

# Shamir Secret Sharing Codex

Revision 2303-1-8822ef51

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/>

NEVER ENTER SECRET DATA INTO THOSE WEBSITES, OR ANY WEBSITE.

Revision 2303-1-8822ef51

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	3	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	
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	5	U	0	V	L	N	R	P	F	T	D	Q	9	7	6	M	4	3	G	2	1	
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

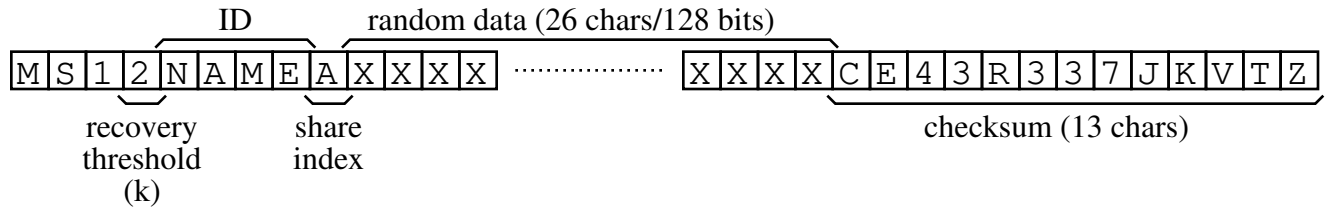
## Translation

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	0	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	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	#	Θ	Λ	τ	Ω	¶	¥	◆	♣	Σ	ε	β	μ	Ξ	Φ	ρ	¢	Δ	†	♥	Θ	α	π</							

## 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 .....	26
Module 2: Extra Share Generation Tables .....	35

# Part I: High-Level Introduction

**C**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

**T**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. In fact, even when using a multisignature scheme, there are still keys to be backed up, which could use SSSS.

[illustration of SSSS vs multisig]

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).

[illustration of SSSS vs multisig]

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 word errors and correct up to one word 100% of the time. Additionally, it will detect other random errors with extremely high 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 character 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

**B**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.

We note that Bitcoin's EC signatures have only 128 bits of security, so as long as your secret is randomly generated and securely stored, additional bit-length serves no purpose except to complicate things.

## I.6. Bech32 and Alternative Alphabets


**C**odex32 inherits its name from the Bech32 alphabet. In order to store 128-bit secrets, we re-use the Bech32 alphabet, which consists of 9 of the 10 Arabic numerals and 23 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.

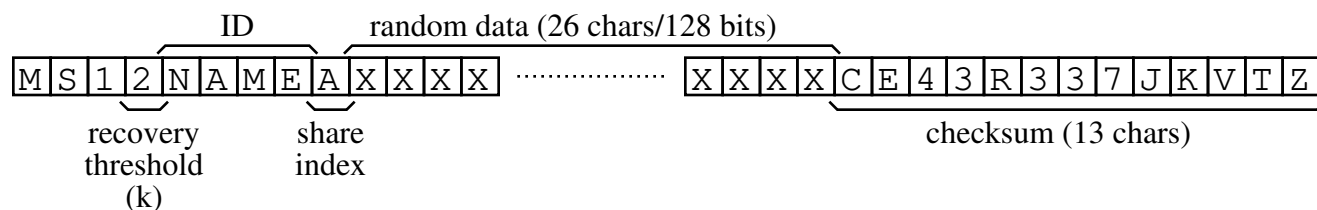
Parts of the codex32 process use an alternate alphabet, consisting mostly of Greek letters. This alphabet is used for intermediate computations, but never for data storage, and nothing represented in it is ever secret data. A table of pronunciation is provided on the Reference page at the beginning of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



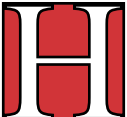
The header consists of:

- 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. 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!

**3. 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.

# 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](mailto: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.

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

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

$k = 2$ . 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.

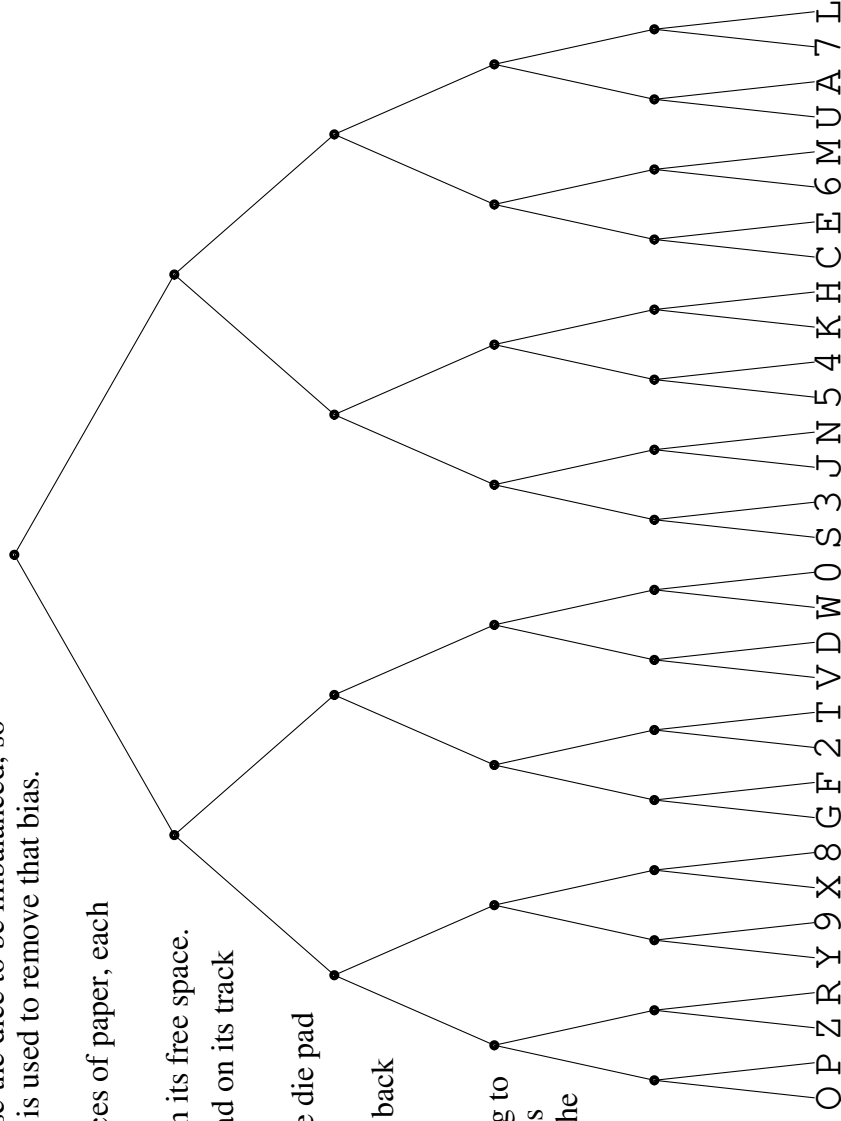
Once you have copied everything into the translation worksheet, follow the Translation Worksheet Instructions page as though you were deriving a new share. The result should be a share with index  $S$ , which is your recovered secret.

# Dice De-biasing Worksheet

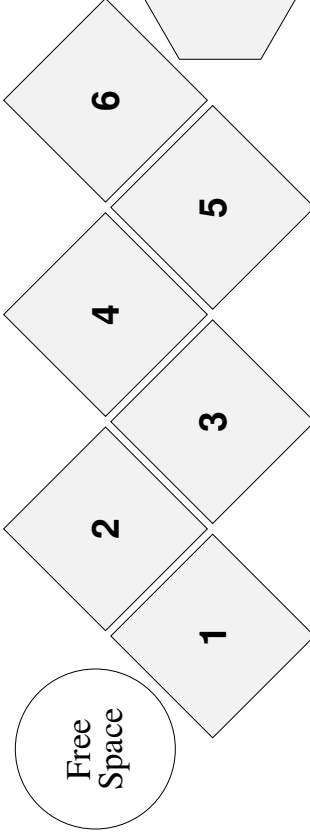
Most dice have small manufacturing imperfections that cause the dice to be imbalanced, so some values appear more often than others. This worksheet is used to remove that bias.

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.

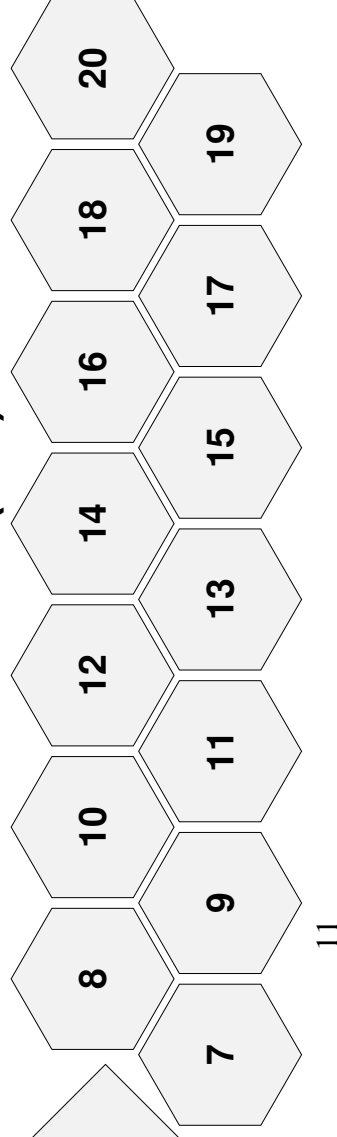
1. Choose a die track for each die. Put the die's marker on its free space.
2. Roll all five dice. Move each die's marker to the die pad on its track indicating its value.
3. Re-roll the same five dice again and set each **die** on the die pad indicating their second values.
4. If a die showed the same 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.

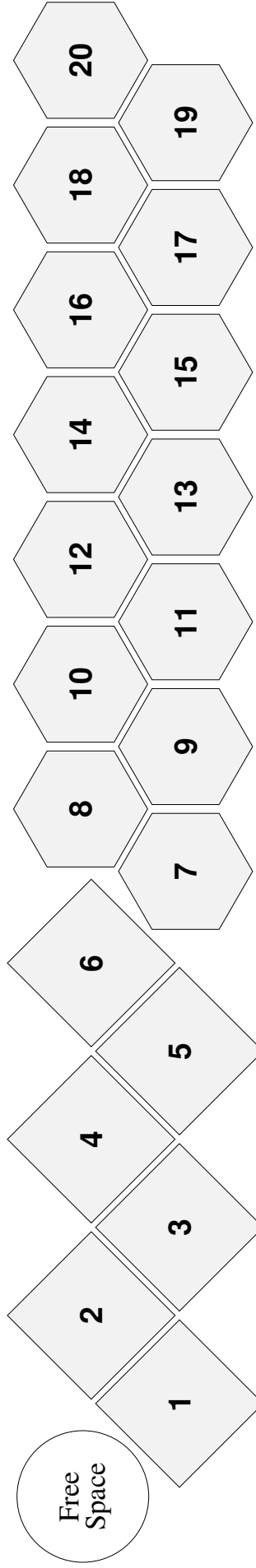
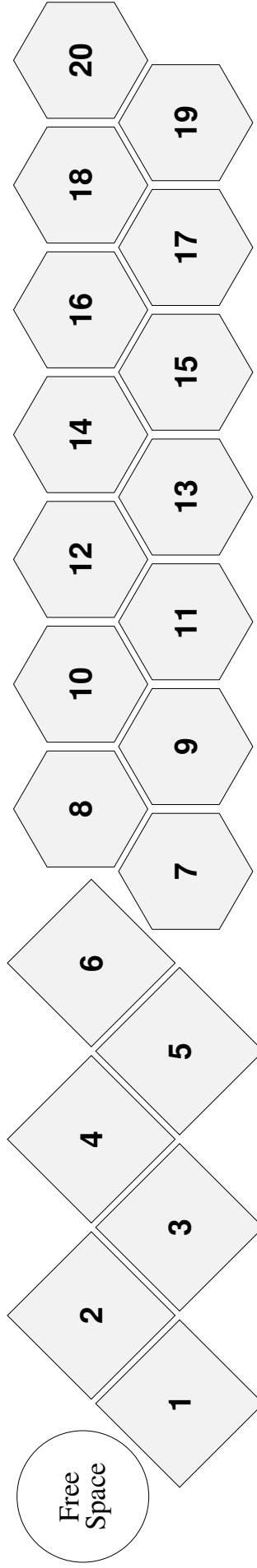
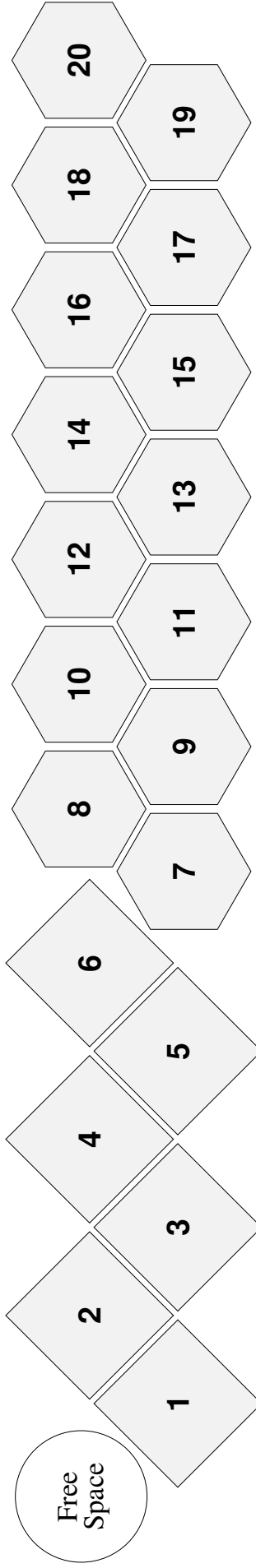
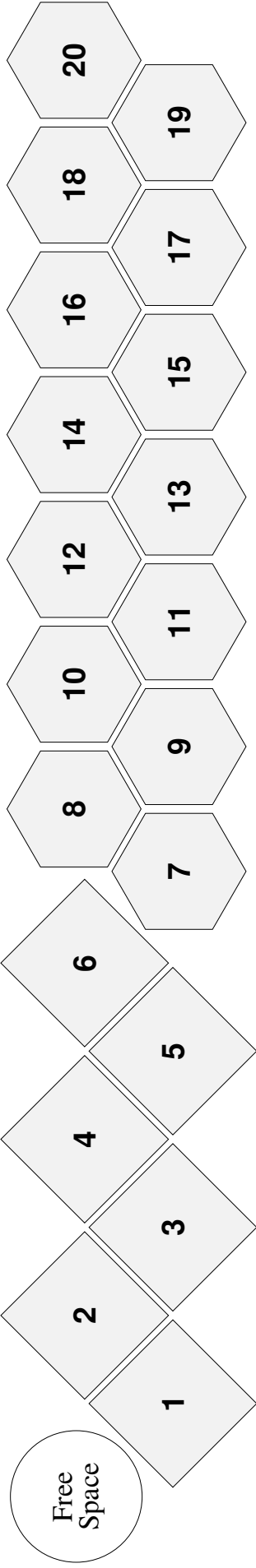


## Die Tracks



## 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 bottom row with the word SECRETSHARE32
2. 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.
3. Add the first row to the second to fill in the third row, using the addition wheel.
4. Look up the two leftmost under hanging symbols from the third row in the checksum table (pages 15-16) to fill in the fourth row.
5. 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.
6. 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).

