

Codex 32

A Shamir Secret Sharing Scheme



Leon Olsson Curr

Pearlwort Snead

MIT License

Copyright © 2022 Blockstream

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Additional materials and copies of this document can be found at:

<https://secretcodex32.com/docs/index.html>

<https://github.com/roconnor-blockstream/SSS32/>

Produced in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN 978-1-7338712-2-8 (Paperback)

Cover and Vovelle Illustrations by Micaela Paez

Illuminated Letters & Inline Illustrations by M. Lutfi' As'ad

Edited & Produced by Arri Isak Beck

Principal Tables

Addition

A	C	D	E	F	G	H	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9	
A	Q	9	S	Y	5	4	2	0	T	Z	X	W	U	A	7	D	K	P	3	N	M	E	L	J	H	V	G	F	8	R	6	C
C	9	Q	4	P	3	S	0	2	W	8	R	T	E	C	M	G	N	Y	5	K	7	U	6	H	J	F	D	V	Z	X	L	A
D	S	4	Q	5	Y	9	6	L	M	J	K	7	V	D	W	A	X	3	P	R	T	F	0	Z	8	U	C	E	H	N	2	G
E	Y	P	5	Q	S	3	W	T	0	X	Z	2	C	E	6	F	J	9	4	H	L	A	M	K	N	G	V	D	R	8	7	U
F	5	3	Y	S	Q	P	7	M	L	K	J	6	G	F	2	E	Z	4	9	8	0	D	T	X	R	C	U	A	N	H	W	V
G	4	S	9	3	P	Q	L	6	7	H	N	M	F	G	T	C	R	5	Y	X	W	V	2	8	Z	E	A	U	J	K	0	D
H	2	0	6	W	7	L	Q	9	P	G	V	Y	K	H	5	8	U	T	M	E	3	N	4	C	A	X	Z	R	D	F	S	J
J	0	2	L	T	M	6	9	Q	Y	D	F	P	N	J	3	Z	E	W	7	U	5	K	S	A	C	R	8	X	G	V	4	H
K	T	W	M	0	L	7	P	Y	Q	F	D	9	H	K	4	X	A	2	6	C	S	J	5	E	U	8	R	Z	V	G	3	N
L	Z	8	J	X	K	H	G	D	F	Q	Y	V	7	L	U	0	5	R	N	3	E	M	A	S	4	W	2	T	9	P	C	6
M	X	R	K	Z	J	N	V	F	D	Y	Q	G	6	M	C	T	S	8	H	4	A	L	E	5	3	2	W	0	P	9	U	7
N	W	T	7	2	6	M	Y	P	9	V	G	Q	J	N	S	R	C	0	L	A	4	H	3	U	E	Z	X	8	F	D	5	K
P	U	E	V	C	G	F	K	N	H	7	6	J	Q	P	Z	A	D	0	8	9	R	W	T	S	5	4	M	L	X	Y		
Q	A	C	D	E	F	G	H	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
R	7	M	W	6	2	T	5	3	4	U	C	S	Z	R	Q	N	G	L	O	D	9	8	P	V	F	J	K	H	E	A	Y	X
S	D	G	A	F	E	C	R	U	E	A	S	C	2	I	G	M	V	U	7	K	5	J	L	6	P	9	Y	2	W	H	4	
T	K	N	X	J	Z	R	U	E	A	S	C	2	I	G	M	V	U	7	K	5	J	L	6	P	9	Y	2	W	H	4		
U	P	Y	3	9	4	5	T	W	2	5	R	0	A	U	L	V	H	Q	S	J	6	C	7	N	K	D	F	G	X	Z	M	E
V	3	5	P	4	9	Y	M	7	6	N	H	L	D	V	0	U	8	S	Q	Z	2	G	W	R	X	A	E	C	K	J	T	F
W	N	K	R	H	8	X	E	U	C	3	4	A	0	W	D	7	9	J	Z	Q	G	2	V	P	Y	L	M	6	5	S	F	T
X	M	7	T	L	O	W	3	5	S	E	A	4	8	X	9	K	D	6	2	G	Q	Z	Y	F	V	H	N	J	U	C	P	R
Y	E	U	F	A	D	V	N	K	J	M	L	H	9	Y	8	5	0	C	G	2	Z	Q	X	T	W	4	3	7	6	R	P	
Z	L	6	0	M	T	2	4	S	5	A	E	3	R	Z	P	J	F	7	W	V	Y	X	Q	D	G	N	H	K	C	U	9	8
0	J	H	Z	K	X	8	C	A	E	S	U	W	0	V	L	N	R	P	F	T	D	Q	9	7	6	M	4	3	G	2		
2	H	J	8	N	R	Z	A	C	U	4	3	E	T	2	F	6	P	K	X	Y	V	W	G	9	Q	M	L	7	S	5	D	0
3	V	F	U	G	C	E	X	R	8	W	2	Z	S	3	J	P	6	D	A	L	H	4	N	7	M	Q	Y	9	T	0	K	5
4	G	D	C	V	U	A	Z	8	R	2	W	X	5	4	K	9	7	F	E	M	N	3	H	6	L	Y	Q	P	0	T	J	S
5	F	V	E	D	A	U	R	X	Z	T	0	8	4	5	H	Y	L	G	C	6	J	S	K	M	7	9	P	Q	W	2	N	3
6	8	Z	H	R	N	J	D	G	V	9	P	F	M	6	E	2	3	X	K	5	U	7	C	4	S	T	O	W	Q	Y	A	L
7	R	X	N	8	H	K	F	V	G	P	9	D	L	7	A	W	4	Z	J	S	C	6	U	3	5	0	T	2	Y	Q	E	M
8	6	L	2	7	W	0	S	4	3	C	U	5	X	8	Y	H	V	M	T	F	P	R	9	G	D	K	J	N	A	E	Q	Z
9	C	A	G	U	V	D	J	H	N	6	7	K	Y	9	X	4	W	E	F	T	R	P	8	2	0	5	S	3	L	M	Z	Q

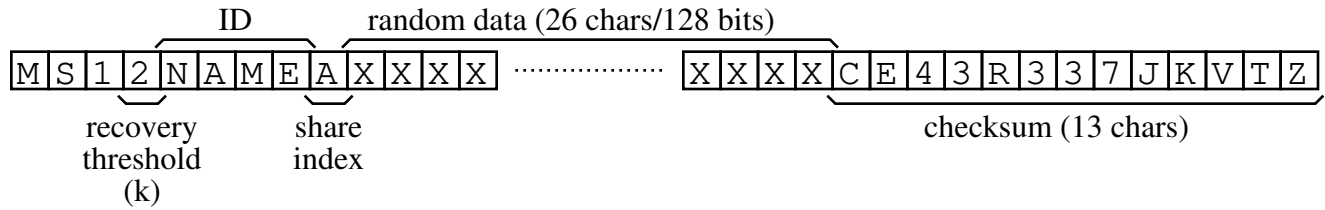
Translation

Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Λ	Μ	Ξ	Π	Ρ	Σ	Φ	Ψ	Ω	@	#	%	¢	¥	€	¤	¢	†	‡	§	¶	♦	♥		
A	A	N	W	0	J	P	7	R	D	S	3	V	Z	L	4	G	X	M	6	8	F	5	T	K	C	9	Y	E	H	2		
C	C	E	P	M	R	Z	6	L	8	X	7	Y	U	A	9	H	0	W	K	V	5	4	D	G	S	F	N	T	2	J		
D	D	6	H	A	S	8	2	N	7	F	Y	W	R	5	E	0	Z	4	C	J	L	G	9	U	3	X	T	P	V	M	K	
E	E	M	Z	L	X	Y	A	H	W	V	4	G	3	N	2	8	7	U	9	C	P	R	6	S	F	T	J	O	K	5	D	
F	F	J	M	D	Y	L	K	6	N	G	P	H	7	9	V	A	5	0	X	S	E	Z	T	8	W	4	U	2	R	C	3	
G	G	S	C	F	P	E	3	J	6	Z	2	M	N	T	R	D	9	A	4	Y	V	5	U	L	H	0	8	K	7	X	W	
H	H	8	S	W	E	F	7	U	T	M	V	J	9	4	Z	3	X	K	P	L	G	C	O	D	6	2	A	R	5	Y	N	
J	J	D	L	6	G	H	9	A	0	S	Z	8	4	2	C	N	P	7	V	F	M	Y	K	W	U	R	3	5	X	E	T	
K	K	9	N	2	U	0	E	5	Z	3	8	7	G	M	D	P	H	Y	J	T	A	W	C	4	R	S	X	L	F	6	V	
L	L	H	G	8	C	S	0	W	3	E	X	F	K	7	P	U	R	T	5	M	Y	V	N	J	D	9	6	4	2	Z	A	
M	M	L	Y	H	V	G	N	8	U	C	R	S	T	0	5	W	4	3	2	E	Z	X	A	F	J	K	D	7	9	P	6	
N	N	0	U	7	D	3	Z	4	X	6	F	T	C	Y	H	R	S	V	L	A	W	J	P	K	9	E	2	G	M	8	5	
P	P	Z	R	Y	9	X	8	G	F	2	T	V	D	W	0	S	3	J	N	5	4	K	H	C	E	6	M	U	A	7	L	
Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q	Q
R	R	X	9	V	0	2	F	C	M	7	A	5	H	J	3	E	6	L	U	4	K	N	S	P	Z	8	Y	D	W	T	G	
S	S	F	E	J	Z	M	T	D	A	Y	5	L	0	K	X	6	2	N	R	G	C	P	3	H	8	7	W	9	4	V	U	
T	T	K	A	9	W	N	C	2	P	U	H	0	Y	E	J	5	L	Z	F	3	6	8	V	7	4	G	R	M	S	D	X	
U	U	3	D	T	H	6	X	K	2	8	M	A	P	V	S	9	E	5	G	W	J	L	R	N	O	Z	7	C	Y	F	4	
V	V	C	5	E	4	P	D	M	H	R	0	Z	W	6	K	L	N	8	T	X	2	7	J	Y	G	U	S	A	3	9	F	
W	W	U	J	3	L	D	R	T	9	H	E	6	5	X	G	K	C	2	Y	8	F	M	4	A	N	P	0	V	Z	S	7	
X	X	V	2	C	7	5	J	E	L	4	N	P	8	D	T	M	A	H	3	R	9	0	F	Z	Y	W	G	6	U	K	S	
Y	Y	G	V	S	5	C	U	F	D	P	9	E	A	3	4	J	K	6	7	Z	X	2	W	M	L	N	H	T	0	R	8	
Z	Z	Y	X	G	2	V	S	J	5	K	C	6	U	7	F	T	D	O	P	R	9	8	E	M	A	L	3	N	4	H		
0	0	7	3	4	6	T	Y	R	V	A	J	K	E	G	8	X	F	C	H	N	U	D	Z	9	2	M	5	S	L	W	P	
2	2	5	7	P	T	4	L	Z	G	K	U	R	F	H	A	Y	W	S	6	9	0	3	M	X	V	J	C	8	D	N	E	
3	3	T	6	K	8	A	V	9	5	W	L	N	Z	C	F	2	M	P	S	U	D	H	X	0	7	Y	4	E	G	J	R	
4	4	R	K	X	N	9	S	V	E	0	6	2	L	F	U	C	D	M	W	7	T	A	G	5	P	H	Z	J	8	3	Y	
5	5	P	4	Z	K	R	H	Y	S	9	3	X	J	8	N	G	U	F	A	2	7	T	L	V	C	D	E	W	6	M	9	
6	6	A	8	N	E	W	5	0	4	J	G	U	X	P	M	7	Y	R	E	D	H	S	2	3	T	V	K	Z	C	L	0	
7	7	7	4	T	R	A	K	G	X	C	N	D	9	M	S	W	V	J	E	8	0	3	6	Y	2	5	L	P	F	H	U	Z
8	8	W	F	U	M	J	4	3	K	L	C	D	2	R	Y	T	V	9	Z	H	S	E	7	6	A	5	N	X	P	G	0	
9	9	2	0	5	3	7	M	P	Y	T	W	4	S	L	6	Z	8	G	D	K	N	U	E	R	X	F	V	H	J	A	C	

Recovery

A	C	D	E	F	G	H	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
A	x	#	@	!	Q	1	¥	♦	€	Σ	ε	β	μ	Ξ	Φ	ρ	¢	Δ	†	♥	θ										

Share Data Format



Bech32 to Binary Conversion

A: 11101	K: 10110	T: 01011	2: 01010
C: 11000	L: 11111	U: 11100	3: 10001
D: 01101	M: 11011	V: 01100	4: 10101
E: 11001	N: 10011	W: 01110	5: 10100
F: 01001	P: 00001	X: 00110	6: 11010
G: 01000	Q: 00000	Y: 00100	7: 11110
H: 10111	R: 00011	Z: 00010	8: 00111
J: 10010	S: 10000	0: 01111	9: 00101

Binary to Bech32 Conversion

00000: Q	01000: G	10000: S	11000: C
00001: P	01001: F	10001: 3	11001: E
00010: Z	01010: 2	10010: J	11010: 6
00011: R	01011: T	10011: N	11011: M
00100: Y	01100: V	10100: 5	11100: U
00101: 9	01101: D	10101: 4	11101: A
00110: X	01110: W	10110: K	11110: 7
00111: 8	01111: 0	10111: H	11111: L

Symbols

ℵ Aleph	α Alpha	β Beta	Γ Gamma
Δ Delta	ε Epsilon	η Eta	Θ Theta
Λ Lambda	μ Mu	Ξ Xi	Π Pi
ρ Rho	Σ Sigma	Φ Phi	Ψ Psi
Ω Omega	@ At	# Hash	% Percent
¢ Cent	¥ Yen	€ Euro	♁ Scarab
⊕ Earth	† Dagger	‡ Double-dagger	§ Section
¶ Paragraph	◆ Diamond	♥ Heart	

Table of Contents

Part I: High-Level Introduction	1
I.1. Shamir Secret Sharing Scheme	1
I.2. codex32	2
I.3. Computers and Trust	2
I.4. Checksumming and Error Correction	3
I.5. Seeds and Seed Words	4
I.6. Bech32 and Alternative Alphabets	4
Part II: codex32 Components	5
II.1. Share Data	5
II.2. Paper Computers & Vovelles	5
Part III: Process Instructions / Cheatsheet	7
III.1 Generate a New Secret	8
III.1.A Create First Share	8
III.1.B Create Derived Shares	9
III.2 Recover a Secret	9
III.2.A Recovery by Table Lookup	10
III.2.B Recovery by Vovelle	10
Worksheets	11
Dice De-biasing Worksheet	11
Checksum Worksheet (Generation Instructions)	13
Checksum Worksheet (Verification Instructions)	14
Translation Worksheet	18
Additional Modules	21
Module 0: Vovelles	21
Module 1: Share Booklet	29
Module 2: Extra Share Generation Tables	38

Part I: High-Level Introduction



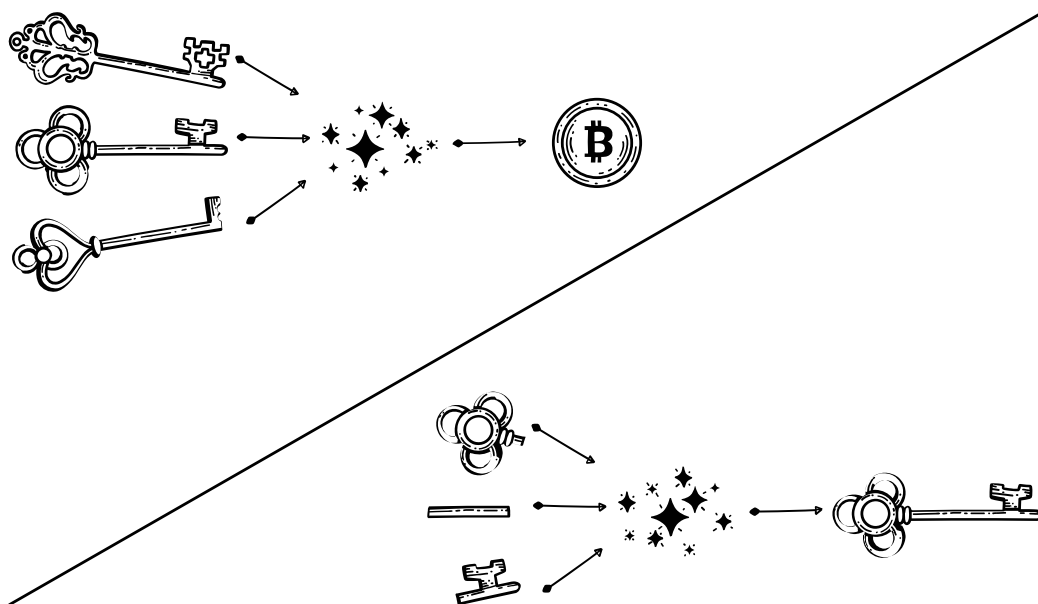
ryptography is the art of hiding information. In particular, **Shamir Secret Sharing Scheme (SSSS)** is used to hide secrets in a distributed way. **codex32** describes a way for users, assisted by paper computers in the form of slide charts and circular slide rules (i.e., volvelles), to perform checksums and SSSS on Bitcoin secrets. If you are ready to begin using this process, jump to page 7 to follow the cheatsheet. For a more in-depth primer, continue reading the high-level introduction below.

I.1. Shamir Secret Sharing Scheme



he **Shamir Secret Sharing Scheme (SSSS)** splits a **secret s** into n **shares**, any k of which can be used to reconstruct the original secret. Shares can be kept in separate places. The shares can later be used to reconstruct the original secret. It is important to emphasize that SSSS is a mechanism for storing backups, not a mechanism for enforcing a signing policy, as is done with multisig.

Multisig



Shamir Secret Sharing Scheme

With SSSS, n is typically five or more, depending on your desire for redundancy, while k is two or three, reflecting your fear of individual shares being compromised. There is an inherent trade-off between the availability of a secret and its risk of theft. If you make many copies of your seed words, one of them may fall into the wrong hands. However, if you make too few, they could become lost, destroyed, or misplaced. The consequence in either case is a complete and total loss of funds. By using shares rather than complete copies of our seed, we can make this tradeoff in a more flexible way.

Shamir Secret Sharing Scheme was first proposed in the 1970s, and has historically required the use of computers to generate secrets and shares. Instead, this codex outlines a novel method of secret sharing that can be done entirely on paper.

I.2. codex32



Using volvelles this codex illustrates a method for Shamir Secret Sharing Bitcoin secrets. This document also defines an error-correcting code and complete scheme for generating, checksumming, splitting, and reconstructing secret data.

The function of this codex is to provide a paper-based means to:

- Securely generate random data from potentially biased dice rolls or coin flips to create shares.
- Split a secret into up to 31 shares, of which some number of them are needed to reconstruct the secret.
- Recombine your shares into your original secret.
- Compute and verify powerful checksums as part of each share.

This scheme does not support passphrases or key hardening, so security rests solely on the strength of your randomness. Because of this, it is extremely important to generate truly random numbers. This document provides a dice de-biasing worksheet to generate random values from dice rolls by hand. If you prefer the added security of passphrase-based key hardening, you should instead use SLIP39. SLIP39 is a non-paper-based Shamir Secret Sharing Scheme for Bitcoin secrets. Which does, however, require the use of electronic computers.

I.3. Computers and Trust



It is impossible to sign a Bitcoin transaction without giving an electronic computer access to secret key data, which puts the user in an unfortunate position. If misused or badly generated, private key data can be used to steal all of your coins. To make matters worse, there is no way to know how exactly an electronic computer is interacting with your keys.

General-purpose computers are so complex and exposed to an adversarial environment (i.e., in the form of Internet connections, arbitrary programs, and human beings). The standard advice is to never expose your key material to such machines. Instead, you should provide your keys only to hardware wallets, which interact with general-purpose computers narrowly, through an interface that does not expose your secret key data. But this introduces additional questions: how can the hardware wallet be sure that it's communicating with the correct user, and under correct circumstances? *Ultimately, there is no hardware wallet that a wizard can fully trust.*

Even hardware wallets are opaque and imperfect:

- If tasked with generating random data, it may do so insecurely.
- It may have bugs that cause key leakage, either now or as a consequence of some future software update.
- Key material stored in physical form can be extracted by an attacker with physical access, even if the wallet has "deleted" it.
- It may expose secret data through side channels, such as the electromagnetic waves emitted by processor activity, or by the varying power draw from a USB hub.

