011011010110
step 116:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000100
00000000000000000000000000000000010000110101001100000000000010
00001001111100010001110101100011100101010110110101100
step 117:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000001000
00000000000000000000000000000000010000110101001100000000000100
00010011111000100011101011000111001010101101101011001
step 118:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000010000
00000000000000000000000000000000010000110101001100000000000001000
00100111110001000111010110001110010101011011010110010
step 119:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000100000
00000000000000000000000000000000010000110101001100000000000010000
01001111100010001110101100011100101010110110101101101
step 120:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000001000000
00000000000000000000000000000000010000110101001100000000000100000

```
10011111000100011101011000111001010101101101011011011
step 121:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000010000000
00000000000000000000000001000011010100110000000000000001000001
00111110001000111010110001110010101011011010110110111
step 122:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000100000000
00000000000000000000000010000110101001100000000000000010000010
01111100010001110101100011100101010110110101101101110
step 123:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000001000000000
00000000000000000000000010000110101001100000000000000100000100
11111000100011101011000111001010101101101011011011101
step 124:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000010000000000
00000000000000000000000100001101010011000000000000001000001001
11110001000111010110001110010101011011010110110010
step 125:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000100000000000
00000000000000000000010000110101001100000000000000010000010011
11100010001110101100011100101010110110101101101100
step 126:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000100000000000
00000000000000000000100001101010011000000000000001000001001111
11000100011101011000111001010101101101011011011000
step 127:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000100000000000
00000000000000000001000011010100110000000000000010000010011111
10001000111010110001110010101011011010110110110001
step 128:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000100000000000
00000000000000000100001101010011000000000000010000010011111
00010001110101100011100101010110110101101101101010
step 129:
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000010000000000000000
00000000000000000100001101010011000000000000010000010011111 0
00100011101011000111001010101101101011011011011101
step 130:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000010000000000000000000
00000000000000010000110101001100000000000001000000001111100
01000111010110001110010101011011010110110110111011
step 131:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000000100001101010011000000000000010000000011111000
10001110101100011100101010110110101101101101111110
step 132:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000000100001101010011000000000000010000000111110001
00011101011000111001010101101101011011011011110101
step 133:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000000000110101001100000000000001000000000111100010
00111010110001110010101011011010110110110111100011
step 134:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000000000110101001100000000000001000000000111100010 0
01110101100011100101010110110101101101101111001111
step 135:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000000000110101001100000000000001000000000101100010 00
11101011000111001010101101101011011011011110010111
step 136:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000100000000000000000000
00000000000000000110101001100000000000001000000000100100010001
11010110001110010101011011010110110110111100101111
step 137:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
```

```
00000000000000000000000000000000000010000000000000000000000
00000000000000110101001100000000000001000000000100100010011
10101100011100101010110110110110110110111001010110
step 138:
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000
00000000000011010100110000000000001000000000100100010001111
01011000111001010101101101101101101101110010101100
step 139:
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000
00000000000010010100110000000000001000000000100101010001110
10110001110010101011011010110110110111100101011000
step 140:
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000
00000000000100101001100000000000001000000000100101110001110 1
01100011100101010110110101101101101101111001010111001
step 141:
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000
00000000000100101001100000000000001000000000100101100001110 10
11000111001010101101101101101101101101111001010111010
step 142:
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000
00000000010010100110000000000000100000000010010110000111010 1
10001110010101011011010110110110110111100101011101101
step 143:
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000
00000000100101001100000000000001000000000100101100001110101 1
00011100101010110110101101101101101111001010111011011
step 144:
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000
00000001001010011000000000000010000000010010110001111010110
00111001010101101101011011011011011110010101110111111
step 145:
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000010000000000000000000000
00000010010100110000000000000100000000100101100011110101100
```

```
01110010101011011010110110110111100101011101110111
step 146:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000001000000000000000000000000000000000000
00000100101001100000000000001000000000100101100011010101100 0