Note. To avoid confusion, we recommend slashing your 0s, 7s, Zs and Ss when handwriting.

**1**

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		
=															

**2-4**

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		
=															

**5**

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		
=															

# 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.

Note. To avoid confusion, we recommend slashing your 0s, 7s, Zs and Ss when handwriting.

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	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			

codex32 Checksum Table

Table with 26 columns (AA-99) and 100 rows, containing alphanumeric checksum data.

Table with 26 columns (AA-99) and 100 rows, containing alphanumeric checksum data.







# 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 9, 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

# Translation Worksheet (k = 2)

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

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

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

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

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

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

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

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

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





# Module 0: Vowelles

K ↔

L ↔

M ↔

N ↔

O ↔

P ↔

Q ↔

R ↔

S ↔

T ↔

U ↔

V ↔

W ↔

X ↔

Y ↔

Z ↔

A ↔

B ↔

C ↔

D ↔

E ↔

F ↔

G ↔

H ↔

I ↔

J ↔

0 ↔

1 ↔

2 ↔

3 ↔

4 ↔

5 ↔

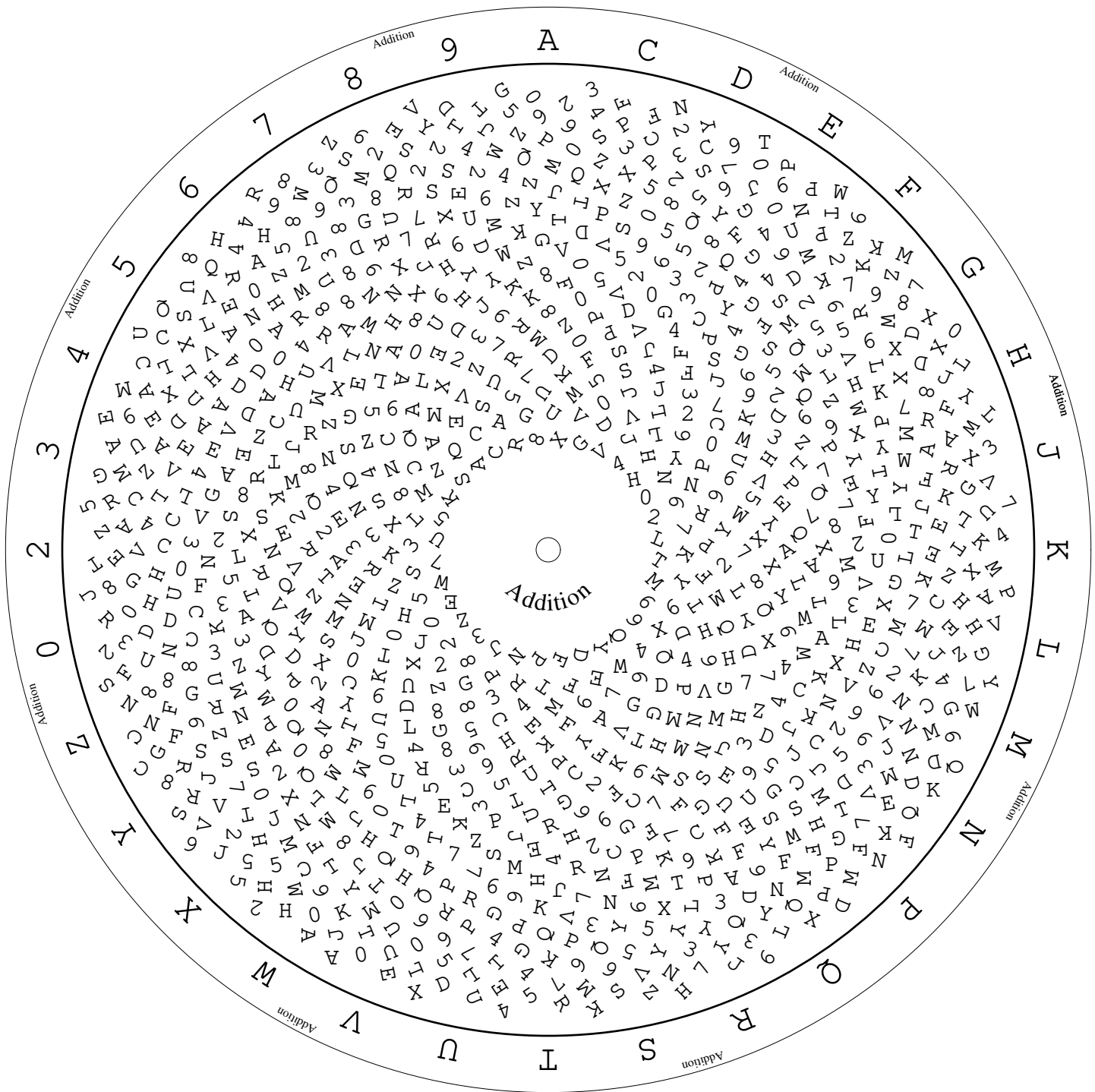
6 ↔

7 ↔

8 ↔

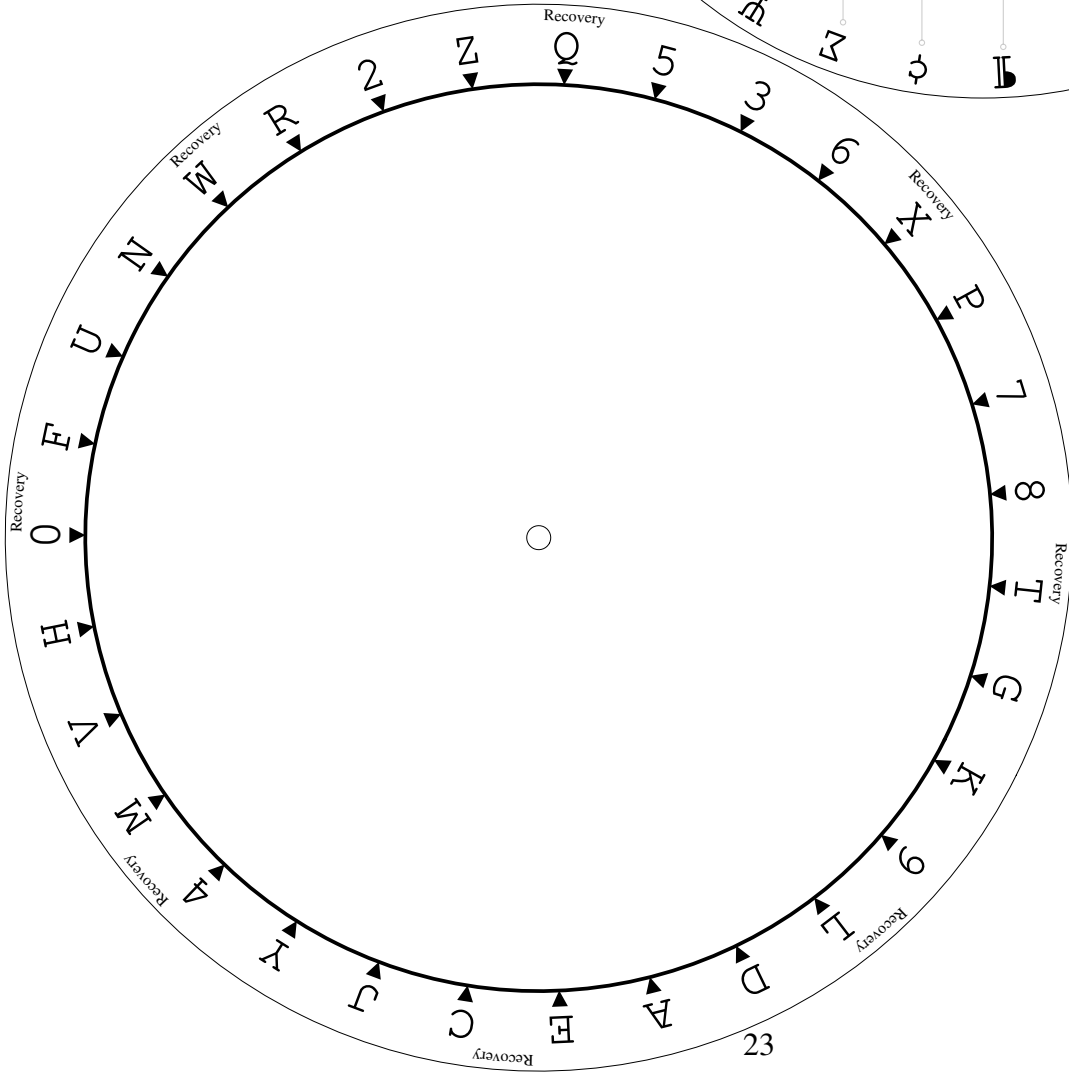
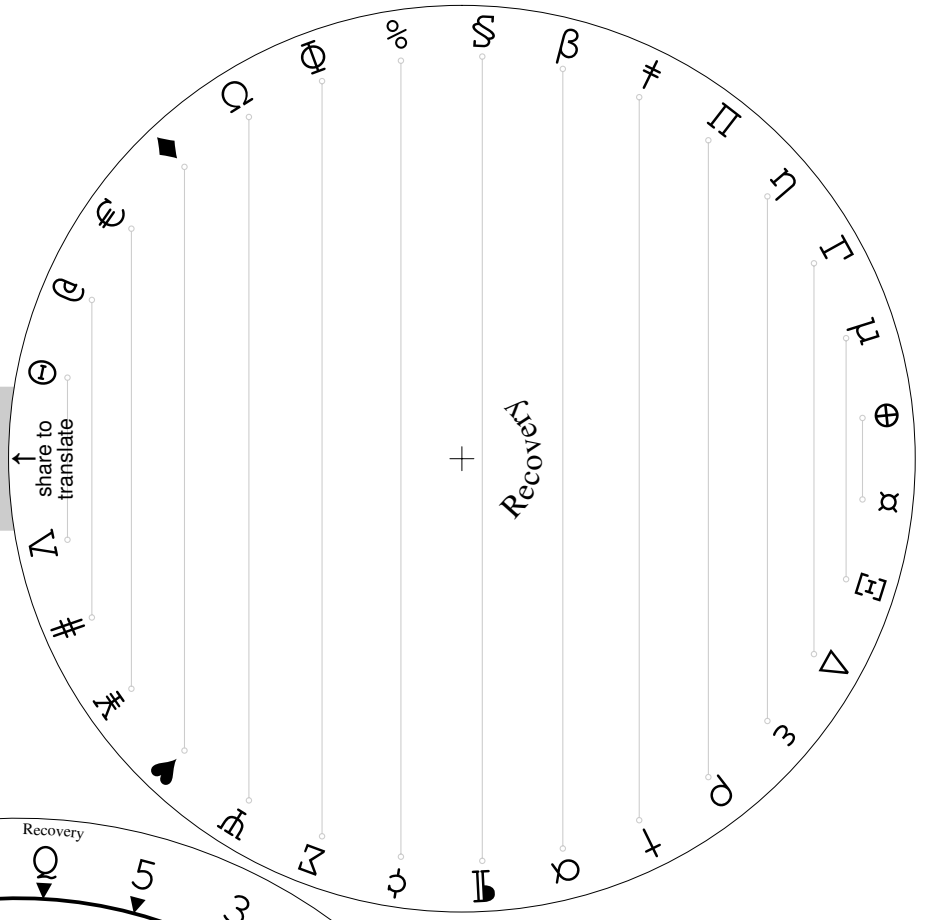
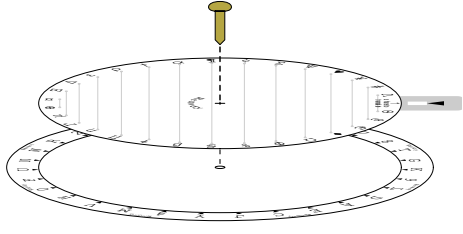
9 ↔