These risks have varying degrees of plausibility, but for a Bitcoin secret, which may exceed any one person's lifetime, even "trivial" risks add up to become very serious.

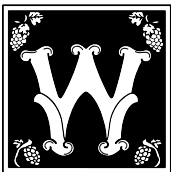
The good news is that unlike electronic computers, paper cannot remember or leak secrets! When handled correctly and disposed of securely, and this can be done without special skills, equipment, or magic.

Some limitations of SSSS include:

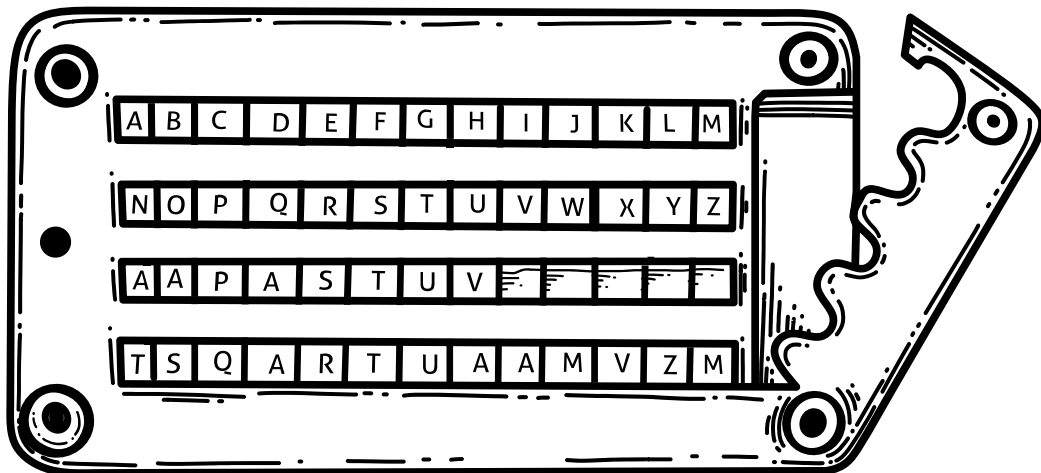
- SSSS requires that the complete secret be reconstructed in a single place before it can be used. If the shares created were initially distributed among different individuals in a group, the party that brings together the shares to recreate the secret has the opportunity to spend funds, even if their single-party authority was not intended.
- SSSS requires the generation of additional random data beyond the original secret, which must be generated securely.
- If any share is corrupted, the reconstructed secret will be wrong, and it's impossible to determine which share was responsible, or how many.

We have addressed the latter issue through the clever use of error-correcting codes, inspired by SLIP39. However, not much can be done about the fact that SSSS involves a single point of failure at the time that the secret key material is used. This is why this scheme is only to be used for backups, and not for enforcing a signing policy.

I.4. Checksumming and Error Correction



When you copy or transfer keys, and especially when you are conducting hand computations, it is possible that errors may arise. A **checksum** is a technique used to determine the authenticity of received data, to detect whether there was an error in transmission, storage, or copying. Errors might also crop up during long-term storage (e.g., if a paper backup suffers water damage, or a cryptosteels' tiles are damaged making some of the letters illegible).



This scheme does not support BIP39 mnemonic codes, which is currently the most popular way of storing private key backups. Users should note that the BIP39 checksum is less than one word long, and may fail to detect even a single incorrect word. Its primary effects are to cause your key data to be an awkward length, and to prevent you from verifying your data's integrity by hand.

In contrast, the SLIP39 Shamir Secret Sharing Scheme can detect up to three errors and correct up to one error 100% of the time. Additionally, it will fail to detect other random errors with extremely low probability. However, the SLIP39 checksum is also quite difficult to compute or verify by hand.

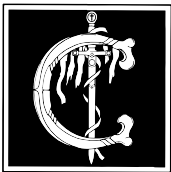
In the Codex32 book, we introduce the **codex32** checksum, which can detect up to eight errors and correct up to four. codex32 has an even higher probability than SLIP39 of successfully detecting random errors. And most importantly, codex32 checksums can be computed and verified entirely by hand.

I.5. Seeds and Seed Words



IP32 is a protocol for deriving an effectively unlimited number of addresses from a single **master seed, which may be between 128 and 512 bits long. Many users interact with BIP32 master seeds indirectly, (e.g., by storing a set of 12 or 24 BIP39 seed words). Unfortunately, these seed words correspond to a 512-bit secret, while codex32 works best with 128-bit secrets. It is recommended that users of codex32 generate a fresh 128-bit seed, using the instructions in this book, and sweep their coins to addresses derived from the new seed.**

I.6. Bech32 and Alternative Alphabets



odex32 inherits its name from the Bech32 alphabet. In order to store 128-bit secrets, we re-use the Bech32 alphabet, which consists of the 10 Arabic numerals and 22 of the 26 letters of the Latin alphabet. The excluded letters are B, which may be confused with 8; O, which may be confused with 0; and I and 1, which may be confused with many things, such as each other.

Parts of the codex32 process use an alternate alphabet, consisting mostly of Greek letters. This alphabet is used for intermediate computations, but never for data storage, and nothing represented in it is ever secret data. A table of pronunciation is provided on the Reference page at the beginning of this document.

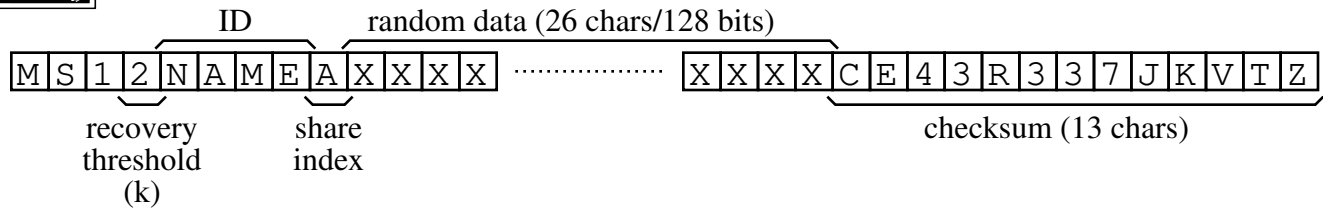
The remainder of this document provides detailed mechanical instructions. If you are interested in learning the mathematical theory behind this all, users are encouraged to check out the mathematical companion or contact Pearlwort at pearlwort@secretcodex32.com.

Part II: codex32 Components

II.1. Share Data



or a 128-bit secret seed, each share is 48 characters long. Shares begin with the three-character prefix MS1. This is followed by a six-character **header**. The next 26 characters are the data portion. The final 13 characters are the checksum.



The header consists of:

- The **threshold**, which is the value k , a digit between two and nine inclusive, although the main document only supports k values two and three. When secret splitting is not used, a zero digit is placed here instead.
- The **identifier**, which is four bech32 characters.
- The **share index**, which is any bech32 character except for S . The S index is the **secret index**. The data portion of the secret index contains the secret seed.

Shares of one secret all have the same threshold and identifiers. If you have multiple secrets, you should use distinct identifiers for each to avoid mixing up shares of different secrets with each other. The identifiers are not considered secret themselves.

If the user merely wants to checksum their secret and not use secret splitting, they should use the same format, but with the digit 0 for the threshold value and S for the share index.

II.2. Paper Computers & Volvelles



and computation for the procedures in this document can be performed either by using the volvelle wheels or principal tables (at the front of this document) to look up values. Although the volvelle wheels take time to cut out and assemble, they are generally easier to use than the tables, when available.

There are three volvelles in this codex. Each serves a different function, and has slightly different usage instructions.

1. Addition. To add two characters, turn the addition wheel to one of them and look through the window corresponding to the other. It does not matter which character is which; addition is symmetric.

2. Translation & Fusion. A common task in the scheme is to "translate" share data by a given recovery symbol. To do so, turn the translation/fusion wheel so that the window on the fusion side is showing the correct symbol. Then, turn the wheel over; the translation side will act like a decoder ring, mapping characters to characters.

Sometimes you will need to translate a share by multiple symbols at once. To accomplish this, turn the fusion wheel to the first symbol. Find the next symbol on the inner wheel; whatever symbol that it's pointing to, turn the fusion wheel to that symbol. Repeat for all of the symbols that you need to combine. The fusion wheel will wind up at the final product. You can now turn the wheel over to translate by this symbol. As with addition, the order in which you take your original symbols does not matter.

3. Recovery. When recovering a secret, you will need to look up "recovery symbols" that will be used. To do this with the recovery wheel, turn the wheel to the share you want to translate. Mark down the symbols pointed to by the *other shares* indices and fuse these together.

Important: Unlike the other wheels, the recovery wheel can easily be used in the wrong order. Be careful!

Part III: Process Instructions / Cheatsheet

1. Volve Assembly

You will need: a craft knife, scissors, card stock, brass fasteners, and the volvelle printouts from Module 0.

1. Cut out each disc with scissors. Then, cut out the windows on the top discs with the craft knife.
2. Cut out the small center circle in each bottom disc. Cut a slit along one of the small lines of the cross in each top disc.
3. Attach the discs with a brass fastener through the center holes.

2. Create New Seed (Initial k Shares)

You will need: the addition volvelle, dice de-biasing worksheet, checksum worksheet, pencil, eraser, and your secret seed. For each of your initial k shares, you should:

1. Generate random data by rolling dice, following the instructions on the dice de-biasing worksheet.
2. Follow the instructions on the checksum worksheet to affix a checksum.

3. Create New Seed (Additional/Derived Shares)

You will need: the addition volvelle, fusion/translation wheel, translation worksheet, and pencil.

1. Translate the initial shares using the symbols in the derived shares section.
2. Add the translated initial shares to get the new derived share.

4. Recover Secret

You will need: the recovery wheel, and everything used for derived share creation (see above). To recover your secret, you must have k shares available.

1. Look up their recovery symbols with the recovery wheel.
2. Fuse all of the symbols for each share with the fusion wheel to get a symbol for each share.
3. Translate the share by that symbol.
4. Add all of the translated shares to get your secret.

5. Verify Shares

You will need: the addition volvelle, checksum worksheet, pencil, eraser, and the share to validate.

1. Copy the share data into the bold boxes of the checksum worksheet.
2. Follow the instructions to complete the worksheet, checking that the final result is SECRETSHARE32.

6. Correct Shares

You will need: the addition volvelle, checksum worksheet, pencil, eraser, and the share to validate.

1. Follow the instructions above to verify your share.
2. If the result is not SECRETSHARE32, enter the result into the online tool. This data does not contain any information about your share data, only about the errors.
3. Add the given values to the given characters in your share, according to the online instructions.

X. Deriving Addresses and Spending Coins

It is an open question as to how to derive addresses or spend coins using paper computers. Please contact Pearlwort at pearlwort@secretcodex32.com if you believe you have insight into this.

III.1 Generate a New Secret

We generate new secrets indirectly by generating our n shares, which will imply the final secret. The process for generating a new secret seed is as follows:

1. Choose a threshold k and total number of shares n that suits your needs. The threshold k should be two or three, and n must be 31 or less. For $k > 3$ see Module 2, but this is not recommended.
2. Choose a four-character identifier for your new secret seed. The identifier can be anything (e.g., a name or nym), as long as it only uses the Bech32 character set. The identifier itself is not secret. However, the identifier should be unique for each secret seed.
3. Follow Section III.1.A to generate the first k shares.
4. Follow Section III.1.B to generate the remaining $(n - k)$.
5. Copy and distribute your n shares into safe and secure locations. Additionally, remember that you will need to recover at least k of these shares to recover your secret seed. Also remember that anyone else who recovers k of these shares can also recover your secret seed and control your coins.
6. Securely dispose of all worksheets that you used in the generation procedure. If these worksheets are not securely disposed of, they could be used to recover your secret seed.
7. (Optional) Load your shares into your codex32-compliant wallet or use the Recover Secret procedure in Section III.2 to compute addresses or access your coins.

III.1.A Create First Share

You will need: $2k$ copies of the checksum worksheet and the dice de-biasing worksheet.

1. Fill out the header portion of the k checksum worksheets with your chosen threshold k and chosen identifier.
2. Place a unique share index on each worksheet starting with share A on the first worksheet, C on the second worksheet, and so on through k characters from the Bech32 characters. Recall that the B and I are not valid characters.
3. Using the dice de-biasing worksheet, generate 26 random characters and write them in the **bold squares** of the checksum worksheet.
4. Once all of the random data is generated, use the rest of the checksum worksheet to generate a checksum for each share.
5. **Critical Step:** Verify your checksum by copying each of the 48 characters of the share into a fresh checksum worksheet. Follow the checksum verification instructions to verify each checksum. If any checksum fails to verify, make more copies of the checksum worksheet and redo the checksum generation and checksum verification steps. Failure to verify each checksum may lead to irrecoverable loss of the secret seed and funds.

Special rules for $k = 1$. If you are not splitting your secret, use a 0 digit in the threshold place, and use the S character in the share index place. Follow the same instructions for generating the data portion and the checksum.

Special rules for pre-existing secrets. If you have a pre-existing seed, include this as an initial share, using S for its share index. When deriving additional shares, use the alternate table in Module 2 rather than the table in the following section. This process is not recommended, but may be useful in some scenarios such as re-sharing an existing secret.

III.1.B Create Derived Shares

You will need: $(n - k)$ checksum worksheets, and the translation worksheet for your value of k .

The remaining $(n - k)$ shares are derived from the first k shares, using the translation worksheet. For each derived share, use the following process to derive it:

1. Make a copy of the translation worksheet for the value of k that you are using and label the shares with the share indices from the shares that you have already generated (e.g., A, C, and D, if $k = 3$).
2. Label the final share index with the new share index that you want to derive. This can be any bech32 character, but most likely you will just want to use the next available character.
3. In the derivation table (below) for your value of k , find the column corresponding to the new share index. Copy the symbols from that column into the translation worksheet, next to the share index for each row. There is an illustration on Page 18 if this is unclear.

k = 2	D	E	F	G	H	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
A	Π	Θ	Δ	ρ	β	α	Ξ	Ω	κ	μ	€	♥	η	@	ç	Λ	Γ	%	ε	¥	Φ	§	¶	#	†	Ψ	⊕	Σ	♦	
C	ρ	Λ	Γ	Π	α	β	μ	Ψ	⊕	Ξ	¥	♦	ε	#	%	Θ	Δ	ç	η	€	Σ	¶	§	†	@	†	Ω	κ	Φ	♥

k = 3	E	F	G	H	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
A	μ	Ξ	✕	⊕	κ	#	¥	¶	@	Γ	Σ	#	Φ	§	Ξ	μ	¶	@	Δ	⊕	€	¥	Δ	Σ	Γ	€	§	κ	Φ
C	η	η	✕	¥	€	Ψ	Δ	α	Ω	%	@	Ω	@	α	ε	ε	β	Ψ	ç	€	Δ	Γ	%	#	ç	Γ	β	¥	#
D	Π	ρ	✕	Σ	Σ	α	@	♦	α	Ω	¶	β	§	♥	Π	ρ	♥	β	Ω	Φ	#	#	Ψ	§	Ψ	@	♦	Φ	¶

4. Follow the translation worksheet instructions to derive the new share.

Derivation tables for k from four up to eight can be found in Module 2. However, we discourage the use of large k values, which are difficult to use and increase the chance of key loss.

III.2 Recover a Secret

You will need: k checksum worksheets, and the translation worksheet for your value of k .

Normally, you would not recover a secret seed yourself. Instead, you would load shares into a codex32-compliant wallet. However, you can recover the secret seed by hand if no compatible wallets are available or you feel a need to demonstrate your conjuring ability. The recovery procedure uses exactly k shares. If you have more than k shares, you can select any k of them and set the other shares aside.

Use the following procedure to recover the share:

1. For each share, fill in a checksum worksheet and verify the checksum. If a checksum fails to verify, you may have made an error on your worksheet, or there may be an error in your shared data. If there is an error in your share data, you can try substituting the share with a different one. Otherwise, you will need to perform the error correction procedure on your share, which will involve the assistance of an electronic computer.
2. Label the translation worksheet as though you were deriving a new share. Use your existing shares' indices as the "initial shares" and S as the "new share" index.
3. Rather than using a derivation table, fill in the symbols for each share on the translation worksheet using one of the following procedures:

III.2.A Recovery by Table Lookup

If your volvelles are missing or otherwise inconvenient to access, you can do the entire process using the Principal Tables located at the front of this booklet, or available online at secretcodex32.com.

$k = 2$. For each share, find that share's **column** in the recovery table and find the symbol on the **row** of the other share. Copy this symbol into the translation worksheet.

$k > 2$. For each share, find that share's **column** in the recovery table and find all of the symbols in the **rows** of the other shares. Fuse these symbols together pairwise using the fusion table until you have only one left. Copy this into the translation worksheet.

Notice that the fusion table is symmetric, so it does not matter if you swap rows and columns. The recovery table is **not** symmetric, so you must use this table in the correct order.

Once you have copied everything into the translation worksheet, follow the Translation Worksheet Instructions page as though you were deriving a new share. Rather than using the addition wheel, you can use the addition table. The result should be a share with index S , which is your recovered secret.

III.2.B Recovery by Volvelle

Using the volvelles is easier and less error-prone than using the tables. If you have your volvelles, the recovery process is as follows.

$k = 2$. For each share, turn the recovery wheel to **that share's index**. Look up the symbol pointed to by the **other share's index** and copy that into the translation worksheet.

$k > 2$. For each share, turn the recovery wheel to **that share's index**. Look up the symbols pointed to by the other shares' indices. Fuse these symbols together using the fusion wheel:

1. Turn the wheel so that it is pointing to the first symbol.
2. Find the next symbol on the inner part of the wheel. This symbol will point to a new symbol. Turn the wheel so that it points to the new symbol.
3. Repeat the above step for all the remaining symbols to fuse. (If $k = 3$, then no repetition is needed.) The resulting symbol is your result. Copy this into the translation worksheet.

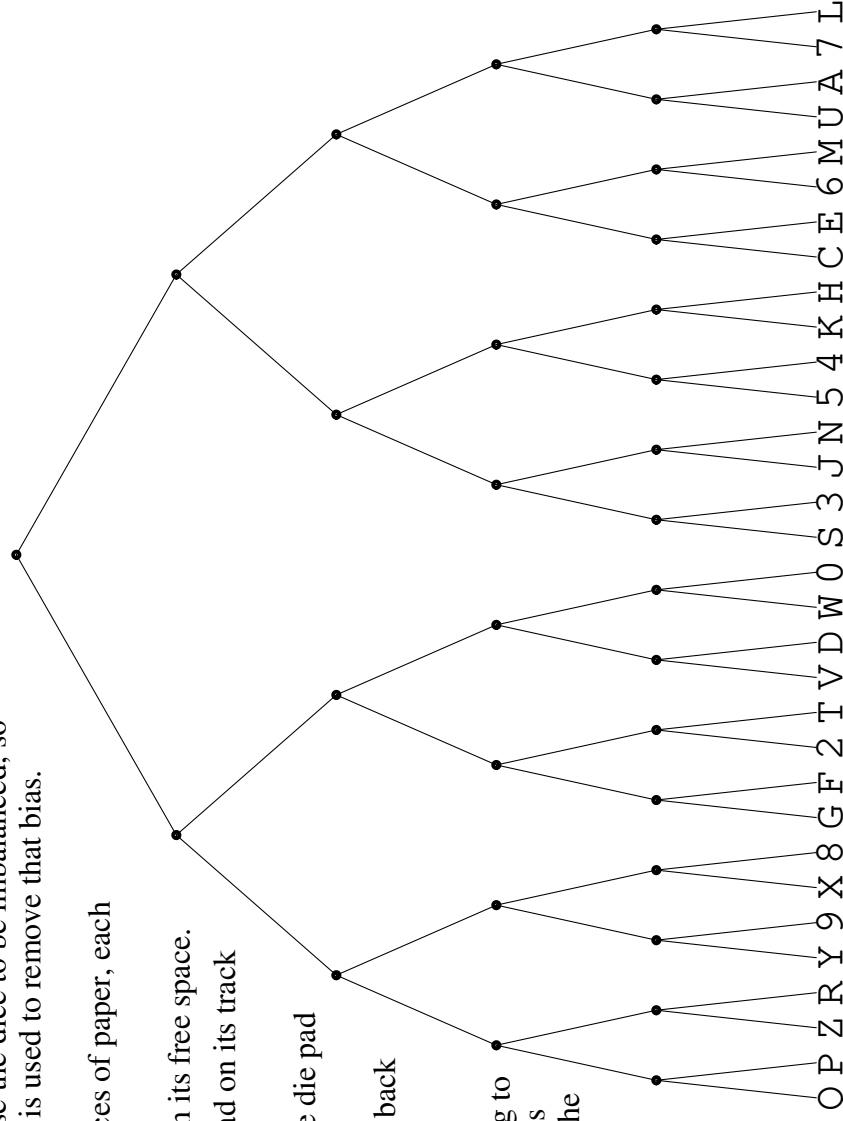
Once you have copied everything into the translation worksheet, follow the Translation Worksheet Instructions page as though you were deriving a new share. The result should be a share with index S , which is your recovered secret.