11100101010110110101101101101101111001010111011101111
step 147:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000001000000000000000000000000000000000000
00001001010011000000000000010000000010010110001100010110001
11001010101101101011011011011011110010101110111011110
step 148:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000001000000000000000000000000000000000000
00010010100110000000000000100000000100101100011000101100011
10010101011011010110110110111100101011101110111101
step 149:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000001000000000000000000000000000000000000
00100101001100000000000001000000001001011000110001011000111
00101010111011010110110110111100101011101110111011
step 150:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000001000000000000000000000000000000000000
01001010011000000000000010000000010010110001100010110001110
01010101101101011011011011011110010101110111011100111
step 151:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000001000000000000000000000000000000000000
10010100110000000000000100000000100101100011000101100011100
10101011011010110110110110111100101011101110111001111
step 152:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000001000000000000000000000000000000000001
00101001100000000000001000000001001011000110001011000111001
01010110110101101101101101111001010111011101110011111
step 153:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000001000000000000000000000000000000000010
01010011000000000000010000000010010110001100010110001110010
10101101101101101101101101111001010111011101111001101 10
step 154:
```

00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000001000000000000000000000000000000000000000000100
10100110000000000000100000000100101100011000101100011100101
01011011010110110110111100101011101110111001101101
step 155:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000100000000000000000000000000000000000000001001
01001100000000000001000000001001011000110001011000111001010
10110110101101101101111001010111011101110011011010
step 156:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000000000010010
10011000000000000100000000100101100011000101100011100101
01101101011011011011110010101110111011100110110100
step 157:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000000000100101
00110000000000001000000001001011000110001011000111001010
11011010110110110111100101011101110111001101100001
step 158:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000000001001010
01100000000000001000000001001011000110001011000110001010101
10110101101101101111001010111011101110011011000010
step 159:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000000100000000000000000000000000000000000010010100
11000000000000010000000010010110001100010110001100010101011
01101011011011011110010101110111011100110110000101
step 160:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000001000000000000000000000000000000000000100101001
10000000000000100000000100101100011000101100011000101010110
11010110110110111100101011101110111001101100000011
step 161:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00000000000010000000000000000000000000000000000001001010011
00000000000001000000000100101100011000101100011000010101101
10101101101101101111001010111011101110011011000001111
step 162:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000

```
00000000000010000000000000000000000000000000000010010100110
00000000000010000000001001011000110001011000110000101011011
01011011011011011110010101110111011100110110000010110
step 163:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000001000000000000000000000000000000000000100101001100
00000000000100000000010010110001100010110001100000101011011 0
10110110110110111100101011101110111001101100000101100
step 164:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000010000000000000000000000000000000000010010100110000
00000000001000000000100101100011000101100011000001010110110 1
01101101101101111001010111011101110011011000001011001
step 165:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000100000000000000000000000000000000000100101001100000
00000000010000000001001011000110001011000110000010101101101 0
11011011011011110010101110111011100110110000010110011
step 166:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000001000000000000000000000000000000000000100101001100000
00000000100000000010010110001100010110001100000101011011010 1
10110110110111100101011101110111001101100000101101111
step 167:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000010000000000000000000000000000000000001001010011000000
00000001000000000100101100011000101100011000001010110110101 1
01101101101111001010111011101110011011000001011010110
step 168:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000100000000000000000000000000000000000010010100110000000
00000010000000001001011000110001011000110000010101001101011 0
11011011011110010101110111011100110110000010110101100
step 169:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00001000000000000000000000000000000000000010010100110000000
00000100000000010010110001100010110001100000101010011010110 1
10110110111100101011101110111001101100000101101011001
step 170:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00010000000000000000000000000000000000000010010100110000000
00001000000000100101100011000101100011000001010100110101101 1
```

```
01101101111001010111011101110011011000001011010110010
step 171:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
00100000000000000000000000000000000000001001010011000000000000
00010000000000001011000110001011000110000010101001101011011O
11011011110010101110111011100110110000010110101100100
step 172:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
01000000000000000000000000000000000000001001010011000000000000
00100000000000101100011000101100011000001010100110101101101