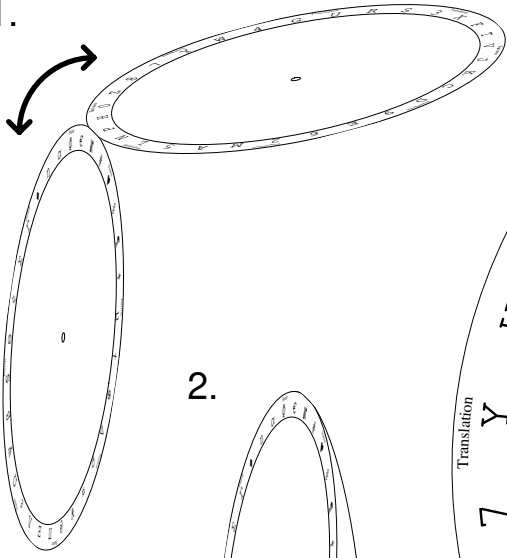


↑ share to translate

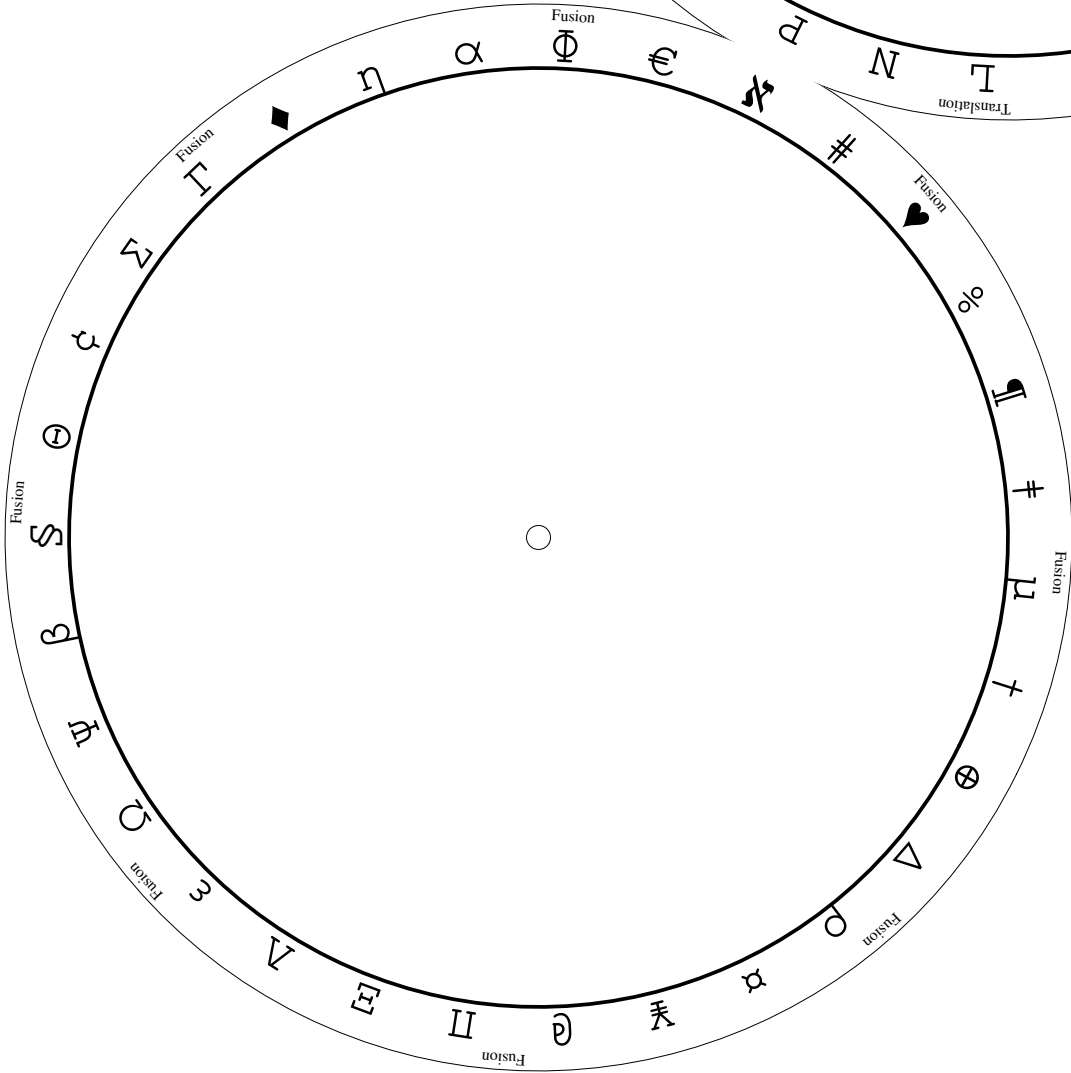
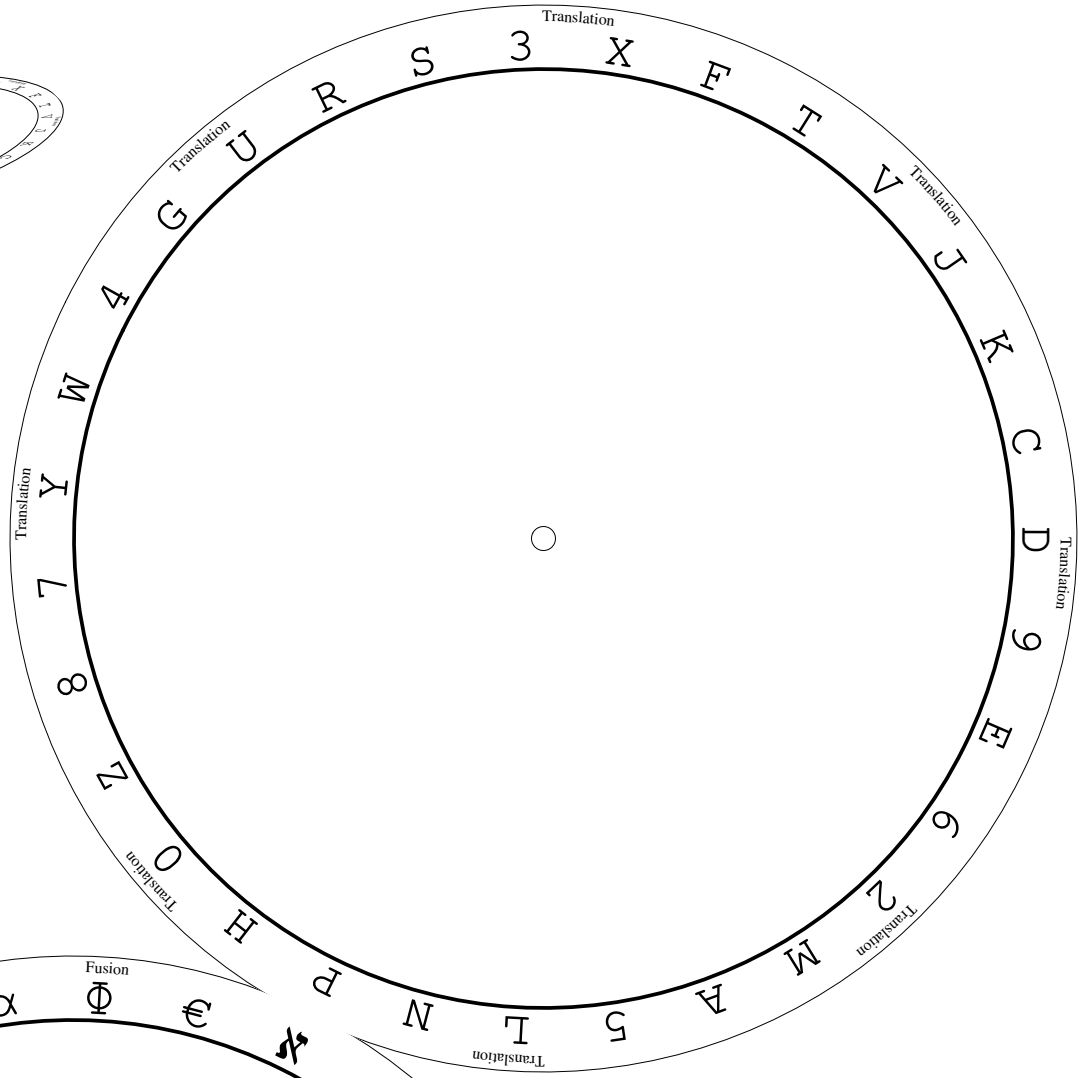
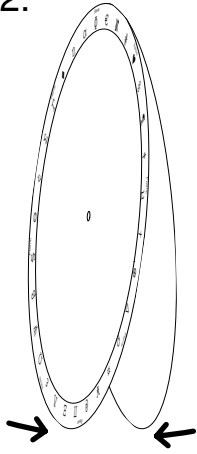
+ Recovery



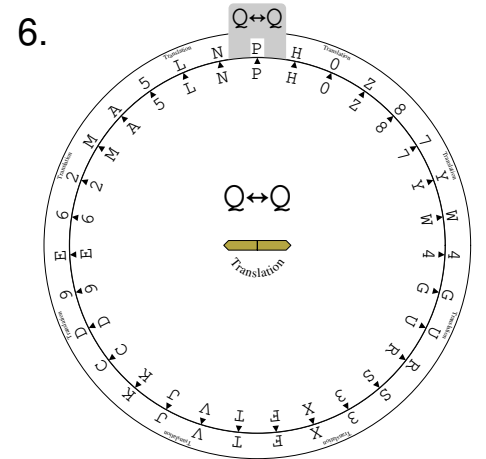
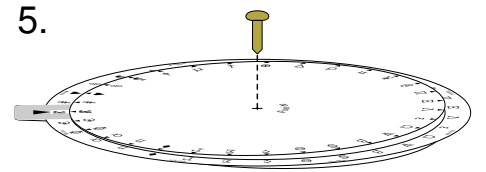
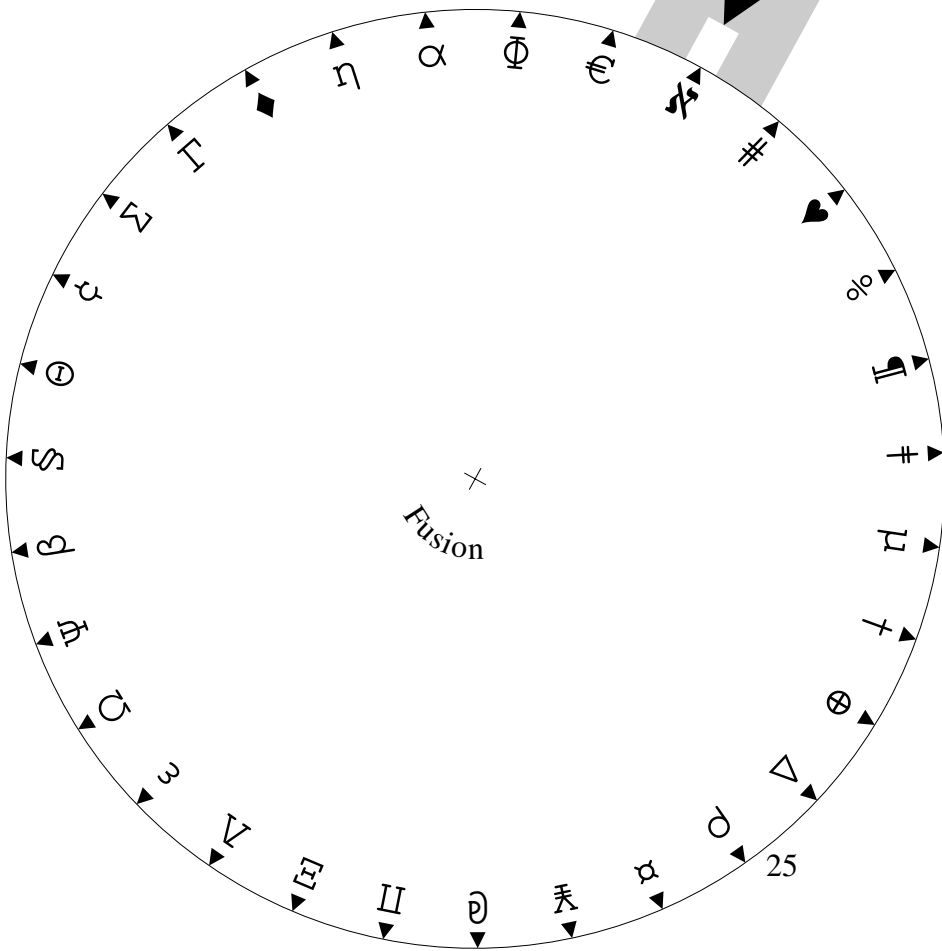
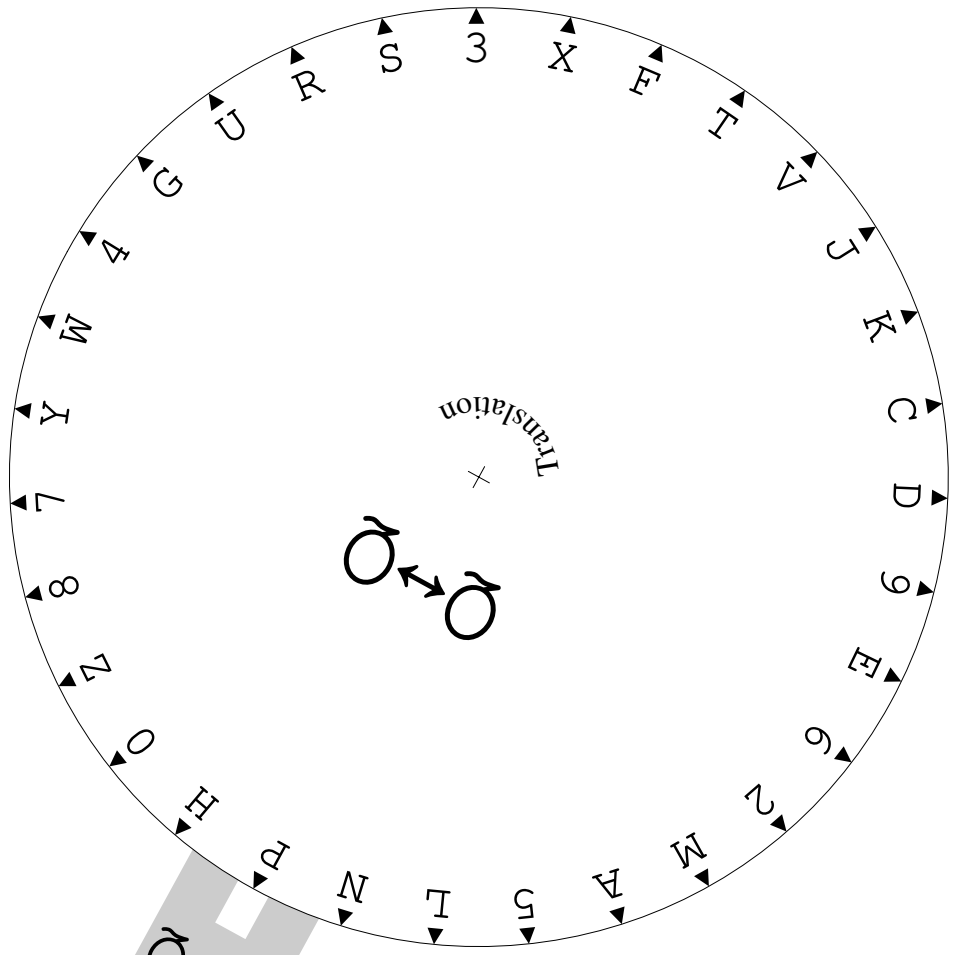
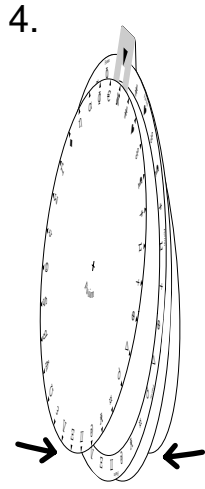
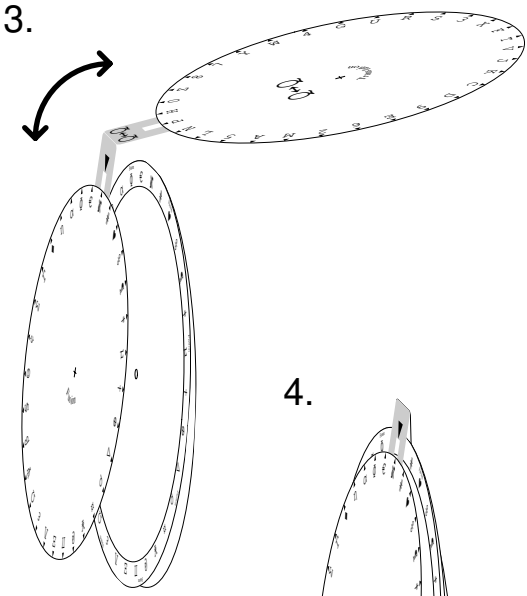
1.