Dice De-biasing Worksheet

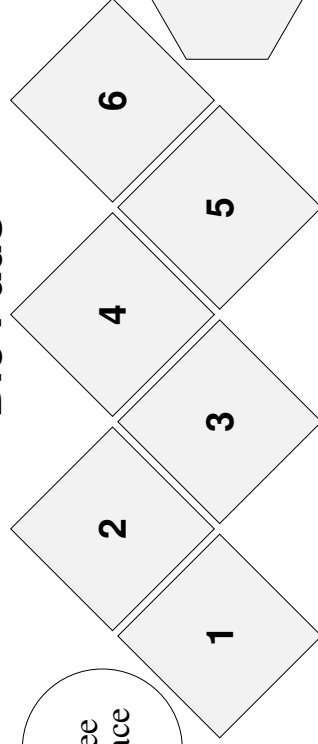
Most dice have small manufacturing imperfections that cause the dice to be imbalanced, so some values appear more often than others. This worksheet is used to remove that bias.

You will need: five distinct dice, five die markers (e.g., pieces of paper, each labeled by which die it corresponds to), and this worksheet.

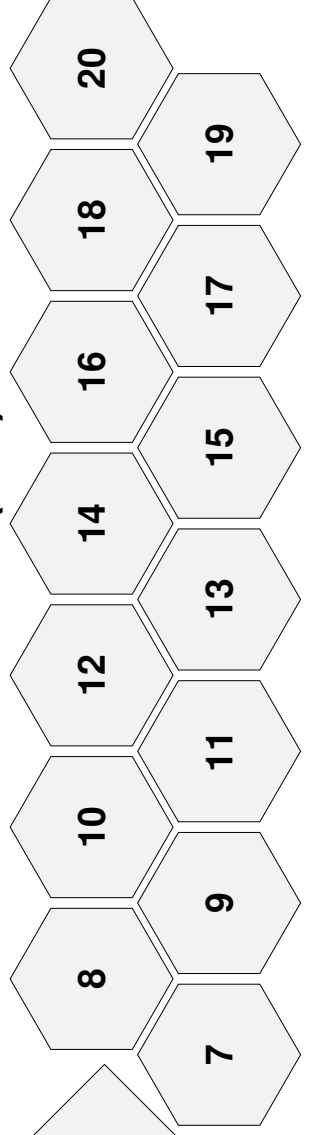
1. Choose a die track for each die. Put the die's marker on its free space.
2. Roll all five dice. Move each die's marker to the die pad on its track indicating its value.
3. Re-roll the same five dice again and set each **die** on the die pad indicating their second values.
4. If a die showed the same value twice, move its marker back to the free space and repeat steps two and three.
You must redo both rolls!
5. Using your finger, follow the tree to the right according to the die tracks. Take the first left branch if the first die is to the left of its marker, and the right branch if it is to the right. Similarly, take the second branch based on the results on the second die track, and so on, until the bottom of the tree, which has the resulting character.
6. Repeat steps one through five for each character.

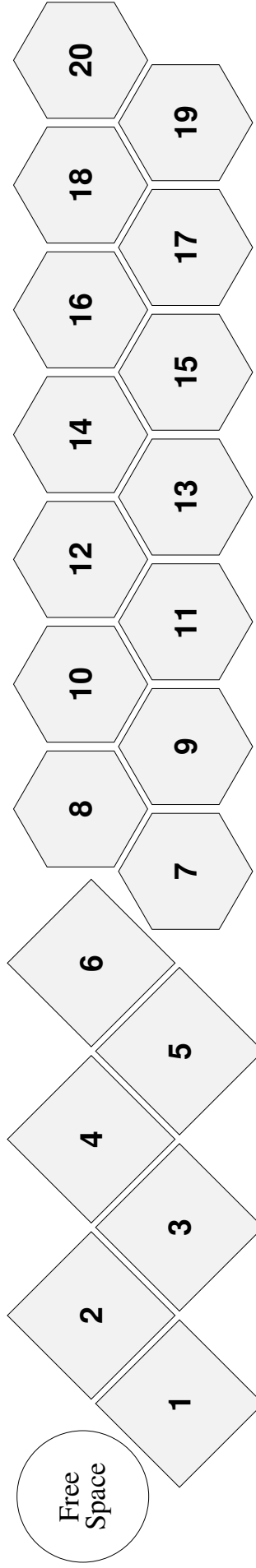
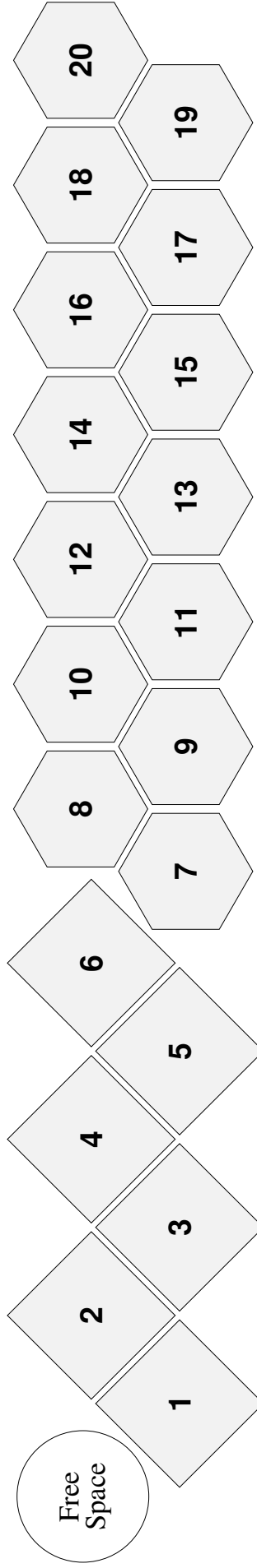
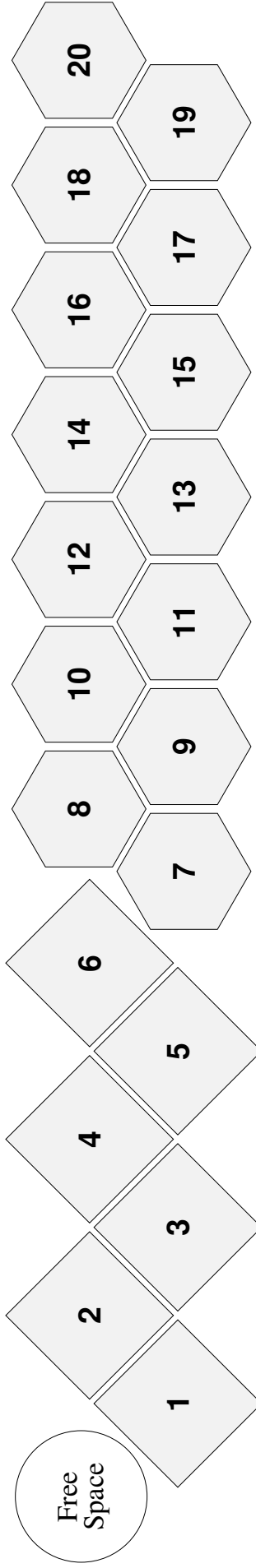
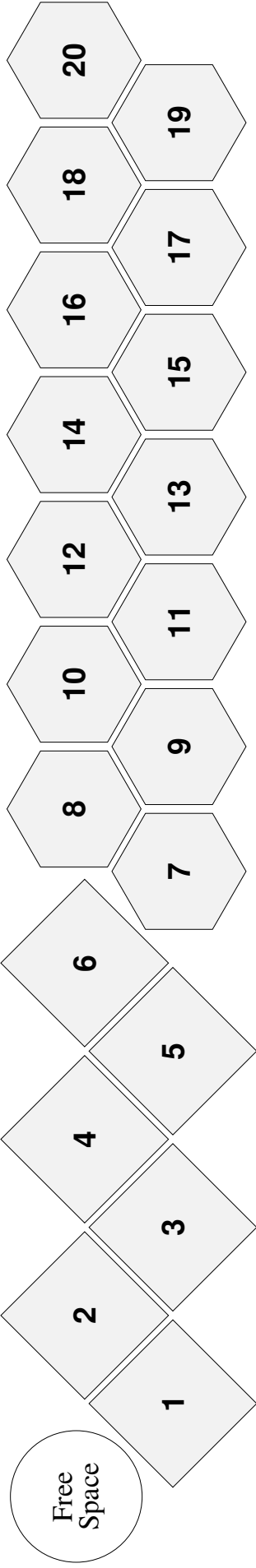


Die Tracks



Die Pads (d7+)





Checksum Worksheet (Generation Instructions)

The checksum worksheets are used to generate and verify checksums. These are the most frequently used and important worksheets of codex32.

You will need: a checksum worksheet, checksum table, and addition wheel.

Generating a checksum:

1. Fill in the top diagonal squares (i.e., the bold ones) with your random data. You should have enough data to fill the non-pink bolded squares.
2. Add the first row to the second to fill in the third row, using the addition wheel.
3. Look up the two leftmost under hanging symbols from the third row in the checksum table (pages 11-12) to fill in the fourth row.
4. Repeat the above two steps, adding the third and fourth rows, looking up the fifth to fill in the sixth, and so on. With this approach, you will be able to complete the entire sheet except for the pink squares.
5. To complete the pink squares, work from the bottom up, adding each row to the one above it until all of the squares are filled. The completed share can now be read from the top diagonal, including the checksum (i.e., the pink bolded squares).

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

5

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

1

M	S	1	2	N	A	E	A	5	Ø	P	R	D	A	K
+3														
=														

2 - 4

M	S	1	2	N	A	E	A	5
---	---	---	---	---	---	---	---	---

Checksum Worksheet (Verification Instructions)

Verifying a checksum:

1. Fill in the top diagonal with your share data; you should have enough to fill all of the bolded squares.
2. (Optional) Fill the bottom diagonal, if you have access to this data. It will help you catch mistakes.
3. Fill in the rest of the worksheet as you did when generating a checksum. If your final row does not match SECRETSHARE32, or if any of your computed bottom diagonal values don't match the expected values, there is a mistake in the worksheet or your data has been corrupted.

In case of error, first recompute every value in the bad column and check that you copied all of the share data correctly. Then, try redoing the worksheet entirely. If the checksum is consistently bad, your data is corrupt and you need to attempt the online recovery process.

1
2
3

M	S	1	2	N	A	M	E	A	5	Ø	P	R	D	A	K	M	S	1	2	N	A	M	E	A	5	Ø	P	R	D	A	K				
+3																+3	X	W	8	7	R	R	3	Y	I	J	G								
=																=	M	Z	M	4	7	R	H	V	\$	8	J	Ø	7	Ø	9	G			
																+2	M	D	J	4	4	A	3	8	E	5	6								
																=	3	W	N	W	9	E	9	6	R	G	8	3	J	L	E				
																+J	4	Ø	D	7	E	E	Y	F	P	A	\$	6							
																=	P	M	2	5	M	R	6	V	W	\$	Ø	Ø	R						
																+Q	W	R	\$	E	Z	Z	Y	V	8	F	E	F							
																=	2	6	C	N	R	W	V	5	R	G	2	3	4	P	6				
																+M	G	4	5	V	Ø	6	5	Q	3	9	P								
																=	R																		
																+D	G	X	H	Z	T	P	F	N	F	7	8								
																=	M	J	X	V	W	E	4	\$	D	H	J	X	7	3	Y				
																+D	4	8	Q	J	J	J	6	D	F	W	W	Q							
																=	T	E	F	E	8	Z	L	D	L	Ø	\$	L	Y	6	N				
																+T	Ø	E	M	4	A	A	W	2	W	\$	T	L							
																=	Z	K	7	E	2	\$	Z	P	6	3	5	3	V	H	9				
																+W	Q	J	N	M	2	2	X	F	J	4	X	Ø							
																=	\$	E	C	R	E	T	\$	H	A	R	E	3	2						

MS32 Checksum Table

Table with 32 columns (TA-T9, UA-ZA) and 100 rows (2A-29). Each cell contains an alphanumeric checksum value.

Table with 32 columns (2A-29, 3A-39) and 100 rows (2A-29). Each cell contains an alphanumeric checksum value.

Translation Worksheet

The translation worksheet is used to derive shares when splitting keys, and during key recovery. In all cases, the process is to translate a set of shares using the translation wheel, and then to add the translated results using the addition wheel.

You will need: translation worksheet, translation/fusion wheel, addition wheel, recovery wheel (i.e., for key recovery), and the derivation table (page 7, for share derivation).

In all cases, the number of shares to combine is your k value, the number of required shares to reconstruct the secret. The process is:

1. Make sure that you have completed checksum worksheets for all input shares.
2. Look up the translation symbols for each share, either in the derivation table or using the recovery wheel and fusion wheel.
3. Mark down each share's index (the sixth character of its header) and translation symbol in the appropriate squares.
4. Character by character, translate each share from its checksum worksheet to its row, using the translation wheel.
5. Using the addition wheel, add all rows together.

Notice that the resulting share will automatically have the correct share index in its header. **If not, you have likely misread the instructions.**

k=2 Example

Share Index Symbol

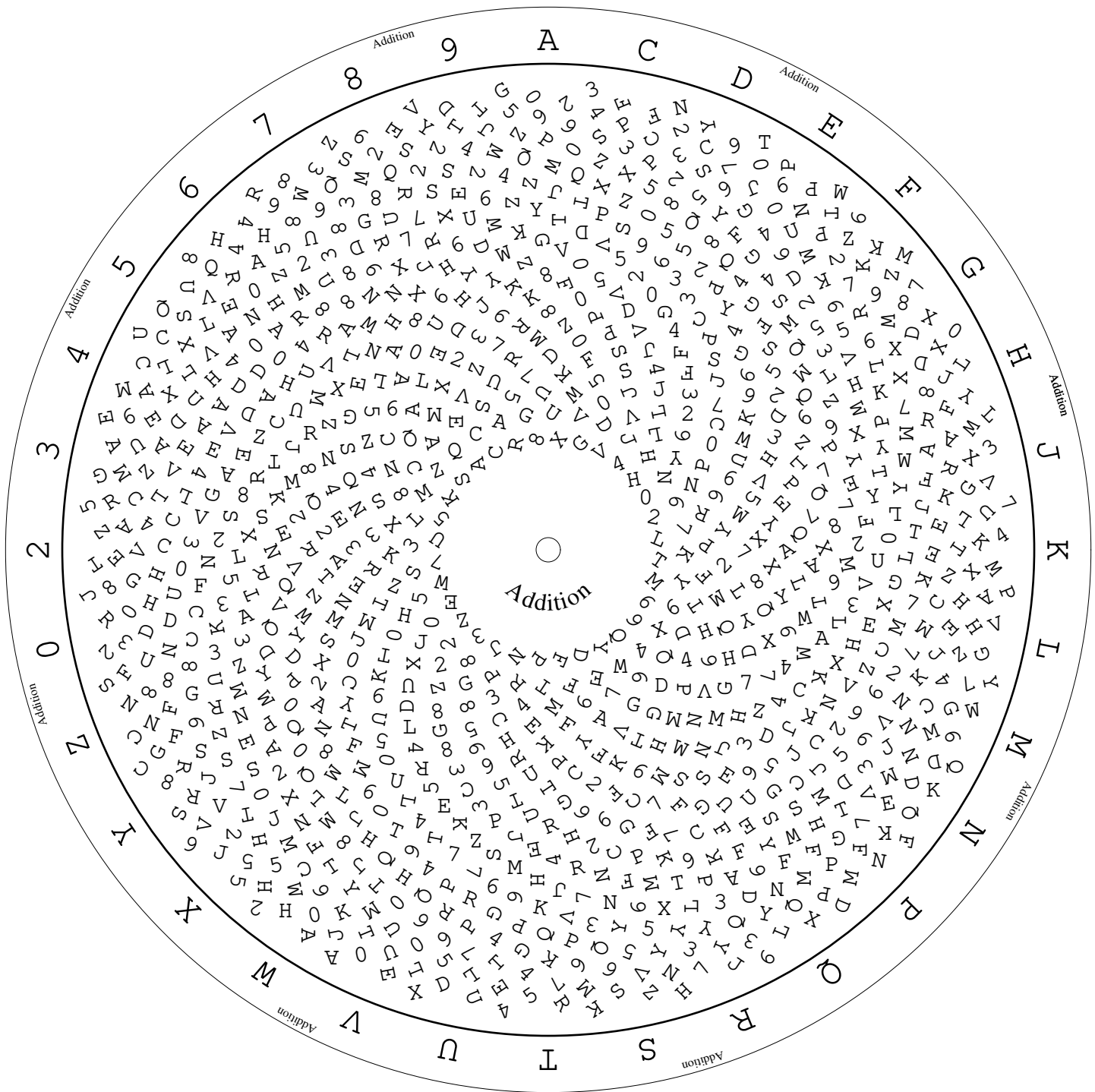
4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48							
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	aa	ab	ac	ad	ae	af	ag	ah	ai	aj	ak	al	am	an	ao	ap	aq	ar	as	at	au	av	aw	ax	ay	az
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	aa	ab	ac	ad	ae	af	ag	ah	ai	aj	ak	al	am	an	ao	ap	aq	ar	as	at	au	av	aw	ax	ay	az
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	aa	ab	ac	ad	ae	af	ag	ah	ai	aj	ak	al	am	an	ao	ap	aq	ar	as	at	au	av	aw	ax	ay	az