10110111100101011101110111001101100000101101011001000
step 173:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000
10000000000000000000000000000000000000001001010011000000000000
01000000000000101100011000101100011000001010100110011011011
01101111001010111011101110011011000001011010110010001
step 174:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000001
00000000000000000000000000000000000000001001010011000000000000
10000000000000011000110001011000110000010101001100011011O110
11011110010101110111011100110110000010110101100101011
step 175:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000010
00000000000000000000000000000000000000001001010011000000000001
00000000000001100011000101100011000001010100110001101101101
10111100101011101110111001101100000101101011001011111
step 176:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000100
00000000000000000000000000000000000000001001010011000000000010
00000000000001000110001011000110000010101001100010011011011
01111001010111011101110011011000001011010110010111110
step 177:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000001000
00000000000000000000000000000000000000001001010011000000000100
00000000000000001100010110001100000101010011000101110110110
11110010101110111011100110110000010110101100101110101
step 178:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000010000
00000000000000000000000000000000000000001001010011000000000001000
00000000000000001100010110001100000101010011000101010110110
11100101011101110111001101100000101101011001011100011
step 179:
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000100000
00000000000000000000000000000000100101001100000000000000010000
00000000000001011000101100011000001010100110001010001101101 1
11001010111011101110011011000001011010110010111000111
step 180:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000001000000
00000000000000000000000000000000100101001100000000000000100000
00000000000001011000101100011000001010100110001010001101101 11
10010101110111011100110110000010110101100101110000110
step 181:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000010000000
00000000000000000000000000000000100101001100000000000001000000
00000000000010110001011000110000010101001100010100001011011 11
00101011101110111001101100000101101011001011100000100
step 182:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000100000000
00000000000000000000000000000000100100001100000000000010000000
00000000001010000101100011000001010100110001010000011011110
01010111011101110011011000001011010110010111000001001
step 183:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000001000000000
00000000000000000000000000000100100001100000000000000100000000
00000000010100001011000110000010101001100010100000111011110 0
10101110111011100110110000010110101100101110000011010
step 184:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000010000000000
00000000000000000000000000000100100001100000000000001000000000
00000000101000010110001100000101010011000101000001110111100 1
01011101110111001101100000101101011001011100000110100
step 185:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000100000000000
00000000000000000000000000000100100001100000000000010000000000
00000001010001101100011000001010100110001010000011001111001 0
10111011101110011011000001011010110010111000001101001
step 186:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000001000000000000
00000000000000000000000000000100100001000000000000010000000000
00000010100010011000110000010101001100010100000110011110010 1
01110111011100110110000010110101100101110000011010011
step 187:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000010000000000000
```

```
00000000000000000000000010010000100000000000000010000000000000
00000101000100110001100000101010011000101000001100111001010
11101110111001101100000101101011001011100000110101111
step 188:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000010000000000000000
00000000000000000000001001000010000000000000010000000000000000
00001010001001100011000001010100110001010000011001011001010 1
11011011100110110000010110101100101110000011010111 10
step 189:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000001000000000000000000
00000000000000000000100100001000000000000001000000000000000000
00010100010011000110000010101001100010100000110010010010101 1
10111011100110110000010110101100101110000011010110101
step 190:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000010000000000000000000
00000000000000000000100100001000000000000010000000000000000000
00101000100110001100000101010011000101000001100100000101011 1
01110111001101100000101101011001011100000110101101010
step 191:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000100000000000000000000
00000000000000000000100100001000000000000010000000000000000000
01010001001100011000001010100110001010000011001000101010111 0
11101110011011000001011010110010111000011010110101 0
step 192:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000100000000000000000000
00000000000000000000100100001000000000000010000000000000000000
10100010011000110000010101001100010100000110010001010101 1101
11011100110110000010110101100101110000011010110100010
step 193:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000100000000000000000000
00000000000000000010010000100000000000000100000000000000000001
01000100110001100000101010011000101000001100100010001011101 1
10111001101100000101101011001011100000110101101000100
step 194:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000100000000000000000000
00000000000000000010010000100000000000000100000000000000000010
10001001100010000001010100110001010000011001000100110111011 1
01110011011000001011010110010111000001101011010000000
step 195:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000100000000000000000000
00000000000000000010010000100000000000000100000000000000000101