2.









## 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
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
CCDEFGHJKLMNPQRTUVWXYZ023456789
DDJQOZ5YE86FU3X4SL9KVTGCNWRH2MP
EEQSDY43U0ZCH79L5FMPNVJK28WTXG6
FF0DLMUGC49VKJ7HETR2Z68NY5SP3WX
GGZYMH7TJ9U8QVCR3WKD4560LX2SFPE
HH54U7FRTE3PVWGCCLKJZX2S69DONMQY
JJY3GTRVQMH0SNKW78PEL4ZDF9X5C6U
KKEUCJTQPF26SWVHN893YDX7L4ZR0M
LL8049EMF5XTCG3UDH7N6PWWZSQKYR2
MM6Z9U3HGXEWJTF7YRC05SP842NQLKD
NNFCV8P02TWYXDS6KZ57GML3JHU9Q4R
PPUHKQVS6CJXZ58NT20M73E9RFLYWDG
QQ37JVWNSGTD52P8R06UFLYEEM94KZH
RRX97CGKW3FS8PTJM0V4ED25UYZ0HNL
TS7RQN825JVE4X60WDZHC3UKGMLPYT
UT4LHRCVWU76N8JKFPQY9X5ZMED2GS3
VUS5E3L7HDYKTRMF4CG62NQPX08V9JZ
WVLF7TWK8NHRZ20QPC6S3M94YGUEXJ57
XW9MRKJP87C506VQGSNLUEX4H3YDT2F
YXKP2DZE9N07MU4Y63LWQJCRSVTG5F8
ZYVNZ4XL365G7FE92MUQ8WTJ0PKRDHS
OZTV6524YPSM3LDXN9EJWRHG8KC70UQ
20GJ86SZDWPLEY25Q4XC THMFVR7UN9K
32CKN06DXV839E5ZPY4RJGF7QTHMSLW
43N2YL9F7Z4JRCUMXGHS08VQD6PWCET5
54W85XD9LS2HFMFE0U3VPKRT6QJCVZ7N
65RWS20X4QNUL9ZD8EYTKC7HPJG63V
76HTXPSN5ZKQ9Y402VXDGR7UMWCF38EJ
872X3FMCRYLQWKHG9JTD5DONSEZ68UV4
98MGWPQ60RK4DZNSJ52FHU9LT73EVXC

CDEFGHJKLMNPQRTUVWXYZ023456789
AAGUVDJHN67KY9XWFEFTRP8205S3LMZQ
CCCCCCCCCCCCCCCCCCCCCCCCCCCC
DD8P7JMWHQEZ9FAX5TNGS3L26R04VYU
EE43U56MJGPNWX7VSADL0K9YR2ZFQH8
FF6VWTNZKJX8029YAL457QSPDEURH3M
GGH9283PUZLVE5RS46QNFWDATXJ07Y
HHP5DWXF97J24KNTMZYU6S8GVQR3ALE
JJ3SEM VXWU5YFTDA6NH8R04LQG2K79P
KKVAFZY20WTPRG4LN83MD765HJEQ9SX
LLJWY4K38N9QPS20FRGD XME7TAV5ZUH
MM0R5V7AXP69TNJDQHW3G2K4U8LZEF5
NNUEAHW9YVD0L4TFJKZQ5PGRM6S8X27
PPFK8SR65L3DMVUQ007E9ZTWH2YNXGJ4
QQYLRUPE70GXDJ658MVZ49NT3KFHSA2
RRNY0G8UQK2M7ESP56T9HAX4FWD3VZ
TSTZ3RGQ640JV7PU2E5FYXWL9HA8MK
UTQ7XNHYZMA32LF9D4K6EUSJ5PWG0V
VU94GSP5E28AJMQ63V7YKFN0ZTWRDL
WV2G67EDASQFNH MJUWX08LZKP34Y5TR
XWS6JXATFEMLKZHN VY9PQR387UGD45
YXRQMADNT5V4ZYWH79FSUG03EP82JK6
ZYEJN9F4LAHR83ZKW027M5UQXV6PTGD
0Z7D7Y9L2XNSG8K4H30VJQE6WM5UFR
20ANK2LGRFZ5QU3SRP5XHDVVM9WJ746T
32DHZL48GT Y6UP039SRAWJ7VFXMEKQN
43XT402RS9KE6Q8GZUPWNAVJYHDVL5F
54MX9KZ03HFUSRL2TGH8JAV5END76YPW
65KOP6QVM8SHXAE7RDJ42ZF9GLYTUW3
76Z2SQU7V3RWAD5EGJMKLYTF849NPX0
87L8QE5JDRUTHWVMPXA234YZS0K96NG
98WFL30SPY4756GRKQUHTXJDZNAM2E9

CDEFGHJKLMNPQRTUVWXYZ023456789
AAZSLJY5FH2EVPK9Q04XUMCGR7N86ET3
CCJ5T8WMZ4VHSUGNP7XA9Y2L0K6QE3F
DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
EEEA0TVC94UXZW8FMN6J2HR5YQ3P7K
FFSQA59PVLJG8W40YET3RUZX6H7MK2C
GGAFEC8ZX0TR3SNMVU7K5JL6P9Y2WH4
HHTC73S2LNX9AJRQZY6E8V40MWP FUKG
JJMU2Y6953QTP0AKR4FZ7N8CHXGWLVS
KKX36AJF4RLWTV952P07SZGNQUMCYEH
LLCZH2QJA730F56YS9KGM8TEUNWVR4X
MM908NG7UVW2RHZXE3S54KYJTF6CQP
NNK4WGCX7PEQH3MZTSU9FA6YV5JL8R0
PPR6504EWJ9ZNGVTKC8QLHUSA237FMY
QQWNSRH6YZUF9K2L7AJ8GEPVXCT035M
RREGUH3L6M75KAQ2XJYWC0TPZ8V4S9N
TSPWZU7RQCMAY63HNL2VE05FGT49XJ8
UT2J4VP8CKF7ZMEW5NGLYQ3H96RS0XA
VU0EM7XHR8NJ6LS3G2QPT495CVFKZYW
WVQYFP0W8A5XMNTE9GC26RS3KLHU4ZJ
XW6KPEETGN50S7X8C4ZMYALRQFJ2HVU9
YXFVZMS3EC62Q7U8RH4P5AKW09JNLT
ZYN7Q6LK9SRV04JAHF5MXGW83ZCE2PU
0Z5PCMNUST8LQRX7WH3F09JAE4KYG2V
20HL94FTEYKMGCPVA8WR237UJQSX5N6
328M3QRYJXS459L6UKACNWVT7GEPHFZ
43V8XSUQ2GZKJYHRM6LTPFPF4NE057AC
5432KF5VT6ANC80PJWEHQXS7YRUZ9GL
65URJ9KOP2YCWEF46TVSH7MZL3XNA8Q7
76GXRL2AKUHP4FYJ35V9NZCEWSM80Q0
874TNXZ3HWGYL2USCQR0VFK98P5AM6E
98Y9VWENMFP3U7CG0XZJ6Q24ALRTS5

CDEFGHJKLMNPQRTUVWXYZ023456789
AAAY5FQ34CTXUN6PMSDL9HGKJWR20ZV7
CC5SA4M6NFPQJ08LD3UV7WHY9ZTRGPKX
DDTFMLCVU3PGJK6NA08WX7RHQS5942Z
EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
FFKYTXSQAR7DC4Z35MPJG0VUH28NWL9
GGMD02JRH8XWY9U754LP3QVCAZKS6
HHDUGR9TW02QZF57JXS6VLM4KNCYP38
JJACUK0Y9DVW752GNHRP4QFZ6M3X8TL
KKQ4V08GYLNT5HJ26R9AM3XFDZSU7C
LL7XPC4NVZA2K0D6Q8UTS59R3WHYMJF
MMRT3PALDSZ0UV4CFN6H792GX5YJQ8W
NNS3C6D80A49G2VUMJKXZ57PFTHL YQ
PP97ZAQCLWF8VNM4X6DR5YJ2SHGK3UT
QQGHX3ZM47SV6DAPWLCYR20KT9J8FN5
RRLV29Y7T8J3FXH5KSWDNC P M064AGZU
TS825WTZ3YHCOMPXFR AQ0JU6N9KVD74G
UTVKR75XF29MAQWSY3ZU0NLDG86CHPJ
VUFADV NKJMLH9Y80CG2ZQXTW43S76RP
WVXQLN60KPCRYGU842JF3S7TMZW5D9A
XWUJHT7FZGR4PASX9Q38KVD6Y0NL5M2
YX0G7SW3Q95L4MFXZHPAK28NVRJU6TCY
ZY46KG2H5V0FSW9R8T7CDMQAULP3JXN
0ZJ9WFXAPHT6LC3Q74M2YKU85G0VSDR
203MN8U2GC67HRKJD9YQFPZSXLAFWV54
32PL8JK9R6UST7GYV5HMCAGZ3N4QF0WD
432RSZFFPM5WNDLQATC4G9J807YKUX6H
54HWQMPD6X3K8UCLZVN5TRGYF792A0S
6568YHRWSKGA3Z7T2FXNUD4CJXVLM9Q0
76WZ4DLU8QMY2JNVPK0SFTH5AX7RCG3
87N095HSXJYPQ3TWGZFFV86CL2UD4RAK
98ZP6UVJ24D5R90KLYG3AFWSQXTNHM



CDEFGHJKLMNPQRTUVWXYZ023456789
AAMET0GUV...
CC TAL6ZN8RHK7M54VWY903PSUGDJXQ2
DD5YESP9GMKVR23TQA048CXZ7N6LJUW
EEEX5MKQSDNWAG9JPTY3ZCLUV86HR704
FFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
GGYQD9T2RESZLH0APVUW7854JMNC3XK
HHJ32VUZ69AYN4LX05CPDE7QGSKMR8T
JJZ47EKD38M20GFSW9T65XVHYCLUQAN
KKLCWQ7PS4Y09TEJH83DX6HRUNZV2MG5
LLPTRHV6CG2W8NXZA45S3JQK0DE7UY9
MMUXNWKYKE64TDS7Q5PJVLR0ACH2G83Z
NN0U645WMHZPEK8YXQ7ARG3TTL29DCJV
PP6NQJE3TY7RA0SDMG9C4ZHLW5XVK28
QQH6Y7MJ58GT3KENDSLZV2R4XUAW9C
RRQP2AHLD94C6UVTZXXK7YVW3EM805S
TSC8KPJT9WQU2AD370G5N6LXM4ZHERY
UTNMPP3D0AQJLVU9GER28W46CKY5ZSH7
VUKS0R2LX3GN5CVH96ZEPQWMTJ7YA4D
WVEDAURXZT0845HLGC6JSKM79PQW2N3
XWRL4Y8QKZ53SPM7CJEUH2G06VA9NDX
YXS9ULHC50RMY8Z62N4DTPKEA3JQVWG
ZY2H58N7QXC DPJWM6EK RVA9GZU0T4SL
0ZDGVXL54AU7WY6CR8N39SEJ2TPKHM0
20WK3G9RUJD6XLA2SHVMQY4NP78M2ZE
327J9A0VHST56ZR0U3XLQEM8YDKWNGCP
434WJDSG07EHURT9K2AN5Z6Q8CXPMV
54GRZ5CYWVXJKQN8L7M209D3HATSPEU
6592XC68YULEQ74NHMWGATSDV03PZKR
7630HZX4N2VQMDC5UY8TGDJPR9SEL7A
87VZ8MWEJCN93DQK4SPHXUA25LROYT6
98AVCN4M7L6SJEYWKZKQ2U0T9XRG35PH