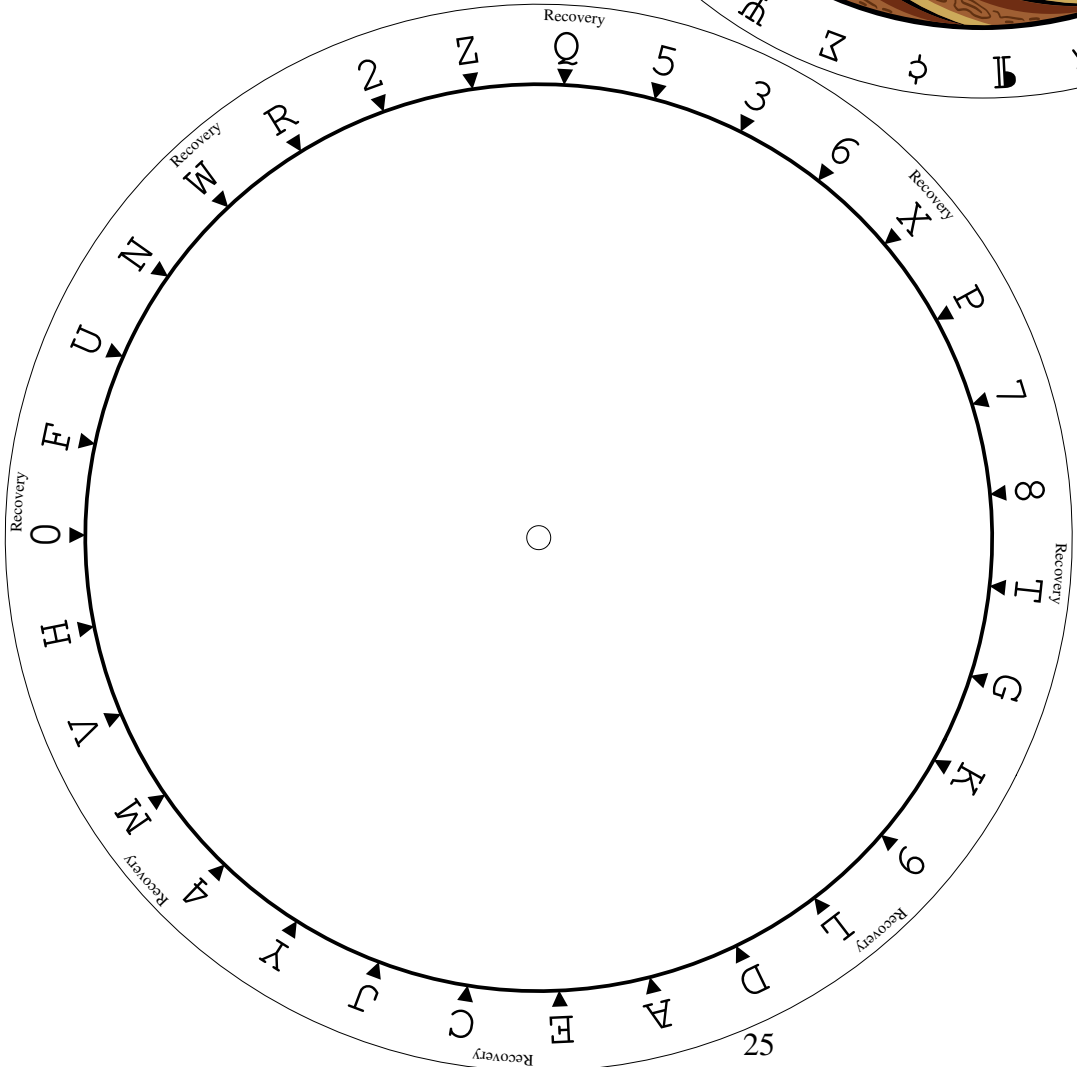
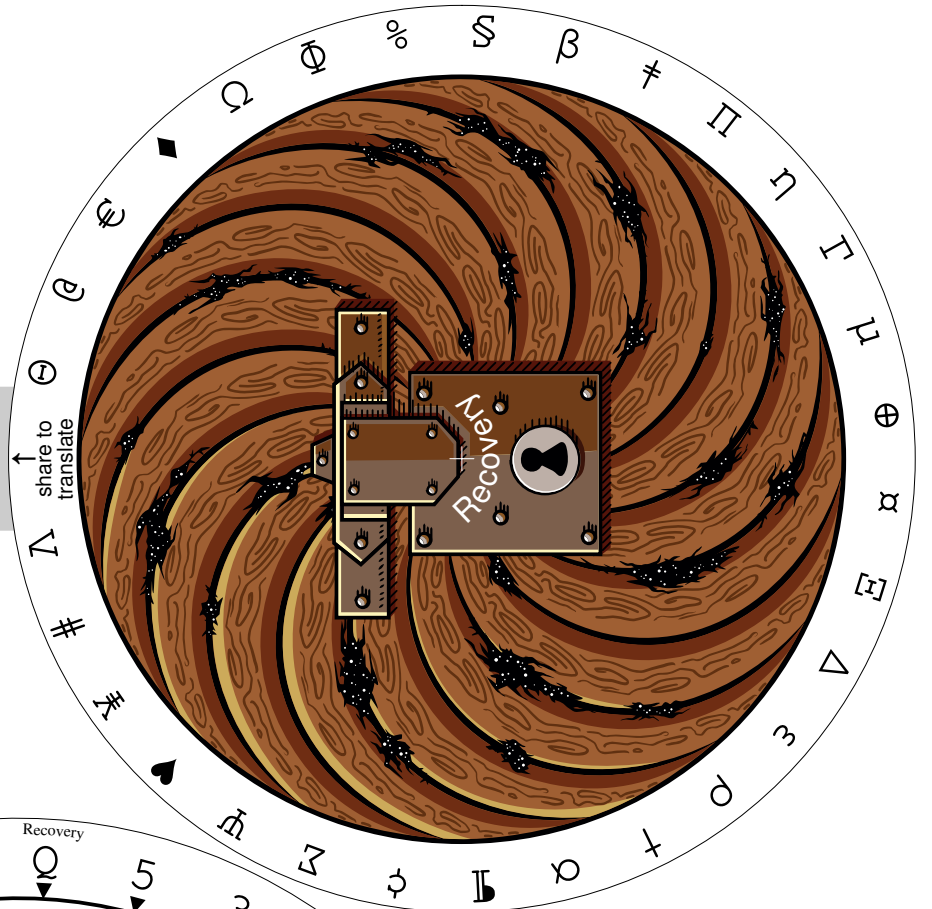
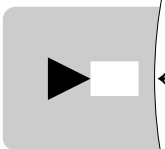
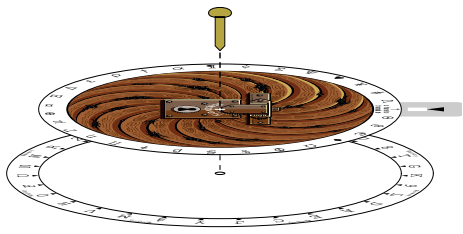
k=3 Example

4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48							
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	aa	ab	ac	ad	ae	af	ag	ah	ai	aj	ak	al	am	an	ao	ap	aq	ar	as	at	au	av	aw	ax	ay	az
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	aa	ab	ac	ad	ae	af	ag	ah	ai	aj	ak	al	am	an	ao	ap	aq	ar	as	at	au	av	aw	ax	ay	az
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	aa	ab	ac	ad	ae	af	ag	ah	ai	aj	ak	al	am	an	ao	ap	aq	ar	as	at	au	av	aw	ax	ay	az

Module 0: Vowelles







Module 1: Share Booklet

In the common case that your threshold value k is 2, there is a much faster way to generate shares rather than using the translation worksheet and the volvelles.

In this case, your two initially generated shares will be A and C. To generate further shares, go through the characters of your A share one by one. For each character, find the table labeled by the character, and then find the row labeled by the corresponding character of your C share.

All of the corresponding characters for the D, E, F, and additional, shares can be read off of this row in the correspondingly labeled column.

We have removed the S share from these tables since this share contains your secret data. If you want to generate the S share, you must use the recovery process.

CDEFGHJKLMNPQRTUVWXYZ023456789
A AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
C PFK8SR65L3DMVUQ07E9ZTWH2YNXGJ4
D U0EM7XHR8NJ6LS3G2QPT495CVFKZYW
E HDUGR9TW02QZF57JXS6VLM4KNCYPY38
F KLCWQ7P54Y09TEJ83DX6HRUNZV2MG5
G QMJPNCS29LZUW3V6FYE5RK7048DHTX
H VUTQ8GN0E953KLWSM4Y2CJF6XPZ7RD
J SGQ62K5XMFYH87NZC3U4WPRDL0ETV9
K 6CP05WZ4F7EGNHSRDRUMYV8TX329JQL
L C4FRJ3KQ5Z8XHDGWY02PU7EV6TN9MS
M J9GKVFQNX46ERYTPLZDS7C385W0UH2
N XV24ES9UTJFWZKDDLQCRM65P3GY780H
P ZK6D48YLCRUJ2T5EWHG3N0V97XMQSF
Q 5JSZXP49GC3T0R2YK7HL86WFEVDUWNM
R 87WN6T0Z3UXFQMP2NH9LDJVG5ES4CKY
S 4Q5Y96LMJK7VDWX3PRTF0Z8UCEHN2G
T NHVS0J2DUM47PF885GL3XKQCZ96YRWE
U TEHJWMV8DXSYC4RQ95ZNFGLP2K6370
V 2TN5DQXEHLR6C04JF79PSKYMZ3W8U
W 0R82ZVDY7H9CSG6XTMFEQNJ4U5LKP3
X M29FH4GTNSK036UC5P8JYLZRQ7WDEV
Y 7Z3HCERK60V5M2FTDNSW9UXJ8GQ4LP
Z 36YUFD7CP8TS9NLH0VQRXE2GWMJ54K
0 E8D932U7WVGP4QPM2HJKH5XSFTLC6ZR
2 9NXLU5MHVQC8YPEFSKVGZ467J3R0DT
3 RY7TKUWPZDN4GXCV2E58MH9Q0JSLF6
4 F5L7GYCJS6W2U0MRZ8NKE3DTPHVX9Q
5 LS43MZFGQPRNE8976WVCDY0HKUT2XJ
6 YPZEL03FKKWHQXV4U8TJ72DNMR9GS5C
7 W3RVP86YE2LJ9KNUX40GTMSDQ5FCZ
8 DW0XYNE3RTMK5JZ9VGCUS2QLH4FP67
9 GXMCTLJV25PD7ZHK460Q3FYWSR8EUN

CDEFGHJKLMNPQRTUVWXYZ023456789
A YVNZ4XL365G7FE92MUQ8WTJ0PKRDHS
C CCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
D 97HYKA40Q68ET5FLVPU3XNM2SZGJWR
E WUJHT7FZGR4PASX9Q38KVD6Y0NL5M2
F 0WK3G9RUJD6XLA2SHVMQY4NP785TZE
G E2U7MRJXZKHL645D09YWSQ3AFVN8PT
H 4D9L0N3R7VPJZMK8A5ESTYXG62UWFQ
J FEW9ZDK2UQ35N6T47SP0AHVLR8MXG
K RF0SU4QEWVHT8NG69AX7L3Y5DPMZ2J
L PY8U625VNTJ9RFSEZWHM0GK7XQD43A
M T5XFYJZLPU06HQNKERS2DW74G93VA8
N LAY23T8SVMUDKJ4GXE7PFZWR50QH96
P M85JAUXNL2F370VWGK4TQERHZD9S6Y
Q DR7AW6HF03YGM8JNSL295VPT4XZUEK
R 70QVJSDWK4N25LEA3YZHP68X9MGUF
S V36MDPAH4LT0E27X8ZKNU5GWYJFRQ9
T 2XZ08FPGMJQA4DLRW7VU9KHS36NY5
U JGEDXQWT2098V3MHR4LF67SNKAYP5Z
V N6AT9MY4SP2QWUHZ5GRLJXEK8F07D3
W KJF42H0GE7SMYVZ3D65RN9A8QLPXTU
X ZMTKLW285ERV97Y0JQ6GFZWR3U4SANP
Y 8NLGSZP6AXEH0W3UTJD5K2FQMR794V
Z GT2RPKU5XW7N3H8QFDAE4096J5VYLM
0 QKR6E37JF9AZPYUV4INTD8SLMH5X2GW
2 UZGQ50EMTFDYS9P7KHJN3R4VW6AL8X
3 64S578VD9YXKUZQMLTFAGP2JNEW0RH
4 S93PQL67HNMFGTR5YXWV28ZEAUJK0D
5 XPMWNETY8GKSDRAF03Z7JQ92H46VL
6 ASVXH5N938ZRJGDT20YEMUFLWKQ74
7 HQDNFV9KRSLUXPWY68G4MA5Z3T2EJ0
8 5LPEVGMAYZW4QK6J2F9XRU0DT7H3SN
9 3H48RYSQDA5W2X0PNMJ6ZLTUVGEFK7

CDEFGHJKLMNPQRTUVWXYZ023456789
A VLFTWK8NHRZ20QPC6S3M94YGUExJ57
C STZ3RGQ640JV7PU2E5FYNXWL9HA8MK
D DDDDDDDDDDDDDDDDDDDDDDDDDDDDD
E X0G7SW3Q95L4MFWZHPAK28NVRJU6TCY
F 8AVCN4M7L6SJEYVUQ2U0T9XRG35PH
G 3EXHQ2Y0PATUVSLWF4JM7865RKZCZ9
H A7LK5RFPNMTXYZJ4UCGVQ6S089E32W
J P25VEAWG3HMQ4R08LUSNKF6C7TZY9X
K T64E89CMRQXLP2VGY3HZ50JNSW7AFU
L FHA4P6VW7UC39S5TRZX8YKQEM0G2JN
M R8UQLZ6AVT9W3EHYC0PKXSGJ425N7F
N HW7R2MLJVPVKEST36840AZUCYFQ9GX5
P N5WMXV73JAR9CKGUF6YHT84SLZQ0E2
Q 9SY5420TZXWUA7KP3NMEJHVGFR6C
R KU69FQ4VMZE7JXA0SGNT2Y3PC5WHL8
S YZQJK394CGPMLN65XW80H27FEAVURT
T Z4CXUESRK92FN5M30JAQWGPY7LV86
U 6MRYASKF8C0N2GL9ZEW43QX5TJP7HV
V MF8Z7TUHAKQ5G9NS4YJREC036X2PWL
W 7PNU38H25F60Z4XRVK9LXMTQASVEGJ
X 2G3LY7J9EWFRCR8QANVT5UHMKP64ZS0
Y 0Q9PTJEC3NRFH4W27UGA5L8XVM6KZ
Z QCS26XYKTE58HWRJGVP973NA0LFMU4
0 G9ENZPXSJYHK8AC75L63VWFU2MR4TQ
2 53JF0LPEX78SKU9VHMZW6ARTN4CQYG
3 JX2A9H50GNVZ6MYF78CPRLU4WKTSSQE
4 CKTGM0ZU6Y3AWJ8X92LSPE57QNHFVR
5 WJP8GFNX2LUY36EMARQ74VKZHC5903
6 4RK0VY28USGH53FEMARQ74VKZHC5903
7 LNH6JUA5W84GQC2KMTFEFSRZ9VY0X3P
8 UVMShcrlf4YPX07QT956GZE2K3JWNA
9 EY0WC5GZQ276VLTNJHRXFpAM38UK4S

CDEFGHJKLMNPQRTUVWXYZ023456789
A NFCV8P02TWYXDS6KZ57GML3JHU9Q4R
C 0ANK2LGRFZ5QU38YPSXHDV9W7J46T
D JMU2Y6953QTP0AKR4FZ7N8CHXGWLVS
E EEEEEEEEEEEEEEEEEEEEEEEEEEEEE
F CTAL6ZN8RHK7M54VWY903PUSUGJXQ2
G W0HSFYZANVM87UT3KD6P95JQLX42RC
H ZGW3A5PC0KD2XJFMYU8L7S94VQ6RTN
J 7U9TS2X3D6AVHN5F8CLQGR0Z4WPKYM
K 5VY7WJSZLDQNR6HXU4C3298FMTA0GP
L KPVGJGDYHZ37A8Q09MXF56U4RS2TCNW
M U3D8K4JYSXRZNFV2QTW9C6AG70HPL5
N GC0YRVHTAPS4JM25L3QWUKD7Z9X68F
P VZLU0MKGWS9F6XNJ37TY4DQ258RACH
Q 6X4NUA8J7TGLSLWDFH52PCZKRZY3M9
R F2TZQHA480LJSKXPGVUC5WYMN3D976
S M534LXDVY98HARP6772GUFQTNJC0WZK
T ARFP4WC62GV93YQLHKJNSZ5D0MU7X8
U 9DJR587SM4FLGCTY6APX02NWQH2VK3
V YLK9HU5WPMXC24G7DQAS8J6T3RFN0Z
W PHZMCSLNGYURQ9AD5J2VX376K48TFL5
X 47QCDF6U9R05PHMNTGY8ZAWV2LKS3J
Y SK5XZ93PVU40T8WQJ6NMR72ADFCHGL
Z LWPDN3V0H5JT47CUS9RKQMX8Y62FAG
0 HNG5TKWFCL369DRSVM4ZJYUXP7Q82A
2 T8RWXGFQ6NPU5V7Z0LDAHYHK3CSMJ94
3 DSM6VQUK572WCTL8XRHAJA4F09NGZPY
4 8Q60JC29XFH3VZUGAWSRLNPHYTK5MD7
5 3YSQP7MLKJ6GF2Z4980DTXRCUANHWV
6 248G9NR7QAWMKPJHC23T0L5FYSDUX
7 Q9XAMT4DJ2NYZG3CR0K6WFHL8PV5SU
8 R62H70TX4CZDY9WNPMPKGVSA53UJQ
9 XJ7F3RQMU8CKW0SA2NV4HTG6ZLY5D

CDEFGHJKLMNPQRTUVWXYZ023456789
A ZTV6524YPSM3L DXN9EJWRHG8KC70UQ
C 7L8QE5JDRUTHWVMPXA234YZS0K96NG
D REGUH3L6M75KAQ2XJYWCT0PZ8V4S9N
E GMD02JRH8XWTY9U754LP3QVCAZKS6
F FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
G LDRX60CNEJYSVPH53QK8AUT47ZW92M
H N268WLMK9GJA5047R3T EXCUQDYPVZS
J 073ARPUL8YZM9WGVQ4NXSTKH526EDC
K 9WSGAE2V4NLY38TRMC5HJD7U60XQPZ
L 8RCYMX7EGOP24ANQUT9JZ5VK3WSH6D
M DNE397G26CU4P5S08XZRQJYALTVWKH
N E6MJS8D9HL0ZQXK3CUVGY75TRPA4W2
P AQT2USVXYW67GM0HKN8ZD9EL4RCJ35
Q TYP90KAU54H8DN32W6CVESMRZGL7JX
R CGL5NU8MD3Q9ZT6Y0P5S7VXAWJ4K2HE
S 2K9RVDHZWMCQ07ALE8Y63GJXNU5PT4
T VPAHX9Z5QKNJREU6SM74G2DCWL830Y
U 50X48VY73TKG69CWASDQHZ2MPNERLJ
V 4AZNYHHWQT9E0CG5M2D3KL6R7S8JUXP
W S4KDTM9AZ6R5JCPGNLX27E80H3UYQV
X YU5W7ZQJ0ASRN28KV9GP64HETMDL3C
Y P5QS3WT0XZ2CE6J94HLAMKNGVDR87U
Z WV4MQ6KPA2DU8RLEYHG0SCNLJ973X5T
0 X3UZCA58JPWDHSL4TKEY2V9NQ6MGR7
2 69HC4RNWSD7TX3Z8GJPMUL0YE5QAVK
3 UJ0VLTXC7Q4E2KRZPWM59AS6YHNDG8
4 KZWEPEPNS TVHGX7LQD6RU98MC32J05YA
5 QXYKJ4P3UV9LMH7SZ2RTNW6DAEGC80
6 MHN7KCES2R3VYUWJL0AD58XPGQTZ49
7 38JTGQ0RC5VNS4DAYZ6UKPW29XHML
8 JC7PDY3GLXA6KZET5VH0WQ49US2NMR
9 HS2LZG64KE8PUJVC7QNR35MXYTAW