00010011000100000010101001100010100000110010001001101110111 0
```

```
11100110110000010110101100101110000011010110100001001
step 196:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000000000000
00000000000000010010000100000000000000010000000000000000001010
00100110001000000101010011000101000001100100010011011101110 1
11001101100000101101011001011100000110101101000011010
step 197:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000000000000
00000000000000010010000100000000000000010000000000000000010100
01001100010000001010100110001010000011001000100110011011101 1
10011011000001011010110010111000001101011010000110100
step 198:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000000000000
00000000000000010010000100000000000000010000000000000000101000
10011000100000010101001100010100000110010001001100110111011 1
00110110000010111010110010111000001101011010000110100 1
step 199:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000000000000
00000000000001001000010000000000000001000000000000000001010001
00110001000000101010011000101000001100100010011001101110 1 10
01101100000101110101100101110000011010110100001101101 0
step 200:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000000000000
00000000000010010000100000000000000010000000000000000010100010
01100010000000101001100010100000110010001001100101111011100 1
11011000001011010110010111000001101011010000110111100
step 201:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000000000000
00000000001001000010000000000000001000000000000000001010001001
10001000000001010011000101000001100100010011001010010111001 1
01100000101101011001011100000110101101000011010111100 1
step 202:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000000000000
00000000010010000100000000000000010000000000000000001010001001 1
10001000000001010011000101000001100100010011001010010111001 1
01100000101101011001011100000110101101000011010111100 1 1
step 203:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000010000000000000000000000000
00000000100100001000000000000000100000000000000000101000100 1 1
00010000000010100110001010000011001000100110010100101110011 0
11000001011010110010111000001101011010000110111101111
step 204:
```

```
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000000000000000
00000001001000010000000000000010000000000000000001010001001100
00100000000101001100010100000110010001001100101001011100110 1
10000010110101100101110000011010110100001101111011110
step 205:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000000000000000
00000010010000100000000000000100000000000000000001010001001100
01000000001010011000101000001100100010011001010010011001101 1
00000101101011001011100000110101101000011011110111100
step 206:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000000000000000
00000100100001000000000000001000000000000000000010100010011000
10000000010100110001010000011001000100110010100100110011011 0
00001011010110010111000001101011010000110111101111001
step 207:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000000000000000
00001001000010000000000000001000000000000000000010100010011 0001
00000000101001100010100000110010001001100101001001000110 1100
00010110101100101110000011010110100001101111011111011
step 208:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000000000000000
00010010000100000000000000010000000000000000001010001001100010
00000001010011000101000001100100010011001010010010101101 1000
00101101011001011100000110101101000011011110111111110
step 209:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000000000000000
00100100001000000000000001000000000000000000010100010011000100
00000010100110001010000011001000100110010100100101111011 0000
01011010110010111000001101011010000110111101111110100
step 210:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000000000000000
01001000010000000000000010000000000000000001010001001100010000
00000101001100010100000110010001001100101001001011110110 0000
10110101100101110000011010110100001101111011111100001
step 211:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000010000000000000000000000000000000000
10010000100000000000000100000000000000000010100010011000100000
00001010011000101000001100100010011001010010010111001100000 1
01101011001011100000110101101000011011110111111001011
step 212:
00000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000100000000000000000000000000000000000000001
```

```
00100001000000000000001000000000000000001010001001100010000
00010100110000010000011001000100110010100100101110110000010
11010110010111000001101011010000110111101111110010111
step 213:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000100000000000000000000000000000000000000010
01000010000000000000100000000000000001010001001100010000000
00101001100000100000110010001001100101001001011101100000101
10101100101110000011010110100001101111011111100100111
step 214:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000001000000000000000000000000000000000000000100
10000100000000000000100000000000000001010001001100010000000
01010011000000000001100100010011001010010010111011000001011
01011001011100000110101101000011011110111111001000111
step 215:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000001000000000000000000000000000000000000001001
00001000000000000001000000000000000001010001001100010000000
10100110000000000011001000100110010100100101110110000010110
10110010111000001101011010000110111101111110010000111
step 216:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000001000000000000000000000000000000000000010010
00010000000000000010000000000000000001010001001100010000001
01001100000000000110010001001100101001001011101100000101101
01100101110000011010110100001101111011111100100001111
step 217:
00000000000000000000000000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000000000000100100
00100000000000001000000000000000001010001001100010000000010