CDEFGHJKLMNPQRTUVWXYZ023456789
AAH3WZT78SFJ4EDKYMRCQP062NL9U5V
CC846HP3VJ0UFYVQ92SRE7ADXMKZLW5
DDCVUAZ8R2WX54K7FEMN3H6LYQP0TJS
EE2U7MRJXZKHL645D09YWSQ3AFVN8PT
FF9PVY2TZC3R7KWUQDA0LX45N6JES8H
GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
HH7E0PLQ359WY2CNXSV8MKZAJRDT6F4
JJWAM5406KRQCHX98PLUZF5S27TYVEND
KKNSTDARMYU9J578W4EQVCLPF3H6ZX2
LLDRX60CNEJYSVPH53QK8AUT47ZW92M
MMSWKR85JTD60E4A9X2FVNQZY3C7LU
NNR5LCHVSX6TWFQ30Y2M48DK9E7APUJ
PPKM9L6NQ4XF2SHCJV37RDTZ58AU0YE
QQMJPNCS29LZUW3V6FYE5RK7048DHTX
RRVFD8745UAL09MEZXXJSY3CNT2QHK6W
TS50NV3FWLCKAZ2YHTUJ94RMPXE8QD6
UTLNYUWDK324MRZAS87PC6X9VH0JFEQ
VU6C2WFADQSER8TZV7KLH0JX3P95YMN
WV49C3QYF6HDZTS2PUW5XE8RLJMN7NA0
XW0HSFYZANVM87UT3KD6P95JQLX42RC
YXUDEJ56L7M3NC90RHPTAW2Y8ZFS4QK
ZYXL32SUTHQ8KDFWNAZ96JE4C05MV7P
0ZPQFTUK7VY5EMAD2R8HNL90SC6XW43
20Z759XPHR4S3Q6LENCATFVMMDUYJV8
32J6QSVWUPN7DAYFCZTX05MEH94R3KL
43EXHQ2Y0PATUVSLWF4JM7865RKCZ9
54YT8EMX9A7CPL5JK60FU23VDWSQRHZ
65FZR4E90D8NHPJX7L6WTYVSKU23MCA
76A8J09HCM52V3LP4QND7ZWUEKTFXSR
87QZ2KDMFEFT0XJ8RU543SNPHWVCLA9Y
983YA7KE4WZ69XRMTJ5V2QHCUSNPD0F

CDEFGHJKLMNPQRTUVWXYZ023456789
AA7LK5RFPNMTXYZJ4UCGVQ6S089E32W
CCWMZPFXX23A96EUY5DNTN4LG8R0V7JSK
DD3ZYT2S9FUL8GE6C7QRAK04P5MNXXVJ
EEERJXDT9QPYYWV86SUG75Z34MCAK02L2F
FFV05JZUYMR6DTPC32XLN8Q7KKG9AE4
GGPXS7QLWT63M48VE0NCYFAZDUJ59K2
HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
JJ8NRZACU43ET2FPKXYVWVG9QML7S5D0
KK67NM45A8WUPF3RLJZSQE2TV9DX0CG
LLYD7V8046QAR3WN9KMXTUFPFS2CJG5E
MMEQW40R5GLCFJK3VZA69DX28STYNP7
NN268WLMK9GJA5047R3TEXCUQDYPVZS
PPL4AFJYXK5SEDCURT2W0V7G3N8QZ6M
QQJUE9S8VXDM0N7GTWLFRCZR52PA364Y
RR9843KZJL0XUC5ANPFQGSDEW76TMYV
TSAPT6E7GU20WL9QXV8ZF5K3YJRMDC
UTKAU2X6SJC VG7DEPQ935MN0FR4WY8Z
VUNKJCP2TRZQS6YXAED0MW8V54LGF93
WVUTQ8GN0E953KLNWSM4Y2CJF6XPZ7RD
XWVEGLV4MS7Z5RN0Q3K2DYPC9TUF8A6
YX4RPYUDEAFGQ92TJS6M30LWZKNVC75
ZY03FUCTD5J79SX2Z6E4KNVLAMW8PQR
0ZGW3A5PC0KD2XJFMYU8L7S94VQ6RTN
20TSVNWK3Q8FZA4MG5RD62UY7EXCLJ9
32M5CXYE6ZP87QTD9SFR4WNJ30LUGA
43SG0KMAZVNYCPR5WFJ976TDLQE24U8
54D9L0N3R7VPJZMK8A5ESTYXG62UWFQ
65QVMR3JFW42YUAZ0CP789E6NGSDKXL
765FD2EDQ7CXNLS9Y8GAJRMKUZ34TWP
87FY6Q9VL2EK40G8DNWPUJ5ATCZRSMX
98C29G7WVND SRKMLV640UXPZJEYFAQ3T

CDEFGHJKLMNPQRTUVWXYZ023456789
AAT78YVR05CQLUEP3GWKS6DZX2FMH4N
CCM6N3XVYK7WRP8HSEADF9L42ZQU50T
DDYR4M96TWL57SZFU2KAPVC8GEH3QN0
EEKNHAY0WX894LPRCUG27TFZ3S6DVQ5
FFEW9ZDK2UQ35N6T47SP0AHVLSRY8MXG
GG58PW04QVE6ZDULAM9XC2NSY37KRFH
HHZKX8CAES5UW0VYNRPFDTQ976M43G2
JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
KK0LZT67NQDHC32SMX5WURAE9GPFYF84
LL3V0UG9MARK6F4QPZDCHX7NE85SWTY
MMVPL9QF60UNS2DZGKTYEH3CWA8X47R
NNLMK7S3CZTEYV5X6H849U0WFQGR2AD
PP25VEAWG3HMQ4R08LUSNKF6C7TZY9X
QQ8AG4LDZPWSKT9M06FHYC5XR3V3NU2E
RRSXYPEGUCVD9Q0WH4L7526T8NKFAM3
TSGQ62K5XMFYH87NZC3U4WPRDL0ETV9
UTRUD6FS74M83XK295N0GPYAQWVZCL
VUXHRGWQ9YPTFZL4EDM385S7ACN206V
WVF23H8EP7XLGWYA50R6KZ9MNTDQCUS
XWNC0RL4HAFDMGUY9Q537K2VXSTPZ8
YXQZS5N8H62REA3CKYV9D4GUTMLW7PF
ZY6SCVHPRN34UGAEXW0T2FMD5KZ98L7
0ZAA0QDMTKG4XN7F6LS2ERY8HUPVC95W
2073ARPUL8YZM9WGVQ4NXSTKH526EDC
32W4FKTN59ZV8CS7D3XGL0EPMURA6HQ
439F7X5HVT50PEC82AYMQLUK4GNR6
54CYWLUMDE02T6Q9RFZ8V3N5PHX7GKA
6564D2N7C8FKPAYX3TVHQMLVY69U0SEZ
76PDMFZ2SL9CX5TKQN7RWEVW4A0HD3U
87U9TS2X3D6AVHN5F8CLQGR0Z4WPKYM
98DT5C3YA2NG0RHV7PEZ6M4QSF9LXWK

CDEFGHJKLMNPQRTUVWXYZ023456789
AAJHNEQT2ZRP3M98UCVW60XD457FGYS
CCZN8V2YPPQ90DXM3AFL4RS56EUHWT7G
DDA2PXFJWCY4SH703659TENG8QRLZV
EE06RQG8TSUYMCA94V2J57FXZWDLN3H
FFQ83LP702MS65XDCWJE9GURVAN4YHT
GGDW4NRA96ZMV2QEST87JXPLH0FYUC5
HHMLJ654UX0AQGSZ7NR3PCT2DYV8WEF
JJTMX076HYFNU4W5LZS2C8EAPV9QDR3
KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
LLG9MPYD7TCH5WFVXVJ0QAN4U2ERZ368
MM4GTCVPLE8J7D3Y9XFUNZ6HARS520Q
NNXJZRUEA5SC2TGQH89D0FYPP67L34VW
PPNACY3XD8V6WJL207GERZ4TQUSM59
QQ75UGN98H43CVEAZ2TOWDLFSJXPRM6
RRF0SU4QEWHVT8NG69AX7L3Y5DPMZ2J
TS3FWH6URDJ9EQZ40GNYLM2V7PCT5AX
UT64E89CMRQXLP2VGY3HZ50JNSW7AFU
VUL7H4ZGQJ6289RN5AEFDPM3WXCYST0
WVSR92T3YGA7XFCMELPZUHW5Q46J8DN
XW23DJ0HSPXGRU56F4ZVMTA9LC8E7NY
XYETYFL0JV3ZH6D7M5WA8QRNC9GUPS2
ZYREV3MFX925J0PLT7DNQUSZ8G4HCWA
OZYX5SHRN7W8AE4UJQSGPF3VC0LM269D
208CF7D563LR4ZJWPHVT9QEY2AGXUM
32HUAT8M3NEDFLVCQPYS46JWGZ509XR
43UQ2MCLFATW07YPR8DXRG4HS9NZ6VJE
54PD6ZSNG05T9AURWEQLXYCMJF3VH87
65VY7WJSZLDQNR6HXU4C3298FMTA0GP
76CP05WZ4F7EGNHSDRUMYV8TX329JQL
879VLDXW5MPUZS0JYH682AGQ3TENF4C
985ZQ9AVCUGFPY2N3M6SW70RHJDEL4

CDEFGHJKLMNPQRTUVWXYZ023456789
AA9DH8METKYF62N704U3CRVWQJZ5XPG
CC4FRJ3KQ5Z8XHDGWY02PU7EV6TN9MS
DD2S9CJQ0VRAMXGKZHY6F4ETWPU7385
EECQ87N42YMVSTUXP6DKJR9HG3ZA50
FFHA4P6VW7UC39S5TRZX8YKQEMOG2JN
GG653DC0YW9N8MKQRXHPS2TUZF4EJAV
HHV9KY08AJG4TEXMD7NWR5PFCZS6QU3
JJUPZ39GKSWMHYCDV0Q46TN752EARXF
KKPVJGDYHZ37A8Q09MFX56U4RS2TCNW
LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL
MMZJ0XHN7DQ64U8AKTER3WS5G9VFY2C
NN3GXA8TUQHSP67EY24JD9WZ0CRMFK
PPY8U625VNTJ9RFSEZWHM0GK7XQD43A
QQFWCKGKHXRJENA0Y38MSVP42956UD7Z
RR745ZWJC6SYQK93FGDEUNM8PTAXV02
TSXN2FPWZE4DJ35VU9RMAHQ0T8YK6C7
UTD0AE726H8W5SU4MFPQGC93XKJRNXY
VUGYNTE6PXA ZV5428SFK0D3JMQC97WH
WV8EP5SR9U6KDCWZ2J3A7MYH4NX0FGT
XWATFV5934PQGDZR6CJNE8HX27MYSKU
XYXW3QHYANCK2UTM8GE7Z9VWZSDR5P04J
ZY5RG0QM83DUE7HXCNAVZS6PJWF2KT9
0ZNU5SWV3J2F0KGR9PDC7TAXM6E8H5Q4
20SZDQKXM9CT7NYHJA85EY263V4GER
32QXE4UFS879ZW6PNV50HKCDAYGJTRM
43T6W9RDGFVXY0JC5QKU2EANS478ZHP
54KH7UTPFMNRWV26A5SQYGC80D3EZ
65M76SFZRT2GCJVW4398NX0YUAHQPDE
760MT24S5AE3RZPF7WVYXQZGDNHKCU98
87JKMNAU40X5FPEH62XC3DZRYD9W8SQ
98RCYMX7EGOP24ANQUT9JZ5VK3WSH6D

CDEFGHJKLMNPQRTUVWXYZ023456789
AAUYR6H3WQLG5D0CZ97FJK8PNV4XEST
CCVQH0A79ZEXGS3WR5DJTF4KLYUP62N
DDNE397G26CU4P5S08XZRQJYALTVWKH
EEFNRLA0JQD7WH6T3C4U8G2YKXSZ9V
FFW5PUKYTG8ELZVJXNQ7D3A029C64RS
GGRA9S58PC7JFU2XWK4E6LQN3HZTDOV0
HH4VZER0CYN5976AQW3KFP2XTU8GLDJ
JJ9GKVFQNX46ERYTPLLZDS7C385W0UH2
KKC9X4PVJ52LNQVFGTY370H6SWAE8ZD
LLXKTZNH6FY73CREJ0A8425SVPGDQWU
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
NNGPJQTREKV30AZLF6H28S9DUX57YC4
PPAWG8XUF9SNTY4K5JV036REDCHL2Q7
QQS8VTYLR4FCA6NZUHEGX579K2DWJ0P
RR8UQLZ6AVT9W3EHYCO PKXSGJ425N7F
TSL675DX80WVUKG234PRHZTQCENY9FA
UT5XFYJZLPU06HQNKERS2DW74G93VA8
VU3D8K4JYSXRZNFV2QTW9C6AG70HPL5
WV7S4FUTQ2PHRLJY8ZN95W0CXD3AKEG
XWYZA3CD5R6PX279HGSTNJUFQVK08L
YXHC52G4KWD TJV8P9FU60E ZL7ARNSY3
ZYD2UJVNZ8KAHETQ4RL5G93WPS7CF6X
0Z24YNQEHUJWC0LRVA6XPGD5F8S9T3K
20FTEA6W7NR2S5C3LD9VYUP4ZJK8HGQ
32E0DGP3439YVFX87UKHARNZ6LQ5JC
43JN6C09DLH82GW7ES5YQVKURTF4AXZ
54072P8FVDGZQTKUSYJCWAEH536RXN9
65ZHW92XA3FK4S9CP8LENYTORQJ7U6
76KJLHEC3TZSD9AGN7WUV4X8QFP2R5Y
87TLOW35SEA48X9D62GQZYFVHNJUCPR
9863SX2KU75QYJPA4DVFACHLR90EZGTW

CDEFGHJKLMNPQRTUVWXYZ023456789
AA8KZF0PY9Q2GR75CV6SX43LUEJTWMD
CCHJKU9W08XY57QEAF TL2RGS3MVDPA
DDGTSKXCQL87F09U4WZMHR Y6J2VPEA3
EE6MJ8S8D9HL0ZQXK3CVUGY75TRPA4W2
FF4XQ0W5PDCMRVA7U6Y8EJT39SLHZGK
GGC0YRVHTAPS4JM25L3QWUKD7Z9X68F
HHW7R2MLJPVKEST36840AZUCYFQ9GX5
JJ76HL3SER2C9D48MKX5YAW0GPFZTUV
KKRH8XEUC34A0WD9JZQG2VPYLM65SFT
LLAY23T8SVMUDKJ4GXE7PFWZR50QH96
MMQF6GRT37YE842HPJLZ0CD95WUKVSA
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
PPXUF57VRQ03H2Y6WGMGK9E48ZDSJATC
QQT4DCU0FSK6P5ZWX7A3JHGMEL2R9Y8
RRUWPV62HF58JLGM73TCZ9XKAQDEY40
TSYGL84KD23WQCEXTU96RPA7HV5FJZM
UT05GH2J4YRDXE3LVS8F7WC6AZUMKP
VU2LX9DZW4EP7ACQSF0H3MVR8TG6K5J
WV9Z56YM2074L3RGATHUQDEHFKSPJW
XWLSUZQA7X9R6Y0FDP5HJ832HK4TMCVE
YXV24ES9UTJFWZKDLQCRM65P3GY780H
ZYKCAP5RGZFLTH6V02MDUX8SW9E473Q
0Z389QCF AEDVYPW0K57L4TM2XJHG U6S
20JECWZ75KUGV6FA9YP4SLHTD832QRX
32ZAVMG3L56XS8HTY4JWJFQ9UP0CDRE7
43FPMTH486G9KXLLJRESA50QZV7WC2DY
545VTJLEXGHQU98S2DKP670FMYAW3CR
65E907A6VCWT2MPY ZGRXDSJ4QK8LFHU
76DQ7YPPGMWAJ3TVRFH29CKSE0UX85LZ
87SDWAFY6UZHM5PQRVEK8LJCX43029
98PR34JXKMTZCUSEH9DYV5FA2670LQG