CDEFGHJKLMNPQRTUVWXYZ023456789
A 4W85XD9LS2HFMYE0U3VPKRT6QJ CZ7N
C F6VWNTNZKJX8029YAL457QSPDEURH3M
D 5URJ9K0P2YCWEF46TVSH7MZL3XNA8Q
E L7XPC4NVZA2K0D6Q8UTS59R3WHYMJF
F 7VZ8MWEJCN93DQK4SPHXUA25LR0YT6
G GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
H Y03FUCTD5J79SX2Z6E4KNVLAMW8PQR
J CM6N3XVYK7WRP8HSEADF9L42ZQU50T
K JTMX076HYFNU4W5LZS2C8EAPV9QDR3
L SXN2FPWZE4DJ35VU9RMAHQ0T8YK6C7
M 6KJLHEC3TZSD9A0N7WUV4X8QFP2R5Y
N WLSUZQA7X9R6Y0FDP5J832HK4TMCVE
P 29QYWHUA45KXVJST0MEDC36ZRF7LN8
Q UP2TA3D890MLF6WKHJXRVC75ZENS4
R 0D56PMHQUTVAXZ9CKF73EJ7NYLS842
S ANWD728ELP5CTHZRQ064YU3M9KJVFX
T MYKE5ZJ03VL28SRXFNQ6A7W9C4PUDH
U R2DM4T59Q36S7V8JYCN0ZKFXHELWAP
V ZCFAKS7M6L4HUPT8N90E2WQRXD53YJ
W 8SAREU4XNQ0VK3752HC9TDYJPM6FZL
X NELQV9SF78UMHRC24DKW0P5YA3TJ6Z
Y K3T7RFM5HCXQANDEVLPJWZS46892U0
Z EF74JAX6VSPYR2M9WQ3LD8U0N5HTKC
0 35HV26YURMZ4NEQF78JLTCXWKS9PD
2 DQUK8YR4PHJNZCAM36L5FTVE07XSW9
3 THYZDVKR06EPWLU7CX9MSFN8JA4Q25
4 P89HN5QSADY76KL3RZ2J0MVCXFEXW
5 HR0CQJ32DKF8L7PVMZAYX6ESTNW49U
6 VJCSYLFTEMA5Q43WX8RZPN9U72D0HK
7 XZE968LCFWQT5UJPA2YNR4DHS03KMV
8 9A40LRPNWU3ZJTXHDYFQM5KC26V7ES
9 Q4P3S02W8RTECMNY5K7U6HJFDDVZXL

CDEFGHJKLMNPQRTUVWXYZ023456789
A XKP2DZE9N07MU4Y63LWQJCRSVTGF5F8
C 6Z2SQU7V3RWAD5EGJMKLYTF849NPF0
D 8Y9VWENMFP3U7CG0XZJK6Q24ALRTS5
E VXQLN60KPCRYGU842JF3S7TMZ5D9A
F D5YESP9GMKVR23TQA048CXZ7N6LJUW
G MSWKR85JTD60E4A9X2FVNQZY3C7LU
H HHHHHHHHHHHHHHHHHHHHHHHHHHHHH
J TRUD6FS74M83XK295N0GPYAQWEVZCL
K PNACY3XD8V6WJLF207GERZ4TQUSM59
L R745ZWJC6SYQK93FGDEUNM8PTAXV02
M FW5PUKYTG8ELZVJXNQ7D3A029C64RS
N A8KZF0PY9Q2GR75CV6SX43LUEJTWMD
P EAX69CQ8K3L5TRD7Z4MVU2JG0SWFYN
Q 0EV4K73AX2JDWTNR6UYZGLS5CMF98P
R U4JY25T6LW90PNC8M8VSAFKEGXQ3Z7
S WTGN49MRUYA2VXLKDP5CQ8E3F0Z67J
T GUS8LDW4JFKCP7N9AZME9X05V326R
U SJT97YGLR5NZEAE68FK3WXDPVMQ0C24
V 3Q0RALZPE6U9MSKJ7TDCW4GF25Y8NX
W 5GMAJNFUS9X73QRP8E6Y0KVCDDZ2L4T
X QPE782VNAZ4FSJ9LRCR50T6UW3GMYDK
Y 9FDQGX8W5A0J6ZSVP3RN2ECLK74UTM
Z 23CTEJ6Q04GKYMXSRWN7FU59LD8APV
0 ZV3JP4CXQ7T85GAULS92MRWY6FDNKE
2 7C6GVTL0ZJMP9FQWU5A4DSYNR8KXE3
3 C0ZXR2EVL SNFPT4G865JMD7Y9KAQ
4 JLRFCMU27GDVA8ZYW9QTK5NXSPE036
5 YMFXTADSWNQ4C0UEKVL9ZP36827RJG
6 L27W0S43CU5X4YEMVTFPR9GDKJNAEQZ
7 46LM3GRZ2TFEND05SYXJ8W9AUKPQVC
8 K9N35VAFDECS46MZQ2TPL07JXRUGWY
9 ND80MQK5YXZTL2W3EUA7V6RP4JSGF

CDEFGHJKLMNPQRTUVWXYZ023456789
A L8049EMF5XTCG3UDH7N6PWVZSQKYR2
C RNY0G8UQK2M7ESPL56T9HAX4FWD3VZ
D 74TNXZ3HWGYL2USCQR0VFK98P5AM6E
E 4HWQMPD6X3K8U4LZNMV5TRGY792A0S
F 592XC68YULEQ7UCLHNWGTSDV03PKR
G ZPQFTUK7VY5EMAD2R8HNL90SC6XW43
H WXEGLV4MS7Z5RN0Q3K2DYP9TUF8A6
J JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
K CZN8V2YPQ90DXM3AFL4RS56EUHWT7G
L 60MT24S5AE3RZPF7WVYXQDGNHKCU98
M SL675DX80WVUKG234PRH2TQOCENY9FA
N YKCAP5RGZFLTH6V02MDUX8SW9E473Q
P QVG9DRZT3C2HL84FY5XK0UA6NMSSEW7
Q KGZ279N3PR8W60T5UAECMFLXYSH4DV
R 9YUMZ0FKC8SV4HQ6AX32WLET5D7PGN
S H6XVA7E0MDGFCZ8PTQ9WN3KR4YU25L
T 3D7CHKVE4QRM59XYZULP2NFAG806SW
U FR96KL2NYAXPDEZS0HV54MW78T3GQC
V G3PU4YQD7NFX05W9PC2SZAR8MKL6HET
W DE426GTSHVNA9YMKPC87UQR23F50LX
X ESHPO3WL6TQ2YKAG7ZF4CVNUDR958M
Y UCRLQA9ZN563XWGM8S7FE0HD24TVPK
Z NQK53FCVGAU4SL7890WY62MHRXEDTP
0 MALDFW628H7YQV9TE3CSG4PKXZNRU5
2 8F5HYSAR9MWZ3DC6E4Q07XTPLVGKNU
3 P7VRWCG4TK9SA2EUNF6Q8Y5LZ0MXHD
4 TWDK5Q7XEPFC0FR6NGYA39ZU5V28LMH
5 A28ERX0UF64KVTYWDGZL3H7GMPQNC9
6 XMS38THAL4P9NQ5VSDGZ7ZYWCRF20
7 VT3YENPWDZU68FHRK9MG5C20QALSX4
8 05AWUHL92SDNP7R4XTKMEV3Q6GZCYF
9 2UFSNM5CR0HGTWKKXLEP8D643A7VQZY

CDEFGHJKLMNPQRTUVWXYZ023456789
A 3N2YL9F7Z4JRCUMXGHS08VQD6PWET5
C D8P7JMWHEZ9FAX5TNGS3L26R04VYU
D SPWZU7RQCMAY63HNL2VE05FGT49XJ8
E MRT3PALDSZ0UV4CFN6H792GX5YJQ8W
F 630HZX4N2VQMWC5UY8TGDJPR9SEL7A
G V49C3QYF6HDZTS2PUW5XE8RLJM7NA0
H QJUE9S8VXDM0N7GTWLFZCR52PA364Y
J UXHRGWQ9YPTFZL4EDM385S7ACN206V
K KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
L 5M76SFZRT2GCJVV4398NX0YUAHQPDE
M 7TL0W35SEA48X9D62GQZYFVHNJUCPR
N 2ZAVMG3L56XS8HTY4JWFQ9UP0CDRE7
P WCD5HLSU8TNV02JZMA9RF734E6GYXQ
Q CU8MYEPXHS742ZVLR56DATNFW30G9J
R TSE2CNMPW5FH9683Z0JLGA4Y7VXUQD
S EWRA8ZTCD73JG0Q25FXM4N6VL9YHUP
T LEMFD27WRN6QYGP0A4U5V39JZXH8CS
U 8HQTVRICYJWL6A59MS70PNEZ3D2F4GX
V X9YD0CJ6GQSALEFW8RNHMPT5U7Z234
W RDSNQ5E8PL2X4FUA73YT6Z09MGVJHC
X HYJS4DUGVCE35M6RPT2Q7WLN8ZAF09
Y JVXW6PH498R27T0SCEAULDMZQ5N3FG
Z A5N9T42M70YQWJEV6XD3UGHCF8PSRL
0 4F6UNJGA3Y8LSPZXC9M9WHDEVRT752
2 FA3X7V05NGHEPQLJ9UR6CY8W4DSTMZ
3 02FJ5Y6ZA9UTD87HVDQ4PXC5GWRMLN
4 96G82UV30JJP5EWAACHD7YRQSMXTRLZNF
5 N7ZGE6ATLFDVUXR90YP2H4J83QCWSM
6 G04QAH92FXC7RDN8JPLVSUWTYEM5DZ7
7 ZL54R0NEM39PHYSGFVCAJ6XQ2U8DWT
8 PQCLXTDJUR5G3NY7EZ4W2MA0SF69VH
9 YGVPF8X04UWNMR3DQSZJTCE7HL5A26

CDEFGHJKLMNPQRTUVWXYZ023456789
A 6HTPSN5ZKQ9Y402VXDGR7UMWCF38EJ
C XRQMADNT5V4ZYWH79FSUG03EP82JK6
D M908NG7UVW2RHZXE3S54KYJTF8CQP
E S825WTZ3YHCMPXFRAQ0JU6N9KVD74G
F 34WJDSG07EHURT9K2ANY5Z6Q8CXPVM
G 83YA7KE4WZ69XRMTJ5V2QHCUSNPD0F
H TKAU2X6SJC7V7DEPQ935MN0FR4WY8Z
J HZKX8CAES5UW0VYNRPFTDQ976M43G2
K AJHNEQT2ZRP3M98UCVW60XD457FGYS
L LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL
M 4072P8FVDDGZQTKUSYJCWAHEH536RXN9
N PXUF57VRQ03H2Y6WMMGK9E48ZDSJATC
P 9UV3CFDQ7Y7TZERG48A0SW2KMH65X
Q NCX7T0U6R9FJ83A4DWZPYMG2VESKH5
R Q5C0H9XA6P7SGFKMV42N3DW8UYEZJT
S J2ZCG5KYETX49U3Q6N7HVRP0ADMFW8
T 5A6VZURJHXD8FMS9N0YC4P73QWGE2K
U VNPWR49CXMGAS853UCVW60XD457FGYS
V DP9GQW0XU486J2CYFETMZ3SH7KA5RN
W GF3K0ZYMA42APC6DHSTU8RJ5XEQNV97
X UQN46MP5CDWKEGTFO3JYV87YS92ZHAR
Y EG8T4H2F3J5DNC76KR9SXAQFPZUV0MW
Z KSJQYRH826NFDPGX5U4A9CVMT07W3E
0 7DMEUY4P93SCAJN2GZRFH8K6WT5QXV
2 ZESR36JG8AQ7VNWCTXMKP5UDH904FY
3 YWGHMJ87FSTVQ50AZ6PECRKN2XU9D4
4 W7FZ923DM8KN5AVJEXH6STCYRQUP0
5 C6RDKVQHTUM234J0P7EXW9FYN8SZA
6 RT59JPCKAN0EW7ZDUM8QFV4GX3Y2SH
7 FM4SVEW90YJX6HPZ8KQ3T2ARG5CNUD
8 2YE6FASWGR0UQ45HCDZNTXVJP9M73
9 0VDYX3MNPFE5K5Q8W267JGZA4HTRCU

CDEFGHJKLMNPQRTUVWXYZ023456789
A 5RWS20X4QNUL9ZD8EYTKC7HPJGF63V
C GH9283PUZLVE5RS46QNFWDKTXJ07Y
D 432KF5VT6AN80PJWEHQXS7YRUZ9GL
E 7N095HSXJYPQ3TGWZFFV86CL2UD4RAK
F LPTRHV6CG2W8NXZA45S3JQK0DE7UY9
G F9PVPY2TZC3R7KWUQDA0LX45N6JES8H
H 9ZCDS6G8YT4NWQ72LVJPA3RFX5KE0U
J EKNHAY0WX894LPRCUG27TZF3S6DVQ5
K NXJZRUEA5SC2TGQH89D0FY67L34VW
L 3T6W9RDGFVXY0JC5QKU2EANS478ZHP
M MMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
N 0JECWZ75KUGV6FA9YP4SLHTD832QRX
P TGFQZ4LH9DASJ5YR2W76KVXE3N08UC
Q AVR03NWD4KSFHU67J8P5ZEY9XCGTL2
R WQA7DE52VJ8TCY3SNUFXH0ZGK9PL64
S U78FJGYN0CLW43KT96AZ2PDQHVR5XE
T 6FLACQ39P45UEKHVX8DNRJ720SYZG
U Z8YLEFH0SG3XQ2N6PD5CVT4A9RWKJ7
V R4QE6JA32X7PZ8L0KSGWYNUC5H9FTD
W XA5847KVREY6GH2U0ZLJ9SCFNPT3DQ
X J5KYQ8NRW7HDF9VZSC3EPUGL0T624A
Y HUZ60TC78PDKR4E3F2X9QLVWGA5JNS
Z CYH37L9SUF2JAV0DT4KGR6Q5PWXNE8
0 SE7GXC8KNZFRDL5PHTQU3964Y2VAWJ
2 VD4JTXQL3WE9U7FN50CR8KKSZAYHGP6
3 26DXPW4FLRJRHSSEKANZV750UQ8Y9C9T
4 Q2VNLKR6D50GYSTEX79AUJ8HWZCPF3
5 KWUVSJA0Z3PC4Y7H6NG89TEFLD2R
6 DL35GA2PTQKZ7N9XRJY40WE8VSUHC
7 80SPK9UJEHTA26XFCLRYDG3VZ4QW5N
8 YSUTNPZE7965VDJLG3WH4F2RCQAXK0
9 PCG4UDFYH6Q0XA8V3RET52WJLKN7SZ

CDEFGHJKLMNPQRTUVWXYZ023456789
A GZYM7TJ9U8QVCR3WKD4560LX2SFPE
C 5K0P6QVM8SHXAE7RDJ42ZF9GLYTUW3
D F5QA59PVLJG8W40YET3RUZX6H7MKC2
E ZJ9WFXAPHT6L3CQ74M2YKU85G0VSDR
F RQPG2AHL94C6UVTZXXJ7Y83EM805S
G 4YT8EMX9A7CPL5JK60FU23VDWSQRHZ
H ERJXD9QPYVW86SUG75Z34MCAK02LF
J 9HEUQ4ZFRG72KMDWT6VA8X30YC5LSP
K 08CF7D563LR4ZJWPSHTV9QEY2AGXUM
L D2S9CJQ0VRAMXGKZHY6F4ETWPU7385
M 863SX2KU75QYJP4DVFAACHLR90EZGTW
N NNNNNNNNNNNNNNNNNNNNNNNNNNNNN
P HFRTLYJSQZX0MW24AUCE6G7893K5VD
Q PDFYVZR2SETK7X5G948HWAUMJ63C0L
R JPH4SGEDFAU537LXYW09MT6KZ8CV2Q
S QLDZ0EF52HY3UTCAGJMPX947RW68KV
T XG4K93UZ9Y60RSVECM5PWL82Q7DFHJA
U 7XW5YC6G482EFSAVKLMQ0DR3PH9ZT
V L52J8RSK0F97TA3EPZWF6GHYXQ4U6MC
W 6U7VG0MTXKL9PDPY2CSE3D5H8RJA4
X W4U0AK7YT3VJQLZ582H6DCSPMFR9G
Y TAG3J64EZWK20H87CQXVM5SULDPR9
Z Y9A6RWGHEX3D5KPM08ST07C24VLQFJ
0 VC5RMF23KDJUY96HQEXLAPZTSG4W78
2 SVLEKHD5PZ64Y89RA7QTJGUFWXM30
3 KM8DULCW6VFGEXQ2PY0JSHZ59AT47
4 UTXCZ8WAGM5HD2903VR7SKLF6QPJEY
5 20VH3PL8CQEWZMJJ5F9USYRA4DTX76K
6 37ML4V8XW0DAHFTS9ZQZKR2PECJ9YGU
7 MW62T534UCSZRQGL0D98PVFJKHEAYX
8 C3KQWS07M2PT9HUFRLRG5EDJAVZY4X6
9 AEZ7PUYRJ4MS08F6X3LGCWVKVT52DQH

CDEFGHJKLMNPQRTUVWXYZ023456789
A 72X3FMCRYLQWKHG9JT5D0NSEZ68UV4
C Z7DTY9L2XNSG8K4H30VJEQ6WM5UFRA
D LCZH2QJA730F56YS9KGM8TEUNWVR4X
E 568YHRWSKGA3Z7T2FXNUD4CJVLM9Q0
F 4GRZ5CYWVXJKQN8L7M029D3HATS6EU
G RVFD8745UAL09MEZXJSY3CNT2QHK6W
H 0TSVNWK3Q8FZA4MG5RD62UY7EXCLJ9
J NLMK7S3CZTEYV5X6H849U0WFQGR2AD
K E06RQG8TSUYMCA94V2J57FXZWDLN3H
L MZJ0XHN7DQ64U8AKTER3WS5G9Vfy2C
M 3JN6C09DLH82GW7ES5YQVKURTF4AXZ
N 9M3EDKQZJS5AFUC80W2TG6V4HRYX7L
P P
Q H9TWJ8SM36VCYFLUEG7045RAK2XDZN
R VF4LWD2UG7M6T35JCNKAQZ9SXH0E8V
S KH0G3U69T5RLXYNFW4ZEA2V2C87DJMQ
T SQHUM50N9EGD2RJV8FCKYWX6A7ZL3
U GWV76AFE5YDHNLC2ZTRMXJ943QS08
V FUGCEXR8W2ZS3J6DALH4N7MQY9T0K5
W V5UXK2G684CTMZ07YDQFJAL3RN9HSE
X CA79RND42JTU6SFCQMHWZK308LE5VGY
Y A42MVLXGRD38SQUNZ9E7HJTKC065WF
Z JDLSATMXC9KRWE20Q6FN5H8V3UG4Y7
0 6SKF9VEQHW4J7235JCNKAQZ9SXH0E8V
2 XYANGJ7F4Z950TV3LQ8CSMH6DKEWUR
3 QN98Z6TLM0WXRVD5KUAHFEGYS427CJ
4 2RYJUZAVFCNEH9WMD36XTLQ07SK85G
5 U8WA0YVKER7QJDSX4C9GL2ZNF3TH6
6 8KE4TF5H0V2NDXQYTGAMWCR7LUZJ39S
7 DXCQ43ZYAMHVEORTNSUL69K5J8WGf2
8 WE52S4U06FX9LCHAR73VZYDMGJNQTK
9 T3Q5LEHJNKU74GZW6VXS8R8F20YACDM