10011000000000011001000100110010100100101110110001001011010
11001011100000110101101000011011110111111001000010111
step 218:
00000000000000000000000000000000000000000000000000000000000000
00000000000000100000000000000000000000000000000000000001001000
01000000000000010000000000000000001010001001100010000000101
00110000000000110010001001100101001001011101100011010110101
10010111000001101011010000110111101111110010000100111
step 219:
00000000000000000000000000000000000000000000000000000000000000
00000000000000100000000000000000000000000000000000000010010000
10000000000000010000000000000000001010001001100010000000101 0
01100000000000110010001001100101001001011101100011010110101 1
00101110000011010110100001101111011111100100001001111
step 220:
00000000000000000000000000000000000000000000000000000000000000
00000000000001000000000000000000000000000000000000000100100001
00000000000000010000000000000000010000100110001000000010100
11000000000000100100010011001010010010111011000110001101011 0
```

```
01011100000110101101000011011110111111001000010011111
step 221:
00000000000000000000000000000000000000000000000000000000000000000
00000000000010000000000000000000000000000000000000001001000010
00000000000001000000000000000001000001001100010000000101001
10000000000000001000100110010100100101111011000110001101011100
10111000001101011010000110111101111110010000100111111
step 222:
00000000000000000000000000000000000000000000000000000000000000000
00000000000100000000000000000000000000000000000010010000100
00000000000010000000000000000100000100110001000000001010011
00000000000001010001001100101001001011101100011000110101100 1
01110000011010110100001101111011111100100001001111111
step 223:
00000000000000000000000000000000000000000000000000000000000000000
00000000001000000000000000000000000000000000000100100001000
00000000000100000000000000001000011001100010000000010100110
00000000000010100010011001010010010111011000110001101011001 0
11100000110101101000011011110111111001000010011111110
step 224:
00000000000000000000000000000000000000000000000000000000000000000
00000000100000000000000000000000000000000000001001000010000
00000000001000000000000000100001000110001000000001010011001 00
00000000000100000100110010100100101111011000110001111011001 01
11000001101011010000110111101111110010000100111111100
step 225:
00000000000000000000000000000000000000000000000000000000000000000
00000000100000000000000000000000000000000000010010000100000
00000000010000000000000000100001000110001000000001010011000
00000000001000001001100101001001011101100011000111011001011
10000011010110100001101111011111100100001001111110001
step 226:
00000000000000000000000000000000000000000000000000000000000000000
00000001000000000000000000000000000000000000100100001000000
00000000100000000000000001000010001100010000000010100110000
00000000010001010011001010010010111011000110001110011001011 1
00000110101101000011011110111111001000010011111101010
step 227:
00000000000000000000000000000000000000000000000000000000000000000
00000010000000000000000000000000000000000001000000010000000
00000001000000000000000010000100001000100000000101001100000
00000000100011100110010100100101111011000110001110011001011 10
00011010110100001101111011111100100001001111111011100
step 228:
00000000000000000000000000000000000000000000000000000000000000000
00000100000000000000000000000000000000000010000000100000000
00000010000000000000000010000100000000100000000101001100000 0
00000001000110001100101001001011101100011000111001100101110 0
00011010110100001101111011111100100001001111111011000
step 229:
```

```
00000000000000000000000000000000000000000000000000000000000000000
00001000000000000000000000000000000000000010000001000000000
00000100000000000000001000010000000010000000010100110000000
00000010001100011001010010010111011000110001110011101011000
00110101101000011011110111111100100010011111101111000
step 230:
00000000000000000000000000000000000000000000000000000000000000000
00010000000000000000000000000000000000010000001000000000
00001000000000000000010000100000000100000000101001100000000
00000100011000110010100100101110110001100011100111010111000
01101011010000110111101111110010000100111111011111001
step 231:
00000000000000000000000000000000000000000000000000000000000000000
00100000000000000000000000000000000000100000010000000000
00010000000000000000100001000000001000000001010011000000000
00001000110001100101001001011101100011000111001110001110000
11010110100001101111011111100100001001111110111111010
step 232:
00000000000000000000000000000000000000000000000000000000000000000
01000000000000000000000000000000000000100000000000000000000
00100000000000000001000010000000000000000010100110000000000
00010001100010001010010010111011000110001110011100011000001
10101101000011011110111111100100001001111110111111010101
step 233:
00000000000000000000000000000000000000000000000000000000000000000
10000000000000000000000000000000000000100000000000000000000
01000000000000000010000100000000000000000101001100000000000
00100011000100010100100101110110001100011100111000111000011
01011010000110111101111110010000100111111011111100011
step 234:
00000000000000000000000000000000000000000000000000000000000000001
00000000000000000000000000000000000010000000000000000000000
10000000000000000100001000000000000000001010011000000000000
01000110001000101001001011101100011000111001110001010000110
10110100001101111011111100100001001111110111111001111
step 235:
00000000000000000000000000000000000000000000000000000000000000010
00000000000000000000000000000000000100000000000000000000000001
00000000000000001000010000000000000000010100110000000000000
10001100010000010010010111011000110001110011100010000001101
01101000011011110111111001000010011111101111100101 10
step 236:
00000000000000000000000000000000000000000000000000000000000000100