CDEFGHJKLMNPQRTUVWXYZ023456789
AAQT2USVXYW67GM0HK8NZD9EL4RCJ35
CCS07F6RD5XG249EK8QUMJHWN2L3TV
DDOKCRE4Z7FWLYQ865TVNMSU3XAJ9HG
EEAX69CQ8K3L5TRD7Z4MVU2JG0SWFYN
FFJMG63ERVKT48C9NQDSA2LHXUWY7Z0
GGLJVKN6FU0QRE73M9CH2YZTAW54XDS
HHFRTLYJSQZX0MW24AUCE6G7893K5VD
JJE8L2WAMZYGNXSU5V0RQ96FTDC3HK4
KKY20NX36SMDE9G7ACFLW54ZUHT8VRJ
LL6EZY52JD4VMAHW8USF93KGQC7NT0R
MM85JAXNL2F370VWGK4TQERHZD9S6Y
NN5WMXV73JAR9CKGUF6YHT84SLZQ0E2
PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
QQVG9DRZT3C2HL84FY5XK0UA6NMSEW7
RRMNF984G6HY5DQ3TZ0XAJ7VU2CLK
TSR4HJ2M0TL7KNUAYXVD8EFC5Q96WGZ
UTGFQZ4LH9DASJ5YR2W76KVXE3N08UC
VUDZWSJ0V5H3GKAMLNQ4RC9Y8EF27T
WVZLU0MKGWS9F6XNJ37TY4DQ258RACH
XWCD5HLSU8TNV02JZMA9RF734E6GYXQ
YXTHAV0G72UECFNK635LZQ8JY4DM9W
ZY3945TW2R80AULHQ5J6C7NKDFGXZME
0ZK6D48YLCRUJ2T5EWHG3N0V97XMQSF
204YSMANKHJC63VZT0XAJ7VU2CLK
329QYWHUA45KXVJST0MEDC36ZRF7LN8
43WUN7GC9MX4QD6FVRE2SH5Y0JLTK8A
54N3R8Q5YFES2WZT9HKL7XMO0CGVADJ6
657C8TZHWEQMUSYLDJ23FGXNR6KV4A9
762AK379E0NZ8QFCXDRJWUWYLVSH5G4M
87HSXGKFC8AV8DR360E9WJLT5M2YZNQ
98X7EQDT569JWH4ZCLYNGVAMFK0UR23

CDEFGHJKLMNPQRTUVWXYZ023456789
AAVR03NWD4KSFHU67J8P5ZEY9XCGLT2
CCU8MYEPXHS742ZVLR56DATNFW30G9J
DDR7AW6HF03YGM8JNSL295VPT4XZUEK
EE3WLGZV8F4R5YCHXPA6NK0JUT7S2D
FF7NDHJMGAWPZCLS6YEKT9R2U0458V3
GGN6FMSCZDH25XEYJFPV3UT7K8A09LRW
HH9TWJ8SM36VCYFLUEG7045RAK2XDZN
JJ4068DLSNUZYEHFAGM9K2X537RPWCT
KKGZ279N3PR8W60T5UAECMFLXYSH4DV
LLK38FHGEUDXVZSMWCY0N7246T9RJP
MMTUHSLYCWJRXPG8VZNA097D3K4F56
NNCX7T0U6R9FJ83A4DWZPYMG2VESKH5
PPDFYVZR2SETK7X5G948HWAUMJ63C0L
QQQQQQQQQQQQQQQQQQQQQQQQQQQQQQ
RRHMV5X97EZANT24C0KFYSJWDYL86P3G
TS0AJLFEY685PVMGDZCT3K49WN72HXU
UTYP90KAU54H8DN32W6CVESMRZGL7JX
VUP2TA3D890MLF6WKHJXRVC75ZENS4
WVWHEZC5RLOG79PXM42DJ63AS8UNYKF
XW5936UJHKNEMSD8TLFR4XZV02PCAG7
YX8LCPV24MYN0K5RE79JFDU6GHWAZTS
ZYADSEGVPJL92RCZF5XUW30TH6NKM48
0Z6JGCYX5FMK94VPS2RW8UN3LDAT7H
20EV4K73AX2JDWTNR60K5RE79JFDU6GHWAZTS
32FGPR57KYVU3N49ZT0LMHD8CSJWXAE
43Z5KNT6W27LHJAU98DVXCGE4PYM0FR
54LEX2RK0CP6A397VNTSGF8JZMHD5UY
65JSZXP49GC3T0R2YK7HL86WEFDUVNM
76X4NUA8J7TGSGLWD0FH52PCZKRVLJ3M9
87MCR94TNV5D6UK0XA3GYSHFPELJ2WZ
982KUDWFLTACEGJH3MS47RPNX95V6Y0

CDEFGHJKLMNPQRTUVWXYZ023456789
AACMQXKGS0HWEL4V97JT8N52Z6U3FDP
CCGL5NU8MD3Q9ZT6Y0PS7VXAWJ4K25H
DDHXJEL354S6CNWYGU2QK9P0VZFZM7TA
EE9KSL7Y3AGTNUDWV25HFZMP4Q08JXC
FF2TZQHA480LJSDKXPGVUC5WYMN3D976
GG8ZXV47LHK5YWSJFDEM06NCQPTUA39
HH3NP9ZKXTMJGVQF84A5UYED62WL0SC
JJPH4SGEDFAU537LXYW09MT6KZ8CV2Q
KKU69FQ4VMZE7JXA0SGNT2Y3PC5WHL8
LLZFG7JWYXVCU2ED4539Q08MAHP6SNK
MMLYC86Z95NAKFP0UQHFW7GS2DJVTX3
NNV73U268EYHZ0CTWPMGJ4KXDSAF59L
PPE3TM89H2C4XK0ZNFQDYLJSJUW7G6A5
QQ5C0H9XA6P7SGFKMV42N3DW8UYEZJT
RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
TSM9AGVLEQX23YJ7KWDPZ8CTF06N45H
UTSE2CNMPW5FHF9683Z0JLGA4Y7VXUQD
VU4JY25T6LW90PNCMD8VSAFKEGXQ3Z7
WV60K4AJ79F3WDGSQEL8PTUNHMC2XYZ
XWQA7DE52VJ8TCS3SNLFXH0ZGK9PL64
YXN8HKFVGP9DL7A4ZJSC6U350T2YQEM
ZYF4LWD2UG7M6T35JCNKAQZ9SXH0E8V
0ZW280PQFN6G4A9HTXKY5D7LC3EJMVU
20D56PMHQUTVAXZ9CKFW3EJ7NLYLS842
32ASW53CT7DZPMUNE864GXQFLVKHY0J
43KVEYWUNSLP86527TCX4F9HJAJZDMG
54TPFAXSJZQYDEVGH76MC2U98N5KW0
65XGD3YNCJEO8M2UL6TAVKHQ74F9WPS
76JDUWTCPOY2KQ8H8M59Z7ES4V3LGFANF
870QVJSDWK4N25LEA3YZHP68X9MTGUF
987WN6T0Z3UXFQMP2H9LDJVG5ES4CKY

CDEFGHJKLMNPQRTUVWXYZ023456789
AANWD728ELP5CTHZRQ064YU3M9KJVF
CCR2ETQGJA86YKF5Z3UPVMWXHDL40N7
DDUNLPR7A9XQE8V2WZH3YJF54KGC0M6
EEWRA8ZTCD73JG0Q25FXM4N6VL9YHUP
FFP7OCTJUHE9NY5G8K3D2RXLZVMWQ6A
GGVHT5F3K8QW96JU0NY2ADMRE7PLC4Z
HH6XVA7E0MDGFCZ8PTQ9WN3KR4YU25L
JJZ2ZCG5KYETX49U3Q6N7HVRP0ADMFW8
KKH0G3U69T5RLXYNFW4ZEAV2C87DJMQ
LLFU9XWPK6ZA7MRN2V5JC0QYGT4H3
MM534LXDVY98HARP672GUFQTNJCD0WZK
NN8TUYK4WFJDRM69GLXEQZ7A50H23PC
PPJYXRM276W08ZDV4HAUKGCF935TLEN
QQLDZ0EF52HY3UTCAGMPX947RW68KV
RRGKWMVLV2N4EZHPD9A7J35TC6UFQX8Y
TSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS
UTMV8Q05G7ZNK3CFHUJRDL4WAPX9EY2
VU78FJGYN0CLW43KT96AZ2PDQHVR5XE
WV36MDPAH4LTOE27X8ZKNU5GWYJFRQ9
XWTGN49MRUYA2VXLKDDPC5Q8E3F0Z67J
YXCJ6W4RP3NH72LMYVDFGTE0K5Q89AU
ZYZQJK394CGPMLN65XW80H27FEAVURT
0Z9L2HA0QRVJ5F8EDCT4X6KYPWN37GM
20XPHE8CFVAKUJQT7G5LRW692M4NZ3D
32K9RVDHZWMCQ07ALE8Y63GJXNU5PT4
43AE5UJN6QFMXWKYC49H87DVTZ2PGL0
54Q5Y96LMJK7VDWX3PRTF0Z8UCEHN2G
65DAQFCU3Z046NGJYEYKV7PLM82RXT9H
76EDC3NYWX5UVP94JML0T8AHGQZ7KDF
87Y4P2VZ8XRFTQAHM0EN9KJUL63GDCW
984M7ZHQT2UG5E0VFCWL9YNDX6KAJR

CDEFGHJKLMNPQRTUVWXYZ023456789
AARFP4WC62GV93YQLHKKJNSZ5D0MU7X8
CCND4QYH3GWZEV6K7X9F20SRP8JMAUL
DDV5N28P7S0CYALGJ4WEZXH3RU9KFQM
EE7Y3VX5MCHF0JUZZKRSWAPDL64G29NQ
FF3ERN0DLZSAW782MPG9VHC65XKQJ4U
GGUS8LDW4JFKCQP7NYAZME9X05V326R
HH2PQK6XVWYS5Z39AUEDG80N4LFJCM7
JJ695RSF8VZ7GL0NUD2K3CAYEHQ4MPX
KK8GY6C9X7AMZUH34EJR2LFLJ0WDRX5P
LLEMFD27WRN6QYGP0A4U5V39JZXH8CS
MMYKE5ZJ03VL28SRXFNQ6A7W9C4PUDH
NNHVS0J2DUM47PF85GL3XKQCZ96YRWE
PPZR2GL4A08H6C7WFOY5SUXVNME9DKJ
QQ02WYAKHL7UVXC6P93N8JMSGFR54ED
RR3ZSMNFXUPLDJ0E286HQ4AVKYW5G9
TSQHUM50N9EGD2RJVF8FCKYW4X6A7ZL3
UTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
VUWQ9EVM5638N0Z5HJR4Y7LGKAPDXFC
WVPAHX9Z5QKNJREU6SM74G2DCWL830Y
XWM0L7PYQFD9HK4A26CSJ5EU8RZV63N
YXG4K93UZ60RSVECM5PWL82Q7DFHJA
ZYJ87A46KDPEX9QCG3H0FR5MLNSZVW2
0Z4CXUESRK92FN5M30JDAQWGPYH7LV86
20KXMJR82E5WPGNF5LJDAH96YQU3CAS7V
32XZ08FGPMJQA4DLRW7VU9KHSE36NY5
43D7CHKVE4QRM59XYZULP2NFA806P9F
54SNGW7QC8LX3K6W48R0MUZ2J5EP5W
65A6VZURJHXD8FMS9N0YC4P73QWGE2K
76FLACQ39P45UEKHWWX8DNRJ720SYZG
875JDPGAYN23K6W48R0MUZ2J5EP5W
989UJFNLG5RY4W2DS7PXE36KMVHC0AZ

CDEFGHJKLMNPQRTUVWXYZ023456789
AA34CS7LKV9H2Z6G5EFMTJPPQ8WXDYNR
CCP3V95SAWREKLY74G6QJDNHTX028FM
DD7G20HXJKZRTWPEM3S6Y59FAC8N4L
EEVCGFKNH76JQPZ2AD089RWTS54MLXY
FFR96KL2NYAXPDEZS0HV54MW78T3GQC
GGWV76AFE5YDHNLC2ZTRMXJ943QS08
HHCAEN2PQGFTM30DKJXY9V8L75RZW6
JJGEDXQWT2098V3MHR4LF67SNKAYP5Z
KK45ALGZ2CSQD0FE7HNR8T3MYVWJ6P9
LLT8SG6EZ97POHKFYN24VWJ3CRMxAD5
MMK2Q3J4RHPY95WTD8VVFZLA60EGSXCN
NN9SF2ZDP6KW3JH0LXQC75RVGY84EMA
PPSLND0J3F2V4TQXZWMAG79CE6Y5HRK
QQAKHPD3MEN8R4XJ2TW6LSCYZG790VF
RR2DM4T59Q36S7V8JYCN0ZKFXHELWAP
TSJT97YGLR5NZE68FK3WXDPVMQ0C24
UTEHJWMV8DXYC4RQ95ZNFGLP2K6370
UUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
WVNPWR49CXMGAS8537YHD2FEJ0ZKT6Q
XWFXNM3RV0Q7C9T4P58E2K6GDZLAJYH
YX6F0QPMWZH5VRJ3N4TGKAY72LSCD8E
ZYQM8C9A6TVZFK7SRLGX3PH04JDN5EW
0Z8YLEFH0SG3XQ2N6PD5CVT4A9RWKJ7
20Y6ZHNQXLE4WMDPF3J7AC85KS9V2JT
3257KZE0DALMJXNHGQP9Y84R6CVTF3S
43LZPJXT4ND58MW0VRKEGSAHF67Q92
54Z03T85PJA7YRVC92HELGKQNGMSD
650X48VY73TKG69CWASDQHZ2MPNERLJ
76MRYASKF8C0N2GL9ZEW43QX5TJPHV
87XW5YC6G482EFSAVKCLJMQ0DR3PH9ZT
98HQTVRCYJWL6A59MS70PNEZ3D2F4GX