CDEFGHJKLMNPQRTUVWXYZ023456789
A F0DLMUGC49VVKJ7HETR2Z68NY5SP3WX
C 4MX9KZ03HFUSRL2TG8JAV5END76YPW
D GAFEC8ZX0TR3SNMVU7K5JL6P9Y2WH4
E RLV29Y7T8J3FXH5KSWDNCPM064AGZU
F ZDGVXL54AU7WY6CR8N39SEJ2TPKHM0
G 5FZR4E90D8NHPJX7L6WTYVVSU23MCA
H CWMZPF23A96EUY5DTN4LG8R0V7JSK
J Y6SCVHPRN34UGAEXW0T2FMD5KZ98L7
K W23DJ0HSPXGRU56F4ZVMTA9LRCM87NY
L V8EP5SR9U6KDCWZ2J3A7MYH4NX0FGT
M XHC52G4KWDITJV8P9FU60EZL7ARNYSY3
N J76HL3SER2C9D48MKX5YAW0GPFZTUV
P KY20NX36SMDE9G7ACFLW54ZUHT8VRJ
Q QQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
R NV73U268EYHZ0CTWPMGJ4KXDSAF59L
S PJYXRM276W08ZDV4HAUKGCF935TLEN
T 89UJFNLG5RY4W2DS7PXE36KMWXC0AZ
U LT8SG6EZ97P0HKDFYN24VWJ3CRM8AD5
V 7ERKTPNULSWG4M93YHF6X2CAJODZ58
W M3HGYDCPK05NLTSAZ97X8FUV4ER6J2
X 0C4T35AWMG8Y7EKUZLSDR9V6GFNP2H
Y 2SP47CKNJHAL5FR0MD83ZXGTW9UEV6
Z 9G570VTAFL6M2S4NEJHUPRY38KWCXD
0 D4A8HTFMX5E26RWLVHGNUR73ZJYK3C
2 3PKA64WJYCFVTZNDXGEH9058MULR7S
3 HKWFSAMY24Z789JG05RUCDTEXLVN6P
4 AX0UW9DHCZLPNV385EYF7TRJG62KM
5 TZ9NARUDGEJCKY06VSM827PWL3HX4F
6 SNJMEWYV7KXTF0LC342PDHAZ2Y5U8R
7 6RWN8KJLVP5AXUH2CZS034FYD9G9TE
8 EULYZJV5TN2AM3GP6K0RHSWX7C4DF9
9 U5T6D78FZVSX3PAJRYCLKN2HEWM40G

CDEFGHJKLMNPQRTUVWXYZ023456789
A EQSDY43U0ZCH79L5FMPNVJK28WTXG6
C 95MHFTK4DWG30YZX2LEV6PUA7QSN8J
D CJ5T8WMZ4VHSUGNP7XA9Y2L0K6QE3F
E NS3C6D80A49G2VUMJJKXZW57PFTHLQY
F JZ47EKD38M20GSPW9T65XVHYCLUQAN
G YXL32SUTHQ8KDFWNAZ96JE4C05MV7P
H GPXS7QLWT63M48VE0NCYFAZDUJ59K2
J 54D2N7C8FKPAYX3TVHQMLWG69U0SEZ
K 79VLDXW5MPUZS0JYH682AGQ3TENF4C
L UGYNTE6PXAZV5428SFK0D3JMQC97WH
M KC9X4PVJ52LNQUGFTY370H6SVAE8ZD
N Z389QCFaedvypw0k57L4TM2XJHGU6S
P XTHAV0G72UECFNKS635LZQ8JY4DM9W
Q S0AJLFEY685PVMGDZCT3K49WN72HXU
R RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
S 3AE5UJN6QFMXWKYC49H87DVT2PGL0
T H2PQK6XVWYS5Z39AUEDG80N4LFFJCM7
V 0Y6ZHNQXLE4WMDPF3J7AC85KS9V2TG
W WK7Y5G2C9H6FEQDUX0ZTSLANP384JM
X T726MYP9VGVQJNSC0LA4H3UEZ8F5K
Y LHGEWAY2P0N9JZ73Q8MU4SF56DCKVT
Z 6LU8P30HGSF7CJTZE4VQ5ND9AMKW2X
0 A6Q4GZSLUNDTKCXJ852E9FM73VWPHY
2 PWT09UHK7LAD8EMQYSJXN63FGZ45CV
3 8ENM05ZQSJKLT7769DVG2CWH4PXUYUA
4 DFJW3V5NZ9TQLHE2KPOCG7XUMY6AS8
5 MDCPZ29FJ7XE6L8HWGSKUTYQV0A3N4
6 QU0FX8AGY3J295H4NDWSMZCVEK73PL
7 2VWUCLTMKX043A56GQFPPEYS8HNZJD9
8 FNZKAM4S357UH2QVCWYJP9TGDXL60E
9 VMKGGJH7DCTY8A64LPUNWQX0E2S3ZF5

CDEFGHJKLMNPQRTUVWXYZ023456789
A CDEFGHJKLMNPQRTUVWXYZ023456789
C YEJN9F4LAHR83ZKW027M5UQXV6PTGD
D ELA0TVCG94UXZW8FMN6J2HR5YQ3P7K
E JACUK0Y9Dvw752GNHRP4QFZ6M3X8TL
F NOU645WMH2PEK8YXHQ7ARG3TL29DCTV
G 9TK4XJLP8EVQN06MCHZDUYFRAW2537
H FV05JZUYMR6DTPC32XLN8Q7KwG9AE4
J 4CYWLUmDE02T6Q9RFZ8V3N5PHX7GKA
K LG9MPYD7TCH5WFxvJ0QAN4U2ERZ368
L A9DH8METKYF62N704U3CRVWQJZ5XPG
M H4VZER0CYN5976AQW83KFP2XTU8GLDJ
N RUWpV62HF58JLGM73TCZ9XKKAQDEY40
P 8X7EQDT569JWH4ZCLYNGVAMFK0UR23
Q 3Z5KNT6W27LHJAU98DvXCGE4PYM0FR
R ZW280PQFN6G4A9HTXKY5D7LC3EJMVU
S SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS
T K8GY6C9X7AMZUH34EV2LFJ0WdNRQ5P
U WFNXM3RV0Q7C9T4P58E2K6GDZLAJYH
V 0MHQC2FJ4W3L8XE5R69U7LPNGTKDAY
Z 2NR7HXZ0U3TYDKV86GJQZ9F5AC4MF
X 76PAZL8Q3KCNVY2E9JUTMD40GHFWR5
Y MJ4RDNVACFZGX5L2UQTH6W370P8K9E
Z 52QGU83NRP9VCDFFK7LM6ETAIXJ4H0W
0 UHF3YQN4V2XAG7J6ZPDWT589RKLCEM
2 QRZTF75UWXKMELOGP943A8DJ6CYVHN
3 X56LrKp2QTAFA4CWDGVE07Y9JW8MUNZ
4 VYM2AWHEJUQKP3DZN5G0XR68F7T9LC
5 6Q39WGXRZ8D0YENLTAHPJKCM74VFU2
6 P3XD297Z5GEUMJRAKCF84LYHTV0NWQ
7 TP8C5AG3XLYR0MQJD4WKHEVU9FN2Z6
8 G7TJ3EK6PD42FV5YAMR90CHNLUWZQX
9 DKLV74A8GJ03RUPHYF5EWMNZC2Q6XT

CDEFGHJKLMNPQRTUVWXYZ023456789
A 9P6XEYUM2DRGHL3Z7F8S QKW5NVJ4C0
C E43U56MJGPNWX7V SADL0K9YR2ZFQH8
D YN7Q6LK9SRV04JAHF5MXGW83ZCE2PU
E 03MN8U2GC67HRKJJD9YQPZSXLAFWV54
F WRL4Y8QKZ53SPM7CJEUH2G06VA9NDX
G C846HP3VJ0UFYNQ92SRE7ADXMKZLW5
H 5QVMR3JFW42YUAZ0CP789E6NGSDKXL
J G58PW04QVE6ZDULAM9XCN2SY37KRFH
K ZYX5SHRN7W8AE4UJQGPF3VC0LM269D
L QFWCKGHRJENA0Y38MSVP42956UD7Z
M 2E0D GSP439YVFX87UKHARNZ6LQ5JC
N 7SDWAFY6UZHM65PQRVEK8LJCX43029
P 62AK379E0NZ8QFCXDRJUWYLVSH5G4M
Q VWHEZC5RLG079PXM42DJ63AS8UNYKF
R LZFG7JWYXVCU2ED4539Q08MAHP6SNK
S DUNLPR7A9XQE8V2WZH3YJF54KGC06
T TTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
U N9SF2ZDP6K3JH0LXQC75RVG984EMA
V JH50FE8LQCPKS6R23AYGUM9D4N7XZ
W HLQ3X4VZF8MD62KEG0N5ACPUJ9S7YR
X RKZJNVFDYQG6MCS8H4ALE532W0P9U7
Y X7KVUQZSDLJJP3G95W82RCH4MFE0A6N
Z FXR8D5L7KH4903NGVC6WMJEPQ2AUSY
0 PM274NACEUK5LZGMS8H4ALE532W0P9U7
2 AOPYCD63MSXJWR4KNZ59L7FHUQV8GE
3 MCEJS9084ADQZ5YN67W2XUKFPRLHVQ
4 3GC9VAE582SLKDHUPNFMY67Z0XRWQJ
5 8VJ2LMGWH3AXN9FPE6K4S0U7CDYZRQ
6 UA9ZMK50P7F4VWERVJLGNHXQJD58C32
7 KDYH9WXUNF52C86VYLJ0Z4QGER3MPAS
8 4JGAQ2CH5M9R7SW60UZ3DPNKEYXFLV
9 S6UR0XN2AYLC5QMFKW4DVZH87JG3EP

CDEFGHJKLMNPQRTUVWXYZ023456789
A 2CKN06DXV839E5ZPY4RJGF7QTHMSLW
C KVAFZY20WTPRG4LN83MD765HJEQ9SX
D T2J4VP8CKF7ZMEW5NGLYQ3H96RS0XA
E QGHX3ZM47SV6DAPWLCYR20KT9J8FN5
F P6NQJE3TY7RA0SDMG9C4ZHLW5XVK28
G H7E0PLQ359WY2CNXSV8MKZAJRDT6F4
H MEQW40R5GLCFJK3VZA69DX28STYNP7
J 8DT5C3YA2NG0RHV7PEZ6M4QSF9LXWK
K DA2PXFJWCY4SH703659TENGUM8QRLZV
L OSZDQKXM9CT7NYHJA85WF263VP4GER
M 9RC7W5SGH0K8P8D5AX2NLTJV6ZYF34E
N 3FPMTH486G9KXLJRESA50QZV7K5G2DY
P 4N3R8Q5YFES2WZT9HLK7XM0CGVADJ6
Q RHMV5X97EZANT24C0KFSJWDYL86P3G
R SM9AGVLEQX23YJ7KWDPEZ8CTF06N45H
S Z9L2HA0QRVJ5F8EDCT4X6KYPWN37GM
T YJ87A446KDPEX9QCG3H0FR5MLNSZVW2
U UUUUUUUUUUUUUUUUUUUUUUUUUUUUU
V AWC6L8KZXJNM73SFTPQ25Y4EDGHR90
W CXVYSTAL0DFQ5P96JNHK483G27EMRZ
X V0W89JCSZ26H4NRYDFEA3TP7K5GQML
Y F86E27NDT4QVLRKH5MWP5G903ZXCAJ
Z XL0JM2WRSAB8GP6QTKY7VNDJ4C35EH9
0 WZXTRD9V9LKYE3F8M26GCPJNF5A4HQS
2 JKD3WNTVA65LQGX4F7S8HPERY9M9Z0C
3 5P49YM76NHLDV08SQZ2D6WRXAEKJTF
4 735S6RGEFPQZJCYLMD0ZEV9WKHA2T8N
5 G47LF9EN3M0TAW6ZRXJHCSV2QKD8YP
6 NYFHDGJPJ85MCZ92Q7RV3LESX40WAKT
7 E5GZNSHP4RX8KVF09W7QALCXM2JY63
8 6TYGK5F2J3HWSMAE4QXN97RZPLOWCD
9 LRSKECZHMWD46TG2VJ30YA8NXFP57Q

CDEFGHJKLMNPQRTUVWXYZ023456789
A M6Z9U3HGXEJWTF7YRC05SP842NQLKD
C 8WFL30SPY4756GRKQUHTXJDDZNAM2E9
D 6GXRL2AKUHP4FYJ359NZCEWSM8TQ07
E 32RSZFP5WVNDLQATC4G9J807YKUX6H
F SC8KPTJ9WQU2AD370G5N6LXM4ZHERY
G XUDEJ56L7M3NC90RHPTAW2Y8ZFS4QK
H 2M5CXYE6ZP87QTD9SKR4WNJ30LUGA
J K0LZT67NQDHC32SMX5WURAE9GPYF84
K UL7H4ZGQJ6289RN5AEFDP3WXYCST0
L 7JKMNAU40X5FPEH62CG3ZRYD9W8SQ
M 5ZHWD92XA3FK4SGCP8LENYTORQJ7U6
N QT4DCU0F5SK6P5ZWX7A3JHGMEL2R9Y8
P Y3945TW2R80AULHQ5J6C7NKDFGXZME
Q J4068DLSNUZYEHFAGM9K2X537RPWCT
R PE3TM89H2C4XK0ZNFQDYLSJUW7G6A5
S N8TUJK4WFJDRM69GLXEQZ7A50H23PC
T 4SNGW7QC8LX3HAYDK6R0MUZ2J5EP9F
U D7G20HXJKZRTWPQEM3S6Y59FAC8N4L
V VVVVVVVVVVVVVVVVVVVVVVVVVVVVV
W FYCJRQ839NKHUUEL47MSD0GAT6Z52P
X AD63KEZ7G59Q8WLP2Y4MFRCTHSN0JU
Y C9W0E4FRPTLMD72JNKZ8GQU6SXAHS3
Z HAMYGP5D6RCLN8UW3FJ2T9SQE40K7X
0 LQJASXKT4GMWR58ZDHY7E62PU39CFN
2 R5E8AC3ZHYTGJ46SWNUP0FQK9L7DXM
3 9RPNHSY5EFQ67JM480XWKTGLGCUA2Z
4 0NQXFGJ8T7A92MC6UZPL5DHRKE3YWS
5 EH2F6WRAM9SU0NX8YT73QC4LPJKGDZ
6 ZXAP7RMUD2Y0SCK9REWQH83FN5T4JLJG
7 GKU5QMD0LAESY342ZR8X9HPC6WFTNJ
8 TFS79LNYC0GEZXPJUD24AK6HQM5R3W
9 WPYQ2NCE3SJZGK50TLAFU47X8D6MHR

CDEFGHJKLMNPQRTUVWXYZ023456789
A US5E3L7HDYKTRMF4CG62NQXP08V9JZ
C QYLRUPE70GXDJ658MVZ49NT3KFHSA2
D AZSLJY5FH2EVPK9Q04XUMCGR7N86T3
E KQ4V08GYLNT5HJ26R9AM3XFPDZSU7C
F HJ32VUZ69AYN4LX059CE67QGSKMR8T
G PKM9L6NQ4XF2SHCJV37RDTZ58AU0YE
H D3ZYT2S9FUL8GE6C7QRAK04P5MNXVJ
J 39F7X5HVT50PEC82AYMZQULKD4GNR6
K M4GTCVPLE8J7D3Y9XFUNZ6HARS520Q
L ECQ87N42YMV5FTUXP6DKJR9HG3ZA50
M NGPJQTRK30AZLF6H28S9DUX57YC4
N 8PR34JXKMTZCUSEH9DYV5FA2670LQG
P RMNFE984G6HY5DQ3TZ0XAJ57VU2CLK
Q 4LEX2RK0CP6A397VNTSGF8JZMHD5UY
R XN8HKFVGP9DL7A4ZJSC6U350T2YQEM
S 5DAQFCU3Z046NGJJEYKV7PLM82RXT9H
T J695RSF8VZ7GL0NUD2K3CAYEHQ4MPX
U 257KZE0DALMJXNHGQP9Y84R6CVTF3S
V TX6SPZ9N8354Y7MAHUEJ0D2LFCQKGR
W WWWWWWWWWWWWWWWWWWWWWWWWWWWW
X 68VDMHTPRFAE0UGS35Q92Z7CJYL4KN
Y L0CN5MQU2K8Z9VARGXHE7P6F4J3DS7
Z SHDC90AJ37QXM4TL2E85GYKNUPRV6F
0 C2YPAGL574RHTXSNK83Q6MVJE9FZDU
2 Y70MSKCAUEN368DP4ARFLVGX9QTJHZ5
3 ZFH067DTJ5CRKQVYULNS42EMAGP8X9
4 GEK6YXMCQR9UZFO2T8J5PHV3SND72L
5 7AU4HQ2ZSCG98P3KLM20RENVYX6JFD
6 9VATHDJRXHUKC2P5Z74FY5SQ3LEGM8
7 0U2GD4YS5QPFVZRZMENJCXK8TL693HA
8 VRXZG36MNSJQ25KDFALT7HUY90CE4P
9 FTJU8A3X6D2MQYR7S0GHL5C4ZEKPNV

CDEFGHJKLMNPQRTUVWXYZ023456789
A Q37JVWNSGTD52P8R06UFLYECM94KZ
C U94GPS5E28AJMQ63V7YKFNH0ZTWRDL
D PR6504EWJ9ZNGVTKC8QLHUSA237FMY
E 2PL8JK9R6UST7GYV5HMCZ3N4QF0WD
F TNMP3D0AQJLVU9GER28W46CKY5ZSH7
G AH3WZT78SFJ4EDKYMRCQP062NL9U5V
H 3SG0KMAZVNYCPR5WFJ976TDLQE24U8
J DYR4M96TWL57SZFU2KAPVC8GEH3QN0
K 85ZQ9AVCUGFPYT2N3M6SW70RHJDEL4
L 9E2VRZCDP5H0Q3JSGKT478AFUNMWY6
M 0FTEA6W7NR2S5C3LD9VYUP4ZJK8HGQ
N MQF6GRT37YE842HPJLZ0CD95WUKVSA
P L6EZY52JD4VMAHW8USF93KGQC7NT0R
Q F7NDHJMGAWPZCLS6YEKT9R2U0458V3
R 6JDUTCPOY2KQH8M59PZ7ES4V3LGNFW
S GVHT5F3K8QW96JU0NY2ADMRE7PLC4Z
T WM0L7PYQFD9HK4A26CSJ5EU8RZVJG3N
U K45ALGZ2CSQD0FE7HNR8T3MYVW6P9
V H8SMUNG5Z702DY4TQWL3RFJPA6E9CK
W J0Y9NLRFTTP4385QCEUGDZ2KS6VHA7M
X XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Y RWJCF2DM0EUAVKNA4L53689ZHPSG7QT
Z YK8NC7S453MEJ09FATPHYQWDGR6L2U
0 VTW2QEJNM6CGZU79P9ATHRKL5VD83AF
2 CL9SD846EKGWNAHRH30UQV7M5FTYJP
3 7GAY80QVHMRUL6ZJATD4NEWP9F2C5KS
4 5CU3EHKL9V7RTNPNASQJZMGFW80YD62
5 ZUK723894HN6WMLQGFVDV0ATJSYRPEC
6 EDPKWULYRC8F3S0Z4VN2G5H79AQMJTJ
7 NAQRSYFH306K9EVDWP5M2JL4TCUZ8G
8 SZVF4QHUKATLRWCM70EGJNY63DP295
9 42CH6VUPLZ3YF7DG8AW5NSQTKM0JRE

CDEFGHJKLMNPQRTUVWXYZ023456789
A W9MRKJJP87C506VQGSNLUEX4H3YDT2F
C J3SEM VXWU5YFTDA6NH8R04LQG2K79P
D 0HL94FTEYKMGCPVA8WR237UJQSSX5N6
E Y46KG2H5V0FSW9R8T7CDMQAULP3JXN
F 987ST3A2KPKXHVGOJURYMNC5EW46DLQ
G NR5LCHVXSX6TWFQ30Y2M48DK9E7APUJ
H LYD7V8046QAR3WN9KMXTUFP52CJG5E
J RSXYPEGUCVD9Q0WH4L75226T8NKFFAM3
K FQ83LP702MS65XDCWJE9GURVAN4YHT
L TD0AE726H8W5SU4MFFPGQC93XKJRN5V
M PAWG8XUF9SNTY4K5JV036REDCHL2Q7
N 45VTJLEXGHQU98S2DKP670FMYAW3CR
P 3WUN7GC9MX4QD6FVRE2SH5Y0JLTK8A
Q 82KUDWFLTACEGJH3MS47RXPXN95V6Y0
R 5XGD3YNCJE0M82UL6TAVKHQ74F9WPS
S XCJ6W4RP3NH72LMYVDFGTE0K5Q89AU
T Q02WYAKHL7UVXC6P93N8JMSGFR54ED
U CP3V95SAWREKLY74G6QJDNHTX028FM
V EN42X06R5DK3AFJQL8U9Y9T7WHMPCSG
W UMP5FR37AJV2HE8NX4TCYGLSD0QK9
X GJRHU653S4LAKTPDE09NQY2FV87MWC
Y D6HFNKLV29XUM57QAJO8WCT3SRG4
Z ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
0 2LTM69QYDFPNJ3EW7U5KSACR8XGV4H
2 KTOPHMD0934RSYUACVFXWJ57GNE6L
3 SUC4ANJMPG680H9E5YKXLLVD2RTQ7W
4 6VEQRTYGNL8CM7XK0F3HA29PDWUSJ5
5 VGN0SD4JRY2P7KCTHQWFEFL8A69MU3X
6 HEY85QDN4T7JPA6F29S6LWKM30UCXR7
7 AF9J2CMQ8URD45TX3GHWVSN6PEYL0K
8 7KFC0U9TQWJYNRSLSPX6A53G4MVEHD2
9 M7AXQSWKFF3GLENERC5DP4JVYU6H0T8