00000000000000000000000000000000000100000000000000000000000010
00000000000000010000100000000000000000101001100000000000001
00011000100000100100101110110001100011100111000100000011010
11010000110111101111110010000100111111011111100100101
step 237:
00000000000000000000000000000000000000000000000000000000000001000
00000000000000000000000000000000000100000000000000000000000100
```

```
00000000000000010000100000000000000000010100110000000000000010
00110001000000001001011101100011000111001110001000000011010 1
10100001101111011111100100001001111110111111001000010
step 238:
00000000000000000000000000000000000000000000000000000010000
00000000000000000000000000000000100000000000000000000001000
00000000000001000010000000000000000010100110000000000000100
01100010000000100101110110001100011100111000100000001101011
01000011011110111111001000010011111101111110010000100
step 239:
00000000000000000000000000000000000000000000000000000100000
00000000000000000000000000000001000000000000000000000010000
00000000000010000100000000000000010100110000000000001000
11000100000001001011101100011000111001110001000000011010110
10000110111101111110010000100111111011111100100000000
step 240:
00000000000000000000000000000000000000000000000000001000000
00000000000000000000000000000010000000000000000000000100000
00000000000010000100000000000000010100110000000000010001
10001000000000001011101100011000111001110001000000011010 1101
00001101111011111100100001001111110111111001000000000
step 241:
00000000000000000000000000000000000000000000000000010000000
00000000000000000000000000000100000000000000000000001000000
00000000001000010000000000000000101001100000000000100011
00010000000000101110110001100011100111000100000011010110 10
00011011110111111001000010011111101111110010000000000
step 242:
00000000000000000000000000000000000000000000000000100000000
00000000000000000000000000001000000000000000000000010000000
00000000010000100000000000000001010011000000000000001000110
00100000000001011101100011000111001110001000000010010110100
00110111101111100100001001111110111111001000000000
step 243:
00000000000000000000000000000000000000000000000001000000000
00000000000000000000000000010000000000000000000000100000000
00000000100001000000000000000010000110000000000000010001100
01000000000000111011000110001110011100010000000100101101000
01101111011111100100001001111110111111001000000000001
step 244:
00000000000000000000000000000000000000000000000010000000000
00000000000000000000000000100000000000000000000001000000000
00000001000010000000000000000010000110000000000000100011000
10000000000000111011000110001110011100010000000100001101 0000
11011110111111001000010011111101111110010000000000011
step 245:
00000000000000000000000000000000000000000000000100000000000
00000000000000000000000001000000000000000000000010000000000
00000001000010000000000000000010000110000000000001000110001
00000000000001110110001100011100111000100000001000111010000 1
```

```
10111101111110010000100111111011111100100000000000110
step 246:
00000000000000000000000000000000000000000000001000000000000
00000000000000000000000010000000000000000000000100000000000
00000010000100000000000000000010000110000000000000010001100010
00000000000010101100011000111001110001000000100011101000011
01111011111100100001001111110111111001000000000000100
step 247:
00000000000000000000000000000000000000000000010000000000000
00000000000000000000000100000000000000000000001000000000000
00000100001000000000000000010000100000000000000100011000100
00000000000100011000110001110011100010000001000111010000110
11110111111001000010011111101111100100000000000000001
step 248:
00000000000000000000000000000000000000000000100000000000000
00000000000000000000001000000000000000000000010000000000000
00001000010000000000000000100001000000000000001000110001000
00000000010001100011000111001110001000000100011101000011101
11101111110010000100111111011111100100000000000000010
step 249:
00000000000000000000000000000000000000000001000000000000000
00000000000000000000010000000000000000000001000000000000000
00010000100000000000000001000010000000000000010001100010000
00000000100011000110001110011100010000001000111000000011011
11011111100100001001111110111111001000000000000000101
step 250:
00000000000000000000000000000000000000000010000000000000000
00000000000000000000100000000000000000000010000000000000000
00100001000000000000000010000100000000000000100011000100000
00000001000110001100011100111000100000010001110011000110111
10111111001000010011111101111100100000000000000011
step 251:
00000000000000000000000000000000000000000100000000000000000
00000000000000000001000000000000000000000100000000000000000
01000010000000000000000100001000000000000010001100010000000
00000010001100011000111001110001000000100011100110011011111
01111110010000100111111011111100100000000000000110
step 252:
00000000000000000000000000000000000000000100000000000000000
00000000000000000001000000000000000000000100000000000000000
10000100000000000000000100001000000000000010001100010000000
00000100011000110001110011100010000001000111001110110111110
11111100100001001111110111111001000000000000000100
step 253:
00000000000000000000000000000000000000000100000000000000000
00000000000000000010000000000000000000000100000000000000001
00001000000000000000010000100000000000010001100010000000000
00000100011000110001110011100010000001000111001110110111101
11111001000010011111101111100100000000000000000000
step 254:
```

```
00000000000000000000000000000000000000010000000000000000000
00000000000000000100000000000000000000010000000000000000010
00010000000000000000100001000000000000001000110001000000000
00001000110001100011100111000100000010001110011100101111011
11110010000100111111011111100100000000000000000000000
step 255:
000000000000000000000000000000000000010000000000000000000
000000000000000100000000000000000000010000000000000000100