CDEFGHJKLMNPQRTUVWXYZ023456789
AAWC6L8KZXJNM73SFTQP25Y4EDGHR90
CC7UW2DEFY5A8R0XG6JMNTKZ49PSQLH
DDP9GQW0XU486J2CYFETMZ3SH7KA5RN
EEN42X06R5DK3AFJDL8UY9T7WHMPCSG
FFUGCEXR8W2ZS3J6DALH4N7MQY9T0K5
GG3PU4YQD7NFX05W9C2SZAR8MKL6HET
HH8SMUNG5Z702DY4TQWL3RFJPA6E9CK
JJH50FE8LQCPKS6R23AYGUM9D4N7XZV
KK2ELS3T7J8YUNZPR9MCD5W60QGAHF
LL52J8RSK0F97TA3EPZWDGHYXQ4U6MC
MMF8Z7TUHAKQ5G9NS4YJREC036X2PWL
NN6AT9MY4SP2QWUHZ5GR LJXEK8F07D3
PP0J3ZKM9RAGYHT7LUNXFCQD8E2WS46
QQZM4W5CONYRJRFD2HEXP7KA3UTSLG69
RR4QE6JA32X7PZ8L0KSGWYNUC5H9FTD
TSDX83APTFRHN9KZ6M720QG5JCW4LUE
UTX6SPZ9N8354Y7MAHUEJ0D2LFCQKGR
VUR37N94GKTCDOHYPW58A6EFZLJXM2S
WVVVVVVVVVVVVVVVVVVVVVVVVVVVVVV
XWK7Y5G2C9H6FEQDUX0ZTSLANP384JM
YX9YD0CJ6GQSALEFW8RNHMPT5U7Z234
ZYLYK9HU5WPMXC24G7DQAS8J6T3RFN0Z
0ZCFAKS7M6L4HUP28N90E2WQRXD53YJ
20MHQC2FJ4W3L8XE5R69U7ZPGNTKDAY
32TN5DQXEHGLR6C04JF79PSKYMZ3W8U
43QORALZPE6U9MSKJ7TDCW4GF25Y8NX
54AZNYHWQT9E0CG5M2D3KL6R7S8JUXP
65STHG4D2MUJEXWQN0CKP38L9ZARYF7
76YWXJFLAD0TZK8RCS345H9N2GUMEFPQ
87ERKTPNULSWG4M93YHF6X2CAJ0DZ58
98GDFR63SCEMTPLAZXK5Q4UH0WYNJ72

CDEFGHJKLMNPQRTUVWXYZ023456789
AAKGS9PJ5DT8UFLNMRQV024XYZH7CE6
CCXVYSTALODFQ5P96JNHK483G27EMRZ
DDE4Z6GTSHVNA9YMKPC87UQR23F50LX
EET9QUF74L5ZK283JHXSPM6DNCYGRVA
FFYJCJRQ839NKHXUEL47MSD0GAT6Z52P
GGF3K0ZYM42APC6DHSTU8RJ5XEQNV97
HHLQ3X4VZF8MD62KEG0N5ACPUJ9S7YR
JJ0Y9NLRFTP4385QCEUGDZ2KS6VHA7M
KKJF42H0GE7SMYVZ3D65RN9A8QLPXTU
LLV6CA95QYS3EUNJTFRZGKXHM024P8D
MM3HGYDCPK05NLTSA97X8FUV4ER6J2
NNZDPLAQRM78EJ5SUF06VH2TGKX93Y
PPHZMCSLNGYURQ9AD5J2VX376K48TF0
QQ6TL5JUECAH47RF938DMGVZPY0KNXS
RRDSNQ5E8PL2X4FUA73YT6Z09MGVJHC
TS4KDTM9AZ6R5JCPGNLX27E80H3UYQV
UT726MYP9VGGJNSC0LA4H3UEZX8FD5K
VUMP5FR37AJV2HE8NX4TCY66LSD0QK9
WV5UXK2G684CTMZ07YDQFJAL3RN9HSE
XWWWWWWWWWWWWWWWWWWWWWWWWWWWW
YXA5847KVREY6GH2U0ZLJ9SCFNPT3DQ
ZY8X0D6SC2ZJLAMTV9P34ERFK7UQGNH
0ZQEHVK6D3XPST0G4MYRU5LN7FJA2C8
20R82ZVDY7H9CSG6XTMFEQNJ4U5LKP3
32NR7HXZ0U3TYDKV86GJQLP9E5AC4MF
43CLF8EXHJRGZV74QK2PASYM59TUD0N
549JE732KQUDG0XHFZVANPTSRLCM865
65GMAJNFUS9X73QRP8E6Y0KVCDZ2L4T
76U7VG0MTXKLR9PDY2CSE3F5QH8RZA4
87PNU38H25F60Z4XRVK9LCMTQASVYEGJ
98SAREU4XNQ0VK3752HC9TDYJPM6FZL

CDEFGHJKLMNPQRTUVWXYZ023456789
AAD63KEZ7G59Q8WLP2Y4MFRCTHSN0JU
CCA79RND42JTU6SFQMHWZK308LE5VGY
DDZ8RFSM6JN3PT4HVGU72LKA9YWECS5Q
EESHPO3WL6TQ2YKAG7ZF4CVNUDR958M
FFL2NSCHMQV5TGD49U6ZYWEKJ7A0RP8
GGJRHU653S4LAKTPDE09NQY2FV87MWC
HHYJS4DUGVCE35M6RPT2Q7WLN8ZAF09
JJ5KYQ8NRW7HDF9VZSC3EPUGL0T624A
KKFM5E0LZUPJ82AWTY7DHSNRG4CV3Q6
LLHGEWAY2P0N9JZ73Q8MU4SF56DCKVT
MM29FH4GTNSK036UC5P8JYLZRQ7WDEV
NNELQV9SF78UMHRC24DKW0P5YA3TJ6Z
PPVS6TG0EDM7LW53HAKNC98Q4RJ2UZVF
QQPE782VNAZ4FSJ9LCR50T6UW3GMYDK
RRKZJNVFDYQGG6MCS8H4ALE532W0P9U7
TSWYVCR4H89PGUFDDJ6ML7A0EQZK3NT2
UT9CMGU30FHZWAP54KEVRJ28DNQY6LS
VUQN46MP5CDWKEGTF03JV87YS92ZHAR
WV0W89JCSZ26H4NRYDFEA3TP7K5GQML
XW4U0AK7YT3VJQLZ582H6DCSPMFR9G
YXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
ZYU5W7ZQJ0ASRN28KV9GP64HETMDLC3
0ZMTKLW285ERV97Y0JQ6GHFD3U4SANP
20C4T35AWMG8Y7EKUZLSDR9V6FNJJP2U7
32G3LY7J9EWFCCR8QANVT5UHMKP64ZS0
43RDG5PKAHU27Z0E6LWCFNJ9MSVQTY4
547QCDF6U9R05PHMTGYZ8AWV2LKS3JY
65NFUPT EK46YZL30MWARSVQJHC98G7D
768VDMHTPRFAE0UGS35Q92Z7CJYL4KN
876PAZL8Q3KCNVY2E9JUTMD40GHFWR5
98T0Z2Y9VKLSDSCQJWRNP3GM6A5UH7FE

CDEFGHJKLMNPQRTUVWXYZ023456789
AA42MVLXGRD38SQUNZ9E7HJTKC065WF
CC9W0E4FRPTLMD72JNKZ8GQU6SXAHS3
DD6HFNKLVE29XUM57QAJ0P8WCT3SRG4
EEG8T4H2F3J5DNC76KR9SXAQPZUV0MW
FF86E27NDT4QVLRKH5MWPSG903ZXCAJ
GG2NAXUS8MVT6P9ZL3WFK74E5RCHQJD
HHUZ60TC78PDKR4E3F2X9QLVWGA5JNS
JJVXW6PH498R27T0SCEAULDmZQ5N3FG
KK3T7RFM5HCXQANDEVLPJWZS46892U0
LL0CN5MQU2K8Z9VARGXHETP6F4J3DS7
MMJ4RDNVACFZGX5L2UQTH6W370P8K9E
NNP02KR5L47GUQDMCAV6T3S8EJWZFXH
PP5QS3WT0XZ2CE6J94HLAMKNGVDR87U
QQEF5AVG9KMPW8UXDSZC24T0N7HJL3R
RRWFJCF2DM0EUAVKNA4L275689ZHPVSG7QT
TSK5XZ93PVU40T8WQJ6NMR72ADFCGHL
UTAG3J64EZWK20H87CQXVM5SULDPR9
VUCRLQA9ZN563WXGM8S7FE0HD24TVPK
WVH7DL5UXFNWSZAQK9G4C06JRETPM82
XWDV98S6JQGC4H3PX0TMLNFRU5K2ZEA
YX7KVUQZSDLJ3G95W82RCH4MFE0A6N
ZYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
0ZRMU9GW3LQHTJS8A6PKDFC7VN2EX05
20Q9PTJEC3NRFH43AGWMDKT2FH8YS9
32SP47CKNJHAL5FR0MD83ZXGTW9UEV6
43MAZWH8JTU97E4P6GH05VDKXLFNSCQ
54XSJH072W6MNKECPRFG5UVA39QLTD8
65TEKMDAQ7RS9GLVFXU04J3P2H6WNZC
76LU8P30HGSF7CJT2FE4VQ5ND9AMKW2X
87Z3HCERK60V5M2FTZDNSW9UXJ8GQ4LP
98NLGSSZP6AXEH0W3UTJD5K2FQMR794V

CDEFGHJKLMNPQRTUVWXYZ023456789
AA5N9T42M70YWQJVE6XD3UGHCF8PSRL
CCYH37L9SUF2JAVODT4KGR6Q5PWXNE8
DDMWH5UNLYGS0VFP3C7XQ2AJK6ET89R
EEF74JAX6VSPYR2M9WQ3LD8U0N5HTKC
FFXR8D5L7KH4903NGVC6WMJEPQ2AUSY
GGT2RPKU5XW7N3H8QFDAE4096JSVYLM
HHAMYGP5D6RCLN8UW3FJ2T9SQE40K7X
JJDLSATMXC9KRWE20Q6FN5H8V3UG4Y7
KKSJQYRH826NFDPGX5U4A9CVMT07W3E
LLWPDN3V0H5JT47CUS9RKQMX8Y62FAG
MMNVA2EQW9THPKX64YRLC35DS7FUJG0
NNQK53FCVGA44SL7890WY62MHRXEDTP
PP4EWKY8UMQL3FGH6D5TJVS0XA9CRN2
QQCS26XYKTE58HWRJGVPV973NA0LFMU4
RR0TXWQPGJMF5UYK28H94VL7ESCN6DA
TSHDC90AJ37QXM4TL2E85GYKNUPRV6F
UTU304SE2LVRQ6AJCXM5F8PG7DHK9WN
VUE6P8HF3WK0C75DYLN2XJ4TRMASGVQ
WVK8NC7S453MEJ09FATPHYQWDGR6L2U
XWV4MQ6KPA2DU8RYEHG0SCNLJ973X5T
YXL0JM2WRS8GP6QTKY7VND4FC35EH9
ZY9A6RWGHEX3D5KPMU8ST07C24VLQFJ
0ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
20PULVC4TDNX2E9S3JAG8KWRFFHYQ7M5
323CTEJ6Q04GKYMXSRWN7FU59LD8APV
426YUFD7CP8TS9NLH0VQRXE2GWMJ94N
548FVS9JENCW6XTA7M2UDHKPL5GY0Q3
652QGU83NRP9VCDFK7LM6ETAYXJ4H0W
7679EXMRY4JUHGQWAPKCOLF3TVND28S
87RGFLN098DEATCV54SYWPWX6UKQM3JH
98JXKHGDFQYV7LU5RN3EMAS4W2T9PC6

CDEFGHJKLMNPQRTUVWXYZ023456789
AA6Q4GZSLUNDTKXCXJ852E9FM73VWPHY
CC2YPAGL574RHTXSNK83Q6MVJE9FZDU
DD4A8HTFMX5E26RWL9VPGN7S7SZJK3C
EE8D932U7WVGP4QVMNJKH5XSFTCPJDR
FFGSHWR965TUQEMVK32AXN48CLDZYL
GG9ENZPXSYJHK8AC75L63VWFU2MR4TQ
HHNG5TKWFC369DRSVM4ZJYUXP7Q82A
JJRVQ7UPZ6ELXCN43AGWMDKT2FH8YS9
KK7PS85DRGX6VMTHCFWJ4UEQANY3L9Z
LLQJASXKT4GMWR58ZDHY7E62PU39CFN
MMALDFW628H7YQV9TE3CSG4PKXZNRU5
NNW9YJ7ZG2Q5SX4PECAFVTRTH3MDKUL6
PPM274NACEUK5LZGYVXSXV6FDRQ9WHJ83
QQKR6E37JF9AZPYU4NTD8SLMH5X2GW
RRPCKDHMVS8Q32WF569ZA47JLGNUTEX
TSE7GXC8KNZFRDL5PHTQU3964Y2VAWJ
UTJZLK8RWAS29VHDXMFT9QYC4UE56G
VUHF3YQN4V2XAG7J6ZPDWT589RKLECM
WVC5RMF23KDJUY96HQEXLAPZTSG4W78
XWZXRTRDV9LKYE3FM826GCPJN5A47HQ5
YX3UZCA58JPWDHSL4TKEY2V9NQ6MGR7
ZYTW2QEJNM6CGZU79P4HRKL5VD8S3AF
0XV3JP4CXQ7T85GAULS92MRWY6FDNKE
200000000000000000000000000000
32LTM69QYDFPNJ3EW7U5KSAACR8XGVP4H
435HV26YURMZ4NEQFJ78TLCXWKS9A9D
54F6UNJGA3Y8LSPZQXCM9WHDEVRT752
65YNCLSTHPAVFW8KGRDUJQ23Z7E6XM4
76SKF9VEQH4J723RUYL8XGAD5CZMNT
87DMEUY4P93SCAJN2GZRFH8K6WT5QXV
98U4X5LHDZC9MFKTAWR7NY3EGJQ2SVP