CDEFGHJKLMNPQRTUVWXYZ023456789
A 8MGWQP60RK4DZNSJ52FHU9LT73EVXC
C M0R5V7AXP69TNDJQHW3G2K4U8LZEFS
D XFGVZMS3EC62Q7U8RH4P5AKW09JNL
E DTFMLCVU3PGJK6NA08WX7RHQS5942Z
F VEDAURXZT0845HLGC6J5K79PQW2N3
G 6A8J09HCM52V3LP4QND7ZWUEKTFXSR
H K67NM45A8WUPF3RLJZSQE2TV9DX0CG
J 2W4FKTN59ZV8CS7D3XGL0EPMURA6HQ
K Q75UGN98H43CVEAZ2T0WDLFSJXPRM6
L ZNUSWV3J2F0KGR9PDC7TAXM6E8H5Q4
M GRA9S58PC7JFU2XWK4E6LQN3HZTDV0
N U2LX9DZ4EP7ACQSF0H3MVR8TG6K5J
P CS07F6RDXG5Z49EK8QUMJHWNAL3TV
Q W5936UJHKNEMSD8TLFR4XZV02PCAG7
R ACMQXKGS0HWEL4V97JT8N52Z6U3FDP
S 0XPHE8CFVAKUJQT7G5LRW692M4NZ3D
T F3ERN0DLZSAW782MPG9VHC65XKQJ4U
U 3LZPJXT4NDC58MW0VRKEGSAHF67Q92
V SDX83APTFRHN9KZ6M720QG5JCW4LUE
W 49JE732KQUDG0XHFZVANPTSRLCM865
X PVS6TG0EDM7LW53HAKNC98Q4RJ2UZ
Y YYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
Z TU304SE2LVRQ6AJCXM5F8PG7DHK9WN
0 RPKDHMVS8Q32WF53HAKNC98Q4RJ2UZ
2 L4NVQFU9JTSHM05XEP6ZRDCG3A87KW
3 EZTC2PFNUXM9HG4RSAQD608KV75WJL
4 NJ2D5ELQW3X6RKPVT8SUCF0AZMGH79
5 9KQZ8LW672TRXFG3NECJVUDP4S0MAH
6 78H2RJKMG9LSEZCNWUX5T43DQFVPOA
7 5HKL2Q6GJZ0DTMU43P9FNEXWVSCR8
8 HG64CW7RAQNXTU029LVK3JZF5EDSPM
9 JQWTHZ475LFAFV6EUDM2S3XCNR0G8K

CDEFGHJKLMNPQRTUVWXYZ023456789
A T4LHRCWVU76N8JKFPQY9X5ZMED2GS3
C HP5DWF97J24KNTMZUY6S8GVQR3ALE
D 28M3QRYJXS459L6UKACNVWT7GEPHFZ
E P97ZAQCLWF8VNM4X6DR5YJ2SHGK3UT
F QH6Y7MJP58GT3KENDSLV2R4XUAW9C
G KNSTDARMYU9J578W4EQVCLPF3H6ZX2
H 65F2EDQ7CXNLVS9Y8GAJRMKUZ34TWP
J XQZS5N8H62REA3CKYV9A47GUTMLW7PF
K VSR92T3YGA7XFCMELPZUHW5Q46J8DN
L WATFV5934PQGDZR6CJNE8HX27MYSKU
M YD2UJVNZ8KAHETQ4RL5G93WPS7CF6X
N SYGL84KD23WQCEXTU96RPA7HV5FJZM
P 57C8TZHWEQMUSYLDJ23FGXNR6KV4A9
Q Z6JGCYX5FMK94VPS2RW8UN3LDATE7H
R 3KVEYWUNSLP86527TCX4F9HJAZQDMG
S RGKMWLV2N4EZHPD9A7J35TC6UFQX8Y
T 9LW63HEUACJSMXVQ5ZG7DF8Y2NKR4
U A34CS7LKV9H2Z6G5EFMTJQP8WXDYNR
V UR37N94GKTCQDQHYPV58A6EFZLJXM2S
W ET9QUF74L5ZK283JHXSMP6WPS7CF6X
X DZ8RFSM6JN3PT4HVGU72LKA9YWE5Q
Y G2NAXUS8MVT6P9ZL3WFK74E5RCHQJD
Z 8JXKHGDFQYV7LU5RN3EMAS4W2T9PC6
0 000000000000000000000000000000
2 NM4Z3GXDRLF7WJAVTHSEU9CKP56Q8
3 4VUPGEASRW5MJFNC9HDLQ76XTZ82YK
4 LUA5KPTR3EFYXQSH762WZCJD98MNGV
5 FCHM986EPZYARGW2GMK4QKDS3JVL7
6 JFQNP2ZCHDSWUR72XNKT3YVA8L9E5
7 CEPXLJ5T96D3G2A8QMVHNZYKFSRU4W
8 MXDV6K2QZGUCWAF3S4PYPTRLLEN975HJ
9 7WEJ46PATHXRYDUZ8K2QMG5NSV3L

CDEFGHJKLMNPQRTUVWXYZ023456789
A JY3GTRVQMH0SNKW78PEL4ZDF9X5C6U
C WS6JXATFEMLKZHNVY9PQR387UG0D45
D NK4WGCX7PEQH3MZTSU9FA6YV5JL8R0
E C5SA4M6NFQJ08LD3UV7WHY9ZTRGPKX
F EX5MKQSDNWAG9JPYT3ZCLUV86HR704
G 0Z759XPHR4S3Q6LENC AKTFWMDUYJV8
H R9843KZJL0XUC5ANPFGQSDEW76TMYV
J L3V0UG9MARK6F4QPZDCHX7NE85SWTY
K 4PD6ZSNG05T9AURWQLXYCMJF3VH87
L HV9KY08AJG4TEXMD7NWR5PFCZS6QU3
M AU9R6H3WQLG5D0CZ97FJK8PNV4XEST
N F4XQ0W5PDCMRVA7U6Y8EJT39SLHZGK
P D0KCRE4Z7FWLYQ865TVNMSU3XAJ9HG
Q MTUHSLYC WJRXPGE8VZNA097D3K4F56
R G8ZXV47LHK5YWSJHCFDE06NCQPTUA39
S 6EC3NYWX5UUVPR94JML0T8AHGQZ7KDF
T UWQ9EVM5638N0Z5HJR4Y7LGKAPDXFC
U YQM8C9A6TVZFK7SRLGX3PH04JDN5EW
V 9JLPM7HY3ZDW5NUKG468F0XSRECTQA
W Q6TL5JUECAH47RF938DMGVZPY0KNXS
X 5NFUPT EK46YZL30MWARSVQJHC98G7D
Y 3MAZ8W8JTU97E4P6GH05VDRKXLFSCQ
Z 7RGFLN098DEATCV54SY PWX6UKQM3JH
0 K7P85DRGX6VMTJHCFWJ4UEQANY3L9Z
2 222222222222222222222222222222
3 VAJ7QZLUY8PCXDYTORKS9NG45HFE6WM
4 XDNT76F0KSU8XJYGQCMH53WALEV9RZP
5 SFEYDUC4XT37HVKAQJG69MLRW8Z0PN
6 TCWVF3Q5SY9DG8XL AHKUJZR0M7P4NE
7 PG0EHFK8ZNCJ UW9SX63DQ5TY4MAVLR
8 ZHRNJJDGV9PFM6E3XK5U7C4ST0WQYAL
9 8LHDAPR3V7NQSFY40XTZEK56GCWUMJ

CDEFGHJKLMNPQRTUVWXYZ023456789
A 0GJ86SZDWPLEY25Q4XCTHMFVVR7UN9K
C NUEAHW9YVD0L4TFJKZQ5PGRM6S8X27
D W6KPE TGN50S7X8C4ZMYALRQFJ2HVU9
E 9CNJYG57UKZXS RH0WTL64AP8DMQ2FV
F Y2H58N7QXC DPJWM6EK RVA9GZU0T4SL
G JWAM5406KRQCHX98PLUZFS27TYVEND
H 7FY6Q9VL2EK40G8DNWPUJ5ATCZRSMX
J ZA0QDMTKG4XN7F6LS2ERY8HUPVC95W
K GDW4NRA96ZMV2QEST87JXPLH0FYUC5
L 2QXE4UFS879ZW6PNV50HKCDA YGJTRM
M QS8VTYLR4FCA6NZUHEGX579K2DWJ0P
N 5E907A6VCWT2MPY2QEST87JXPLH0FYUC5
P SR4HJ2M0TL7KNUAYXVD8EFC5Q96WGZ
Q X8LCPV24MYN0K5RE79JF DU6GHWAZTS
R 4TPFAXS JZQYDEVHXL76MC2U98N5KW0
S 84M7ZHQT P2UG5E0VFCWL9YNDX6KAJR
T PZR2GL4A08H6C7WFQY5SUXVNME9DKJ
U EVC GFKNH76JQPZ2AD089RUWTS54MLXY
V C7UW2DEFY5A8R0XG6JMN TKZ49PSQLH
W AKGS9PJ5DT8UF LNMQRV024XYZH7CE6
X FL2NSCHMQV5TGD49U6ZGRXDSJ4QK8LFHU
Y VH7DL5UXFNWSZAQK9G4C06JRETPM82
Z R0TXWQPGJMF5UYK28H94VL7ESC N6DA
0 TJZLKR8RWAS29VHDXMFPN7QYC4UE56G
2 HXF9MEY8LU6RAKS5CDT7GNW0VJZP4Q
3 333333333333333333333333333333
4 MPSY0F8ZRXVW9CJ72UKQJH66L5DGT
5 D96TU0KCNAPHQSVR J4FW8ZMXGL2Y7E
6 K5DR CZWE9J4YLMUP05SHGQT82AXF7VN
7 UYVXX6C2H9GMTJLW5ASEZD0PNR48QF
8 LMQR7XP SHEJD9TCYNA26V5WFKG0Z4
9 6N5ZVJDU EGRF847TAP2KM0SLWQXHYC

CDEFGHJKLMNPQRTUVWXYZ023456789
A PUHKQVS6CJXZ58NT20M73E9RFLYWDG
C AGUVDJHN67KY9XWFEFTRP8205S3LMZQ
D QWNSRH6YZUF9K2L7AJ8GEPVXCT035M
E TVKR75XF29MAQWSY3ZU0NLDG86CHPJ
F 27J9A0VHST56ZRUX3LQEM8YDKWNGCP
G U6C2WFADQSER8TZV7KLH0JX3P95YMN
H SAPT6E7GU20WL9QXV8ZF5K3YJRM DNC
J VF23H8EP7XLGWYA50R6KZ9MNTDQCUS
K XETYFLOJ V3ZH6D7M5WA8QRNC9GUPS2
L Y5RG0QM83DUE7HXCNAVZS6PJWF2KT9
M WYZA3CD5R6PX279HGSTNJUFEGVK08L
N 6DQ7YPGMWAJ3TVRFH29CKSE0UX85LZ
P 7HSXGKFC AV8DR360E9WJLT5M2YZNQ
Q GN6FMS CZDH25XEYJ PV3UT7K8A09LRW
R MLYC86Z95NAKFP0UQHEW7GS2DJV TX3
S FP70CTJUHE9NY5G8K3D2RXLZVMWQ6A
T 0KX MJR82E5WPGNFZLDH96YQU3CAS7V
U HCAEN2PQGFTM30DKJXYS9V8L75RZW6
V KZELS3T7J8YUNZPR5MFCXD5W60QGAH
W NYZAPL AQRMC78EJ5SUF06VH2TGKX93Y
X 8T0Z2Y9VKLDS CQJWRNP3GM6A5UH7FE
Y ZRMU9GW3LQHTJVS8A6PKDFC7VN2EX05
Z DMWH5UNLYGSOFTV3PC7XQ2AJK6ET89R
0 9X8WVM3ETRN7U62DYQS5CZGHLAPFJK
2 EJV5P9KSF0RCDMHL8YGTW3ZQXN6UA7
3 L95QTD RX8ZG2P UK6WCJYHNA7MSFVE0
4 444444444444444444444444444444
5 R3L6XNY09WCVS ATGDU2MPQH FZ7JEK8
6 CQGJZ7UWNPVLOKM2SE5AXFT9H83RYD
7 JSF8UX2APK3QMLC9T5NVY0RWEZD6GH
8 309DEZ5KXYQFAGVNM67LUWCPRHSJ2T
9 583NKWLT0M6JHCEQZGFRADUSYP72VX

CDEFGHJKLMNPQRTUVWXYZ023456789
A KEUCJTQPFPG26SWVHN893YDX7L4ZR0M
C 2DHZL48GTY6UP039SRAWJ7VFXMEKQN
D HTC73S2LNX9AJRQZY6E8V40MWPFUKG
E UFADVNKJMLH9Y80CZ2QZXTW43S76RP
F AMET0GUV P3CZX2RDL7KWN8SWQY496J
G TLNYUWDK324MRZAS87PC6X9VH0JFEQ
H 8C29G7W NDSRKMVL640UXPZJ EYFAQ3T
J MVPL9QF60UNSDZGKTYEH3CWA8X47R
K 3UQ2MCLFATW07YP8DXRG4HS9NZ6VJE
L N3GXA8TUQHSP67EY24JD9WZ0CRVMFK
M EFPNRLA0JQD7WH6T3C4U8G2YKXSZ9V
N DGTSKXCQL87F09U4WZMHR Y6J2VPEA3
P FJMG63ERVKT48C9NQDSA2LHXUWY7Z0
Q LK38FHGEUDXVZSMWCY0N7246T9RJPA
R V60K4AJ79F3WDG SQEL8PTUNHMC2XYZ
S 7Y4P2VZ8XRFTQA HM0EN9KJUL63GDCW
T CND4QYH3GWZEV6K7X9F20SRP8JMAUL
U QAKHPD3MEN8R4XJ2TW6LSCYZG790VF
V P0J3ZKM9RAGYHT7LUNXFCQD8E2UQ46
W Y8X0D6SC2ZJLAMT7V9P34ERF7WQGNH
X SWYVCR4H89PGUFDJ6ML7A0EQZK3NT2
Y 4XSJH072W6MNKECPRFGZUVA39QLTD8
Z 679EXMRY4JUHGWAPKC0LF3TVND28S
0 JRVQ7UPZ6ELXC N43AGWMDKT2FH8YS9
2 WH86NZXT C40QFJGR7VKY M9PASEU3LD
3 GQLWE2NAKCYJ94FXHSVTZ87RDE60PMU
4 ZS7M8J9WY0ED3U2FVAT6QPKGR LNCX
5 555555555555555555555555555555
6 09RUSEV4ZMQ8TLYKFC32JNAGCPDHWX7
7 94ZFWP6XSVACLK8EJUDR3MQN0GTH2Y
8 X2WRT9YDH7V3EPN0ZJQS F6MU4AKLGC
9 RZ6AYF0S7PK2N3XUMQHVGELDJTC8W4

CDEFGHJKLMNPQRTUVWXYZ023456789
A RX97CGKW3FS8PTJMQV4ED25UYZ0HNL
C LJWY4K38N9QPS20FRGD XME7TAV5ZUH
D Z5PCMNUST8LQRX7WH3F09JAE4KYG2V
E FKYTXSQAR7DC4Z35MPJG0VUH28NWL9
F GYQD9T2RESZLH0APVUW7854JMNC3XK
G 7Q2ZKDMFEFT0XJ8RU543SNPHWVCLA9Y
H J8NRZACU43ET2FPKXYVWG9QML7S5D0
J WNCE0RL4HAFDMGUY9Q537K2VXSTPZ8
K YREV3MFX925J0PLT7DNQUSZ8G4HCWA
L XW3QHYANCK2UTM8GE7Z9VFSDR5P04J
M V7S4FUTQ2PHRLJY8ZN95W0CXD3AKEG
N CHJKU9W08XY57QZEAFTL2RGS3MVDPA
P UDZWSJ0V5H3GKAMLNXQ4RC9Y8EF27T
Q 2FGPR57KYVU3N49ZT0LMHD8CSJWXA
R E9KSL7Y3AGTNUDWV25HFZMP4Q08JCX
S TMV8Q05G7ZNK3CFHURJDL4WAPX9EY2
T DV5N28P7S0CYALGJ4WEZXXH3RU9KFMQ
U 4Z03TW85PJA7YRVXC92HELKQNFMSD
V 5STHG4D2MUJEXWQN0CKP38L9ZARYF7
W 3CLF8EXHJRGZV74QK2PASYM59TDU0N
X 93A2JQRCLYM4DVN7FS0K5GTZEPU8HW
Y QEF5AVG9KMPW8UXDSZC24T0N7HJL3R
Z 0PULVC4TDNX2E9S3JAG8KWRFFHYQ7M5
0 8U4X5LHDZC9MFKTAWR7NY3EGJ2SV
2 MG7UEPSYQ54ACHK0D8XVJZNLTW39RF
3 ALXGNF9JWE70S5H2YMURTQVPKDZ48C
4 H08AD3NPUWRSQE59LKMJFFXY2CGV7TZ
5 PTDJ7HZMV4WF932C8LYUANXK0REQGS
6 666666666666666666666666666666
7 S2M0YZVFGD89WNE4PHATCUJ35LXRKQ
8 N4H9PXJZ0LKVGYDR3ES0QAF7W2MT5U
9 KARMW2ELXQVHZ5CSGT8YP7D0FU4NJ3

CDEFGHJKLMNPQRTUVWXYZ023456789
A DJQ0Z5YE86FU3X4SL9KVTGCNWRH2MP
C 3XT402RS9KE6Q8GZUPWNAMJYHVDL5F
D VQYFP0W8A5XMNTE9GC26RS3KLHU4ZJ
E 8ZP6UVJ24D5R90KLYG3AFWSCQXTNHM
F XS9ULHC50RMY8Z62N4DTPKEA3JQVWG
G S50NV3FWLCKAZ2YHTUJ94RMPXE8QD6
H UNKJCP2TRZQS6YXAED0MW8V54LGF93
J Q8AG4LDZPWSKT9M06FHYC5XRV3NU2E
K 2HUAT8M3NEDFLVCQPYS46JWGZ509XR
L G653DC0YW9N8MKQRXHP52TUZF4EJAV
M Z24YNQEHUJWC0LRVA6XP6D5F8S9T3K
N KRH8XEUC34A0WD9JZQG2VPYLM65SFT
P JE8L2WAMZYGNXSU5V0RQ96FTDC3HK4
Q YADSEGVPLJ92RCZF5XUW30TH6NKM48
R H3NP9ZKXTMJGVQF84A5UYED62WL0SC
S 90FK6NXLG32DP4WURMQCEHZJT8AYV5
T RC3ZSMNF XUPLDJ0E286HQ4AVKYW5G9
U MK2Q3J4RHPY95WTD28VFLA60EGSCN
V 6YWXJFLAD0TZKR8CS345H9N2GUMEPQ
W LV6CA95QYS3EUNJTFRZGKXHM024P8D
X T9CMGU30FHZWAP54KEVRJ2VPLM65L
Y WDV98S6JQGC4H3PX0TMLNFRU5K2ZEA
Z P4EWKY8UMQL3FGH6D5TJSV0XA9CRN2
0 FGSWHR965TUQEMVK32AXZND48CPJDYL
2 4UMDRAZLNK8VXG63YJW9E5QLSP0FCTH
3 NTRF4H9C285YASPMJLKDZQWUV6G0X
4 EMZVHDPK2A6TS5NWQLC80YGJFX3RU
5 0LGRYTSV6XHXJ4UDNCK8FM32E9ZPAQW
6 5WLTXGDFVRFP2HA39NE0UCK4SMZ8JY
7 777777777777777777777777777777
8 APJ5M6Q4EV0HCF2GWSNDXL93YTRKUZ
9 CFX25KGTGSN4VJELMHZY38UPQRADW60