00100000000000000001000010000000000000010001100010000000000
00010001100010000111001110001000000100011100111000011110111
11100100001001111110111111001000000000000000000000
step 256:
000000000000000000000000000000000000010000000000000000000
000000000000000100000000000000000000010000000000000000100
01000000000000000001000010000000000000010001100010000000000
00100011000100001110011100010000001000111001110001111101111
11001000010011111101111100100000000000000000000000
step 257:
000000000000000000000000000000000000010000000000000000000
000000000000000100000000000000000000010000000000000000010000
10000000000000000010000100000000000000010001100010000000000
01000110001000011100111000100000010001110011100010111011111
10010000100111111011111100100000000000000000000000
step 258:
000000000000000000000000000000000000010000000000000000000
000000000000000100000000000000000000010000000000000000100001
00000000000000000100001000000000000001000110001000000000000
10001100010000111001110001000000100011100111000100110111111
00100001001111110111111001000000000000000000000001
step 259:
000000000000000000000000000000000000010000000000000000000
000000000000100000000000000000000000010000000000000000001000010
00000000000000001000010000000000000001000110001000000000001
00011000100000110011100010000001000111001110001000101111110
01000010011111101111100100000000000000000000000010
step 260:
000000000000000000000000000000000000010000000000000000000
000000000001000000000000000000000000010000000000000000010000100
00000000000000010000100000000000000001000110001000000000010
00110001000000100111000100000010001110011100010000011111100
10000100111111011111100100000000000000000000000101
step 261:
000000000000000000000000000000000000010000000000000000000
000000000010000000000000000000000000010000000000000000100001000
00000000000000100001000000000000000000110001000000000000100
01100010000000001110001000000100011100111000100000111111001
00001001111110111111001000000000000000000000000011
step 262:
000000000000000000000000000000000000010000000000000000000
000000000100000000000000000000000000010000000000000001000010000
```

```
00000000000001000010000000000000000011000100000000000001000
11000100000000011100010000000100011100111000100000011111001 0
0001001111110111111001000000000000000000000000000110
step 263:
00000000000000000000000000000010000000000000000000000000000000
00000000100000000000000000000001000000000000000000100001000000
00000000000010000100000000000000000011000100000000000010001
10001000000000111000100000001000111001110001000000011110010 0
0010011111101111110010000000000000000000000000000101
step 264:
00000000000000000000000000000010000000000000000000000000000000
00000001000000000000000000000010000000000000000010000100000 0
00000000000100001000000000000000000110001000000000000100011
00010000000001100010000000100011100111000100000000111001000
01001111110111111001000000000000000000000000000011
step 265:
00000000000000000000000000000010000000000000000000000000000000
00000010000000000000000000000010000000000000000000000010000000
00000000001000010000000000000000000100010000000000001000110
00100000000000100010000000100011100111000100000000110010000
10011111101111110010000000000000000000000000000110
step 266:
00000000000000000000000000000010000000000000000000000000000000
00000100000000000000000000000010000000000000000000000100000000
00000000010000100000000000000000000001000000000000010001100
01000000000000001000000010001110011100010000000000100100001
00111111011111100100000000000000000000000000000100
step 267:
00000000000000000000000000000010000000000000000000000000000000
00001000000000000000000000000100000000000000000000001000000000
00000000100001000000000000000000000001000000000000100011000
10000000000000001000000010001110011100010000000000001000010
01111110111111001000000000000000000000000000000000
step 268:
00000000000000000000000000000010000000000000000000000000000000
00010000000000000000000000000100000000000000000000010000000000
00000001000010000000000000000000000001000000000000100011001
00000000000000010000000100011100111000100000000000010000100
11111101111110010000000000000000000000000000000000
step 269:
00000000000000000000000000000010000000000000000000000000000000
00100000000000000000000000000100000000000000000000100000000000
00000010000100000000000000000000000010000000000001000110001 0
00000000000000100000001000111001110001000000000000100001001
11111011111100100000000000000000000000000000000000
step 270:
00000000000000000000000000000010000000000000000000000000000000
00000000000000000000000000000100000000000000000000000000000000
00000100001000000000000000000000000000000000000000100011000100
00000000000000000000010001110011100010000000000000000000010011
```

```
11110111111001000000000000000000000000000000000000001
step 271:
00000000000000000000001000000000000000000000000000000000
00000000000000000000001000000000000000000000000000000000
00001000010000000000000000000000000000000000001000110001000
00000000000000000000100011100111000100000000000000000100111
11101111110010000000000000000000000000000000000011
step 272:
00000000000000000000010000000000000000000000000000000000
00000000000000000000010000000000000000000000000000000000
00010000100000000000000000000000000000000000010001100010000
00000000000000000001000111001110001000000000000000001001111
11011111100100000000000000000000000000000000000110
step 273:
00000000000000000000100000000000000000000000000000000000
00000000000000000000100000000000000000000000000000000000
00100001000000000000000000000000000000000000100011000100000
00000000000000000010001110011100010000000000000000010011111
10111111001000000000000000000000000000000000000101
step 274:
00000000000000000001000000000000000000000000000000000000
00000000000000000001000000000000000000000000000000000000
01000010000000000000000000000000000000000001000110001000000