CDEFGHJKLMNPQRTUVWXYZ023456789
A H54U7FRTE3PVWGCLKJZX2S69D0NMQY
C TQ7XNHYZMA32LF9D4K6EURSJ5PGW0V
D KX36AJF4RLWTV952P07SZGNQUMCYEH
E AY5FQ34CTXUN6PMSDL9HGKJWR20ZV7
F UKS0R2LX3GN5CVH96ZEPQWMTJ7YA4D
G 2J6QSVWUPN7DAYFCZTX05MEH94R3KL
H X4RPYUDEAFGQ92TJS6M30LWZKNVC75
J 6PGMFZ2SL9CX5TKQN7RWEVY40AHD3U
K S3FWH6URDJ9EQZ40GNLYM2V7PCT5AX
L JUPZ39GKSWMHYCDV0Q46TN752EARXF
M CVQH0A79ZEXGS3WR5DJTF4KLYUP62N
N V9Z56YM2074L3RGATHUQDEXFCKSPJW
P GLJVKN6FU0QRE73M9CH2YZTAW54XDS
Q 7MCR94TNV5D6UK0XA3GYSHFPELJ2WZ
R DHXJEL354S6CNWYGU2QK9P0VFM7TA
S LFU9XWPDK6ZA7MRN2V5JC0QYGTAE4H3
T E7Y3VX5MCHF0JUZKRSWAPDL64G29NQ
U PSLND0J3F2V4TQXZWMAG79CE6Y5HRK
V QZM4W5C0NYRJFD2HEXP7KA3UTSLG69
W 920TPCNJ6MEFRALYQ5KZH74DVX3SUG
X 3RDG5PKAHU27Z0E6LWCFNJ9MSVQTY4
Y 5TEKMDAQ7RS9GLVFXU04J3P2H6WNZC
Z MNVA2EQW9THPKX64YRCL35DS7FUJG0
0 NW9YJ7ZG2Q5SX4PECAFVRTH3MDKUL6
2 06W7LQ9PGVYKH5UTME3N4CAXZRDIFSJ
3 FDK24GSHXP0YMNAWJ9TUV6ZCLQ7E5R
4 RAHLTSXY5KJM067PFGVDWU2N39ZQCE
5 4EASCKH7YDLZ2JQU3PNR6FG0XW9VMT
6 WG2CUM0LJZT34ES7VYD9AQ5RNHXKFP
7 YCTDZREVQ4KWPSN3HF25LXUGAJ609M
8 888888888888888888888888888888
9 ZONEGTV6WCAUDHJ574SMXYRKQ3FLP2

CDEFGHJKLMNPQRTUVWXYZ023456789
A S7RQN825JVE4X60WDZHCFL3UKGMLPYT
C V2G67EDASQFNHMJUWX08LZKPK34Y5TR
D 3V8XSUQ2GZKJYHRM6LTWPF4NE057AC
E 6WZ4DLU8QMY2JNVPK0SFTH5AX7RCG3
F MUXNWKYKE64TDS7Q5PJVLROACH2R8G3Z
G DCVUAZ8R2WX54K7FEMN3H6LYQP0TJS
H PL4AFJYXK5SEDCURT2W0V7G3N8QZ6M
J AT78YVR05CQLUEP3GWKS6DZX2FMH4N
K HMLJ654UX0AQQGSZ7NR3PCT2DYV8WEF
L 4KH7UTPFMNRWV26A5SQYGC80D3EZX
M LKXTZNH6FY73CREJ0A8425SVPGDQWU
N T05GH2J4YRDXE3LVS8F7WCQ6AZUMKP
P 04YSMANKHJC63VX27GZ58RDWTQEUFLL
Q E3WLGZV8F4R5YCHXPA6NK0JUT7S2D
R 2ASW53CT7DZPMUNE864GXQFLVKHY0J
S CR2ETQGJA86YKF5Z3QUPMWWXHDL40N7
T 75JDPGAYN23K6W48CQPMRZVEFSUXLH0
U X6F0QPMWZH5VVRJ3N4TGKAY72LSCD8E
V 8GDFR63SCEMTPLAXZK5Q4UH0WYNJ72
W ZQEHVK6D3XPST0G4MYRU5LH7FJA2C8
X KFM5E0LZUPJ82AWTY7DHSNRG4CV3Q6
Y NP02KR5L47GUQDMCAV6T3S8EJWZFXH
Z UE6P8HF3WK0C75DYLN2XJ4TRMASGVQ
0 5YNCLSTHPAVFW8KGRDUJQ23Z7E6XM4
2 GSCZJWV7R3U0LXT6QFYD KEM48HPN5A
3 W8QKXEGDUHANP2LF47Z0MYT65JRSV
4 YHPRX70MLT2Z8GFSJCENDAVQ53W6UK
5 JNTV4C7P0S8MZQH23XAEJGWR6FKLY
6 FZUY34XQELNGAT80H5CM7PJSKR2VDW
7 RJA30DSNTGWHFZYQVEL2U86MCXK4P5
8 QD3M2FWCV6L704SKUHJEYXP5ZNTARG
9 999999999999999999999999999999

Module 2: Extra Share Generation Tables

The main instructional section contains share derivation tables for k values of two or three, assuming that initially generated shares are A, C, and (sometimes) D. This page provides tables for higher k values; the next provides tables for the case where your S share is an initial one.

Even higher values of k can be obtained by editing the PostScript source of this document. Search for the text EDITME to find the right section.

We caution users that higher k values, in our view, are a bad trade-off between usability and robustness (which are damaged) and security (which is improved).

k = 4	F	G	H	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
A	Δ	Σ	†	#	†	¶	†	Γ	α	†	Π	β	Δ	€	¶	Δ	§	Ψ	Ψ	ε	Ψ	μ	@	ρ	Γ	ç	¥	Γ
C	Ξ	Ω	†	Π	ε	♦	Γ	Σ	Π	§	Γ	Φ	ρ	♦	Δ	Ψ	§	η	¶	§	♦	Γ	Ξ	♥	Π	Λ	†	Θ
D	♦	Ξ	Π	†	ç	η	β	¥	♦	Δ	Σ	%	¥	β	€	Φ	Ψ	Ψ	¥	ρ	€	†	#	♦	€	Ψ	§	Θ
E	Ω	ç	Π	Δ	Λ	Δ	§	μ	Ξ	α	η	Θ	♥	Ξ	Σ	†	Ω	ε	ρ	¥	α	ç	Ξ	ρ	♦	ρ	Ω	Δ

k = 5	G	H	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
A	§	€	%	¶	@	ε	Φ	€	β	ε	Γ	%	Ψ	ρ	♥	Λ	♦	¥	¥	†	Π	¥	α	ç	Ψ	♥	†
C	¥	@	#	#	β	β	@	Ω	Ψ	@	Ψ	β	¥	¥	α	#	Ψ	#	β	α	Ω	Ω	Ψ	α	α	@	Ω
D	§	Ξ	Λ	¥	Π	Θ	@	ç	¥	†	ε	Ξ	Ω	Π	¥	ç	¶	§	♥	Γ	Σ	α	†	μ	Σ	β	α
E	¶	§	♦	€	♥	Ξ	Γ	Ω	Γ	Π	α	α	Ψ	Π	@	α	Θ	μ	ρ	¥	Ξ	ε	♦	Θ	§	ç	¶
F	¶	#	Ω	Θ	β	¥	μ	β	€	α	Ω	Δ	Ω	ρ	Φ	Π	Ξ	Δ	Ω	μ	Θ	ç	Π	Δ	¥	†	Σ

k = 6	H	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9	
A	Δ	Ω	†	β	♦	α	ç	Ψ	Δ	@	¥	€	¥	η	ε	ε	Π	¶	ρ	€	¥	Δ	@	Θ	Θ	ç	
C	η	β	Θ	¥	♥	€	Ξ	Θ	Δ	α	§	Ξ	μ	♦	μ	Π	Ψ	α	μ	†	%	§	Θ	§	ç	Δ	
D	Θ	Π	ε	¶	¥	Δ	€	@	α	Ψ	♦	¥	€	Θ	†	Δ	Ω	β	@	ç	ç	ε	ρ	η	Δ	¥	
E	†	Ξ	†	Π	Φ	¥	Λ	Φ	α	§	@	¶	§	¥	%	¶	Γ	Ψ	Θ	†	%	Φ	%	Ξ	%	η	
F	%	Γ	%	Ψ	@	α	†	§	¥	Φ	Φ	§	¶	Ξ	ε	†	Φ	Ξ	Π	%	Λ	η	¶	Θ	¥	†	%
G	ç	Ψ	μ	α	§	Δ	†	α	€	Θ	♥	μ	Ξ	§	Θ	§	β	¥	Θ	Ξ	Δ	Π	μ	♦	η	%	

k = 7	J	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
A	§	#	†	%	€	Φ	¥	μ	%	Ξ	Θ	α	Φ	Φ	¥	¥	Δ	Θ	ρ	%	ç	ε	Ω	§	€
C	¥	ρ	Σ	Ω	♦	Φ	ç	Π	Δ	@	#	@	†	♦	ε	@	ρ	¥	♦	♥	¥	%	Θ	@	Σ
D	Λ	ρ	Ψ	†	η	¶	†	†	Θ	Δ	β	α	Δ	Ω	%	%	Σ	ρ	♥	@	€	@	†	§	Ω
E	¥	Π	¥	Ψ	@	¶	¶	§	Σ	ρ	♦	♥	η	α	¥	Σ	μ	♥	Φ	†	Γ	β	†	Φ	Λ
F	Θ	Θ	Ξ	#	ρ	Θ	Γ	β	€	Σ	ρ	Π	α	€	¶	¥	€	¥	†	Φ	Φ	μ	♥	Φ	§
G	♦	¶	ε	♥	#	ε	Θ	¥	Λ	§	Ψ	Ω	η	Σ	β	ρ	Δ	Σ	€	¶	#	Π	Δ	@	Π
H	#	†	♥	α	β	Φ	Θ	β	Θ	α	Ξ	Ξ	@	†	ρ	#	♥	Γ	Φ	Ψ	ρ	Γ	@	¥	Ψ

k = 8	K	L	M	N	P	Q	R	S	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
A	α	ε	§	#	#	η	♦	♥	Γ	%	Θ	§	%	♦	§	Ξ	§	#	#	α	ρ	%	Ξ	¥
C	Φ	α	α	β	Σ	€	♦	¥	Ξ	Ψ	♥	β	ç	ç	♦	Θ	ε	Δ	ç	ç	Ψ	Λ	€	ε
D	♥	η	§	Γ	%	Λ	ç	%	Ξ	α	§	α	Ψ	@	β	♦	Ψ	β	†	Π	♥	ε	α	Ξ
E	Σ	†	β	Φ	§	♥	Σ	μ	%	¶	@	♥	Λ	Δ	α	♦	Φ	%	Ψ	η	η	μ	μ	¥
F	α	β	¥	†	Ω	♥	♥	Θ	η	Φ	¶	@	α	α	♦	Φ	Ω	Θ	β	ρ	ç	α	Λ	Ξ
G	Θ	β	α	Γ	α	Π	α	μ	Θ	†	μ	ρ	β	Γ	Θ	μ	α	¥	Σ	α	Φ	Λ	Φ	μ
H	ç	%	¶	α	§	¥	Φ	η	η	€	#	Γ	#	@	α	η	Δ	μ	Ω	†	Ω	Π	Δ	
J	α	Π	#	¶	¥	α	@	Γ	Θ	β	ρ	α	Ξ	#	Π	%	Ψ	†	μ	β	Γ	Ψ	Δ	Θ

These tables allow you to generate shares in the case that your S share is an initial share. However, in some cases, you need to generate the S share first, when using an existing seed with this scheme.

k = 2	C	D	E	F	G	H	J	K	L	M	N	P	Q	R	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
S	€	Δ	Ξ	Σ	@	η	Ψ	†	Ω	†	Π	α	Γ	Λ	♥	§	⊕	⊕	♦	Φ	¢	%	β	¶	¥	μ	ε	ρ	α	#
A	¥	Γ	μ	Φ	#	ε	Ω	†	Ψ	†	ρ	⊕	Δ	⊕	♦	¶	α	Λ	♥	Σ	%	¢	α	§	€	Ξ	η	Π	β	@

k = 3	D	E	F	G	H	J	K	L	M	N	P	Q	R	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
S	ρ	Σ	β	ρ	%	%	†	¢	†	†	α	Π	€	¥	Σ	β	¥	€	α	⊕	α	α	Φ	¥	Φ	¢	†	⊕	Π
A	⊕	α	†	α	μ	Ξ	⊕	μ	Λ	Λ	†	α	Ω	Ω	β	†	Ψ	Ψ	†	#	@	#	α	¥	β	Ξ	⊕	@	⊕
C	¢	ρ	α	%	♥	♦	#	♦	#	@	⊕	¢	η	ε	Π	⊕	η	ε	α	Ξ	Ξ	μ	Π	¥	ρ	♥	@	μ	%

k = 4	E	F	G	H	J	K	L	M	N	P	Q	R	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9
S	¢	ρ	Ξ	μ	#	⊕	#	@	Φ	†	β	Ψ	Γ	†	Λ	α	ρ	Δ	♥	¶	%	Ξ	†	♥	Ω	♥	ρ	#
A	¥	@	#	Γ	Φ	⊕	¥	Δ	¢	♥	♥	¢	Λ	Ψ	ε	β	Ξ	♥	α	⊕	Δ	€	¢	Δ	♦	¶	¥	α
C	§	@	β	ε	†	Π	Λ	§	Λ	€	ρ	α	α	Δ	Λ	Σ	Δ	¶	§	β	ρ	¥	Ψ	♦	Δ	⊕	β	♥
D	♦	Π	†	Λ	ε	ρ	†	μ	@	#	Ψ	ε	%	Σ	η	Σ	α	ε	†	€	¶	Ξ	♦	Δ	Ξ	Ξ	Σ	¢

k = 5	F	G	H	J	K	L	M	N	P	Q	R	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9	
S	@	ε	Π	¥	§	♥	Γ	Ψ	♥	¥	ρ	⊕	¥	¢	α	η	α	⊕	¥	Ψ	§	†	η	⊕	@	β	μ	†
A	†	Γ	Σ	Ψ	¥	¢	¥	¶	Ψ	Λ	†	¶	%	Ξ	Γ	¢	¥	⊕	Δ	Λ	η	¥	Γ	§	μ	#	†	
C	#	†	ρ	β	¥	♥	Ω	⊕	ρ	Ω	¶	Σ	Ω	⊕	¢	α	⊕	♦	#	α	%	♥	†	Φ	Ω	Ξ	¢	
D	€	⊕	μ	Φ	†	§	¥	Λ	Δ	Σ	§	@	€	Λ	β	η	Ω	Δ	⊕	♥	%	μ	Λ	#	⊕	Λ	ρ	
E	ρ	Ω	Γ	§	ε	⊕	β	ρ	ε	¥	⊕	@	Φ	Ξ	Π	¶	†	Φ	♥	†	⊕	Ψ	Λ	Ω	#	Λ		

k = 6	G	H	J	K	L	M	N	P	Q	R	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9	
S	Δ	#	α	Ξ	μ	€	Ω	Λ	β	⊕	α	Ξ	Σ	€	ε	€	Φ	Δ	¥	Σ	Ω	¢	Ξ	¶	†	¥	
A	⊕	Λ	§	α	#	⊕	⊕	†	Φ	⊕	Φ	¢	§	Ω	Ψ	%	¢	♥	¶	€	†	♦	Ψ	♥	⊕	Ω	Ψ
C	β	Ω	†	η	η	⊕	α	Π	Λ	⊕	¢	€	#	⊕	⊕	¥	⊕	⊕	@	η	⊕	€	⊕	⊕	⊕	⊕	⊕
D	α	†	Δ	@	⊕	Ψ	Φ	μ	Ξ	♥	Λ	α	¥	⊕	⊕	⊕	€	⊕	⊕	α	β	⊕	♥	♦	⊕	⊕	⊕
E	€	#	†	¥	Ξ	Π	Φ	β	α	€	¥	ε	Ψ	¢	Σ	♦	%	η	%	β	ρ	¶	Φ	Φ	μ	α	
F	†	β	†	ε	ρ	Ψ	Ω	€	ρ	§	μ	Σ	¢	Λ	Ω	ρ	Π	¥	†	¥	Σ	¥	†	Λ	Φ	⊕	

k = 7	H	J	K	L	M	N	P	Q	R	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9	
S	¢	Σ	€	Γ	β	¥	§	¥	β	†	†	†	€	♥	Σ	Γ	ε	§	α	♥	ε	†	¢	α		
A	♦	†	Π	⊕	¥	§	¥	α	¥	♦	%	¢	¥	Ψ	ρ	ε	η	†	β	α	Σ	Δ	⊕	α	†	
C	#	♦	%	†	♦	α	α	Ψ	Ψ	¶	†	†	€	η	♦	ρ	⊕	⊕	Λ	§	Σ	μ	¢	¥	ρ	
D	¥	†	€	Φ	α	Γ	@	α	η	♥	♥	♦	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	†	Σ	¢	ε	%
E	Λ	η	⊕	η	α	Λ	Ω	#	†	♦	η	ε	♥	α	Σ	β	⊕	♥	⊕	α	§	β	α	η	Ω	
F	⊕	Π	¥	Δ	β	¢	Ξ	#	†	♦	η	ε	♥	α	Σ	β	⊕	♥	⊕	α	§	β	α	η	Ω	
G	⊕	†	¥	Ξ	Γ	Ω	Π	Ψ	β	η	Δ	Γ	†	Δ	@	⊕	⊕	Ω	¶	†	α	#	⊕	Ξ	Ξ	

k = 8	J	K	L	M	N	P	Q	R	T	U	V	W	X	Y	Z	0	2	3	4	5	6	7	8	9			
S	μ	%	β	†	ρ	μ	%	Ψ	Π	♦	Λ	ε	§	†	β	Δ	Γ	α	¥	♦	¥	Γ	Ξ	α			
A	⊕	⊕	Σ	⊕	Δ	μ	§	α	ρ	¥	⊕	Π	¥	†	%	#	†	¥	♥	†	@	#	ρ	¥			
C	μ	†	¥	α	Σ	Γ	β	Σ	η	Σ	¥	Δ	Λ	Φ	¶	§	α	¶	β	Ξ	Ω	ρ	§	Γ			
D	Δ	¢	@	Ω	¥	Ω	β	†	μ	ρ	€	Σ	†	μ	Σ	€	€	⊕	⊕	⊕	⊕	⊕	⊕	⊕			
E	¥	Ξ	Γ	Ω	ε	μ	†	@	§	Λ	η	Ξ	%	€	ε	Σ	α	€	⊕	⊕	⊕	⊕	⊕	⊕			
F	α	ε	†	⊕	⊕	#	†	@	§	Λ	η	Ξ	%	€	ε	Σ	α	€	⊕	⊕	⊕	⊕	⊕	⊕			
G	¥	ρ	¶	μ	ρ	Σ	Λ	§	Ξ	⊕	α	α	Γ	¥	¥	¥	¥	¥	¥	¥	β	Ψ	♥	Ξ	Δ		
H	♥	α	¶	Λ	@	β	¥	Γ	μ	♥	Δ	¥	¢	ε	Ω	⊕	†	♥	#	⊕	¶	ρ	%	Ξ			