00000000000000000100011100111000100000000000000000100111111
01111110010000000000000000000000000000000000000011
step 275:
00000000000000000010000000000000000000000000000000000000
00000000000000000010000000000000000000000000000000000000
10000100000000000000000000000000000000000010001100010000000
00000000000000001000111001110001000000000000000001111110
11111100100000000000000000000000000000000000000110
step 276:
00000000000000000100000000000000000000000000000000000000
00000000000000000100000000000000000000000000000000000001
00001000000000000000000000000000000000000100011000100000000
00000000000000010001110011100010000000000000000011111101
11111001000000000000000000000000000000000000000101
step 277:
00000000000000001000000000000000000000000000000000000000
00000000000000001000000000000000000000000000000000000010
00010000000000000000000000000000000000001000110001000000000
00000000000000100011100111000100000000000000000111111011
11110010000000000000000000000000000000000000000010
step 278:
00000000000000010000000000000000000000000000000000000000
00000000000000010000000000000000000000000000000000000100
00100000000000000000000000000000000000001000110001000000000
00000000000000001110011100010000000000000000000111110111
11100100000000000000000000000000000000000000000100
step 279:
```

```
00000000000000100000000000000000000000000000000000000000000000000
00000000000000010000000000000000000000000000000000000000000001000
01000000000000000000000000000000000000010001100010000000000000
00000000000000001110011100010000000000000000000000000111101111
11001000000000000000000000000000000000000000000000000000
step 280:
00000000000001000000000000000000000000000000000000000000000000000
00000000000000100000000000000000000000000000000000000000010000
10000000000000000000000000000000000001000110001000000000000
00000000000000011100111000100000000000000000000000111011111
10010000000000000000000000000000000000000000000000000000
step 281:
00000000000010000000000000000000000000000000000000000000000000000
00000000000001000000000000000000000000000000000000000100001
00000000000000000000000000000000000010001100010000000000000
00000000000000111001110001000000000000000000000000110111111
00100000000000000000000000000000000000000000000000000001
step 282:
00000000000100000000000000000000000000000000000000000000000000000
00000000000010000000000000000000000000000000000000001000010
00000000000000000000000000000000000010001100010000000000000
00000000000000110011100010000000000000000000000000101111110
01000000000000000000000000000000000000000000000000000010
step 283:
00000000001000000000000000000000000000000000000000000000000000000
00000000000100000000000000000000000000000000000000010000100
00000000000000000000000000000000000010001100010000000000000
00000000000000100111000100000000000000000000000000011111100
10000000000000000000000000000000000000000000000000000101
step 284:
00000000010000000000000000000000000000000000000000000000000000000
00000000001000000000000000000000000000000000000000100001000
00000000000000000000000000000000000011000100000000000000000
00000000000000001110001000000000000000000000000000111111001
00000000000000000000000000000000000000000000000000000011
step 285:
00000000100000000000000000000000000000000000000000000000000000000
00000000010000000000000000000000000000000000000001000010000
00000000000000000000000000000000000011000100000000000000000
00000000000000011100010000000000000000000000000000111110010
00000000000000000000000000000000000000000000000000000110
step 286:
00000001000000000000000000000000000000000000000000000000000000000
00000000100000000000000000000000000000000000000010000100000
00000000000000000000000000000000000011000100000000000000000
00000000000000111000100000000000000000000000000000111100100
00000000000000000000000000000000000000000000000000000101
step 287:
00000010000000000000000000000000000000000000000000000000000000000
00000001000000000000000000000000000000000000000000100001000000
```

```
00000000000000000000000000000000000011000100000000000000000000
00000000000000110001000000000000000000000000000000000111001000
00000000000000000000000000000000000000000000000000000011
step 288:
00000100000000000000000000000000000000000000000000000000000000
00000010000000000000000000000000000000000000000000000010000000
00000000000000000000000000000000000100010000000000000000000000
00000000000000100010000000000000000000000000000000110010000000
00000000000000000000000000000000000000000000000000000110
step 289:
00001000000000000000000000000000000000000000000000000000000000
00000100000000000000000000000000000000000000000000000100000000
00000000000000000000000000000000000010000000000000000000000000
00000000000000001000000000000000000000000000000000100100000000
00000000000000000000000000000000000000000000000000000100
step 290:
00010000000000000000000000000000000000000000000000000000000000
00001000000000000000000000000000000000000000000000001000000000
00000000000000000000000000000000000010000000000000000000000000
00000000000000010000000000000000000000000000000000001000000000
00000000000000000000000000000000000000000000000000000001
step 291:
00100000000000000000000000000000000000000000000000000000000000
00010000000000000000000000000000000000000000000000010000000000
00000000000000000000000000000000000010000000000000000000000000
00000000000000010000000000000000000000000000000000010000000000
00000000000000000000000000000000000000000000000000000010
step 292:
01000000000000000000000000000000000000000000000000000000000000
00100000000000000000000000000000000000000000000000100000000000
00000000000000000000000000000000000010000000000000000000000000
00000000000000100000000000000000000000000000000000100000000000
00000000000000000000000000000000000000000000000000000100
step 293:
10000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000
step 294:
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000
```