CDEFGHJKLMNPQRTUVWXYZ023456789
AAOPYCD63MSXJWR4KNZ59L7FHUQV8GE
CCL9SD846EKGWNARHZ30UQV7M5FTYJP
DDQUK8YR4PHJNZCAM36L5FTVE07XSW9
EEJVV5P9KSF0RCDMHL8YGTW3ZQXN6UA7
FFA3X7V05NGHEPQLJ9UR6CY8W4DSTMZ
GGS CZJWV7R3U0LXT6QFYDKEM48HPN5A
HHXF9MEY8LU6RAKS55CDT7GNW0VJZP4Q
JJKD3WNTVA65LQGX4F7S8HPERYM9Z0C
KKTQPHM8D0934RSYUACVFXWJ57GNE6L
LL4NVQFU9JTSHM05XEP6ZRDCG3A87KW
MMG7UEPSYQ54ACHK0D8XVJZNLTW39RF
NNMY4Z3GXDRLF7WJAVTHSEU9CKP56Q8
PPWT09UHK7LAD8EMQYSJXN63FGZ45CV
QQRZTF75UWXXMEL0GP943A8DJ6CYVHN
RR5E8AC3ZHYTGJ46SWNUP0FQK9L7DXM
TSVLEKHD5C5PZ64Y89RA7QTJGUFXWM30
UT8RWXGFG6NPU5V7Z0LDAHYHK3CSMJ94
VUZGQ50EMTFDYS9P7KHNJ3R4VW6AL8X
WVD4JTXQL3WE9U7FN50CR8KSZAYHGP6
XWH86NZXTCA40QFJGR7VKYMPASEU3LD
XYANGJ7F4Z950TV3LQ8CSMH6DKEWUR
ZY70MSKCAUEN368DP4RFLVGX9QTJHZ5
0ZESR36JG8AQ7VNWCTXMKP5UDH904FFY
206W7LQ9PGVYKH5UTME34CAXZLDFSFJ
32222222222222222222222222222222
43PKA64WJYCFVTVZNDXGHEH9058MULR7S
54UMDRAZANK8VXG63YJW9E5QLSP0FCTH
653JF0LPEX78SKU9VHMZ6ARTN4CQYG
769HC4ARNWSD7TX3Z8GJPMUL0YE5QAVK
87C6GVTL0ZJMP9FQW5A4DSYNR8KXE3
98F5HYSAR9MWZ3DCE64Q07XTPLVKGKU

CDEFGHJKLMNPQRTUVWXYZ023456789
AALXGNF9JWE705SH2YMURTOVQPKDZ48C
CC0ZUXR2EVLNSNFWPT4G865JMD7Y9KAQ
DDSM6VQUK572WCTL8XRHJA4F09NGZPY
EEZTC2PFNUXM9HG4RSAQD608KV75WJL
FFDK24GSHXP0YMNAWJ9TUV6ZCLQ7E5R
GGQLWE2NAKCYJ94FXHSVTZ87RD60PMU
HHKWFSAAMY24Z789JG05RCUDTEXLVN6P
JJX2A9H50GNVZ6MYF78CPRLU4WKT SQE
KKM8DULCW6VFGEXQ2PY0JSHZ59AT47
LL95QTD RX8ZG2PUK6WCJYHNA7MSFVE0
MMCESJ9084ADQZY5N67W2XUKFPRLHV G
NNTREF4H9C285YASPMJLKDZQWUV6G0X
PP7VRWCG4TK9SA2EUNF6Q8Y5LZ0MXHD
QQN98Z6TLM0WXRVD5KUAHFEGYS427CJ
RRY7TKUWPZDN4GXCV258MH9Q0JSLF6
TSUC4ANJMPG680H9E5YKXLDV2RTQF7W
UTHYZDVKR06EPWLU7CX9MSFN8JA4Q25
VUJ0VLTXC7Q4E2KRZPWM59AS6YHNDG8
WVAJ7QZLUY8PCXDT0RK9NG45HF66WM
XW8QKCEGDUHANP2LF47Z0MYT65JRSV
YX56LRKP2QTAF4CWDGE07Y9JV8MHUNZ
ZYWGHMJ87FSTVQ50AZ6PECKRN2XU9D4
0ZFH067DTJ5CRKQVYULNS42EMAGP8X9
202FJ5Y6ZA9UTD87H5QLE4PCCSGWRMLN
326DXPW4FLRJHSEGANZV750UQ8YC9T
43333333333333333333333333333333
54VUPGEASRW5MJFNC9HDLQ76XTZ82YK
65P49YM76NHLDV08SQZ2GWRXAECJTF
764S578VD9YXKUZQMLTFAGP2JNEW0RH
87GAY80QVHMRUL6ZJTD4NEWP9F2C5KS
98ENM05ZQSJKL769DVG2CWH4PXYUA

CDEFGHJKLMNPQRTUVWXYZ023456789
AAX0UW9DHCZLPNV385EYF7TRJG6S2KM
CCKTGM0ZU6Y3AWJ8X92LSPE57QNHVFR
DDFJW3V5NZ9TQLHE2KPOCG7XUMY6AS8
EEMZVHDPK2A6T55NRWQLC80YG9JFX3RU
FF5L7GYCJS6W2U0MRZ8NKE3DTPHVX9Q
GGEK6YXMCQR9UZFO2T8J5PHV3SNDA72L
HH08AD3NPURWSQE59LKMJFXY2CG7VTZ
JJLRFECMU27GDVA8ZYW9QTK5NXSPE036
KKZWEPNSTVHGXLQD6RU98MC32J05YA
LLUA5KPTR3EFYXQSH762WZCJD98MNGV
MMPSY0F8ZRXXVW9CJ72UKQNH6L5DGAT
NNJ2D5ELQW3X6RPKVT8UCF0AZMGH79
PP89HN5QSADY76KL3RTZ2J0MVUCFEXW
QQ2VNLKR6D50GYSTEX79AUJ8HWZCPF3
RRAHLTSXY5KJM067PFGVDWU2N39ZQCE
TS93PQL67HNMFGTR5YXWV28ZEAUJK0D
UTWDKSQ7XEP0CFR6NGYA39ZU5V28LMH
VUTXCZ8WAGM5HD2903VR7SKLF6QPJEY
WVY2AWHEJUQKP3DZN5G0XR68F7T9LC
XW7F923DM8KN5AVJEHXG6STCYRQUP0
YXDNT76F0KSU8JYGCMMH53WALEV9RZP
ZYHPRX70MLT2Z8GFSJCEMDAVQ53W6UK
0ZS7M8J9WY0ED3U2FVAT6QPKGR LNCX
20NQXFGJ8T7A92MC6UZPL5DHRKE3YWS
32RYJUZA VFCNEH9WMD36XTLQ07SK85G
43GC9VAE582SLKDHUPNFMY67Z0XRWQJ
54444444444444444444444444444444
65CU3EHL9V7RTNPAFQJZMGFW80YD62
76VEQRTYGNL8CM7XK0S3HA29PDWUSJ5
8735S6RGFPQZJCYXLM0DEV9WKHA2T8N
98Q60JC29XFH3VZUGAWSRLNPYTK5MD7

CDEFGHJKLMNPQRTUVWXYZ023456789
AA28ERX0UF64KVTYWSDZL3H7GMPQNC9
CC6RDKVQHTUM234J0P7EXW9FYNG8SZA
DD96TU0KCNAPHQSVRJA4FW8ZMXGL2Y7E
EEH2F6WRAM9SU0NX8YT73QC4LPJKGDZ
FFCHF86EPZYARGW2XN4QKDS3JVULT7
GG47LF9EN3M0TAW6ZRXJHCSV2QKD8YP
HHR0CQJ32DKF8L7PVMZAYX6ESTNW49U
JJNTV4C7P0S8MZQHD23XAEGWUR6FKLY
KKWXUVSJA0Z3PC4Y7H6NG89TEFLD2R
LLS43MZFGQPRNE8976WVCDY0HKUT2XJ
MMDCPZ29FJ7XE6L8HWGSKUTYQV0A3N4
NN7ZGE6ATLFVDUXR90YP2H4J83QCWSM
PPTDJ7HZMV4WF932C8LYUANXK0REGQS
QQXYKJ4P3UV9LMH7S2ZRTNW6DAEGC80
RR3L6XNY09WCVSA TGDU2MPQHfZ7JEK8
TSFEYDUC4XT37HVKAQJG69MLRW8Z0PN
UTZ9NARUDGEJCKY06VSM827PWL3HX4F
VU8WA0YVKER7QJDSX4C9GL2ZNF3TH6
WVGN0SD4JRY2P7KCTHQWEFL8A69MU3X
XWJP8GFNX2LUYT6EMAR74VKZHC903
YXPMMWNETY8GKSDRAFU03Z7JQ92H46VL
ZYMFXTADSWNQ4C0UEKVL9ZP36827RJC
0ZUK723894HN6WMLQGFVDV0ATJSYRPEC
20LGRYTSV6XHJ4UDNCK8FM32E9ZPAQW
320VH3PL8CQEWGZMJF9USYRA4DTX76K
43YSQP7MLKJ6GF2Z4980D2XRCUANHWV
54EASCKH7YDLZ2JQU3PNR6FG0XW9VMT
65555555555555555555555555555555
76Q39WGX8D0YENLTAHPJKCM74VFU2
87AU4HQ2ZSCG98P3KLMT0RENVYX6JFD
98VJ2LMGWH3AXN9FPE6K4S0U7CDYZRQ

CDEFGHJKLMNPQRTUVWXYZ023456789
AASVXH5N938ZRJGDT20YEMUFLWKQ74
CCQGGJZ7UWNPV10KM2SE5AXFT9H83RYD
DDL35GA2PTQKZ7N9XRJY40WE8V5UHCF
EEDPKWULYRC8F3S0Z4VN2G5H79AQMTJ
FFZUY34XQELNGAT80H5CM7PJSKR2VDW
GG04QAH92FXC7RDN8JPLV5UWTYEM5Z3
HHEY85QDN4T7JPAFG29SLWK30UCXRV
JJFQNP2ZCHDSWUR7GMKTX3YVA84L9E5
KK5DRCZWE9J4YLMUP0SHGQT82AXF7VN
LL27W0S43CU5X8YVMTFFPR9GDKJNAEQZ
MMHN7KCES2R3VYUWJL0AD58XPQGTTZ49
NNYFHDGJP85MCZ92Q7RV3LESX40WAKT
PP3XD297Z5GEUMJRAKCF84LYHTV0NWQ
QQU0FX8AGY3J295H4NDWSMZCVEK7TPL
RR259JPCKAN0EW7ZDUM8QFV4GX3Y2SH
TSNJMEWYV7KXTF0LC349PDHAZ2G5U8R
UTCWVF3Q5SY9DG8XLAHKUZR0M7P4NE
VUA9ZMKS0P7F4VWERYLGNHXQJD58C32
WVJCSYLFTEA5Q43WX8RZPN9U72D0HK
XWG2CUM0LJZT34ES7VYD9AQ5RNHXKFP
YXMS38THAL4P9NQ5VDGUEK7ZYWCRF20
ZYPZEL03FKWHQXV4U8TJ72DNMR9GS5C
0ZXAP7RMUD2Y0SK9EWFH83FN5T4JLG
209RUSEV4ZMQ8ETLYK732JNAGCPDHWX7
3248G9NR7QAWMKPJHCZ3TV0L5FYSDUX
437ML4V8XW0DAHFT5SQZKR2PECJ9YGU
54RK0VYT8USGH53FEQ7CJ92WZPNLAM
65WLTQXGVDVFRP2HA39NE0UCK4SMZ8JY
766666666666666666666666666666
878H2RJKMG9LSEZCNWUX5T43DQFVP0A
98KE4TF5H0V2NDXQYGAMWCR7LUZJ39S

CDEFGHJKLMNPQRTUVWXYZ023456789
AAF9J2CMQ8URD45TX3GHWVSN6PEYLOK
CCEPXLJ5T96D3G2A8QMVHNZYKFSRU4W
DDXCQ43ZYAMHVE0RTNSUL69K5J8WGF2
EE5GZNSHP4RX8KVFO9WTQALCDM2JY63
FFM4SVEW90YJX6HPZ8KQ3T2ARG5CNUD
GGKU5QMD0LAESY342ZR8X9HPC6WFTNJ
HHQDNFV9KRSLUXPWY68G4MA5Z3T2EJ0
JJSF8UX2APK3QMLC9T5NVY0RWEZD6GH
KKDYH9WXUNF52C86VLJ0Z4QGER3MPAS
LLNH6JUA5W84GQC2KMTFESRZ9VY0X3P
MMW62T534UCSZRQGL0D98PVFJKHEAYX
NNAQRSYFH306K9EVDWP5M2JL4TCUZ8G
PPG0EHFK8ZNCJUW9SX63DQ5TY4MAVLR
QQ9XAMT4DJ2NYZG3CR0K6WFHLC8PV5SU
RRJA30DSNTGWHFYZMVENL2U86MCLX4P5
TS2M0YZVFGD89WNE4PHATCUJ35LXRKQ
UTP8C5AG3XLYR0MQJD4WKHEVU9FN2Z6
VUYVXK6C2H9GMTJLW5ASEZD0PNR48QF
WVT3YENPWDZU68FHRK9MG5C20QALSX4
XW3RVPH86YE2LJ9KNUX40GTMSDQ5FCZ
YXZE968LCFWQT5UJPA2YNR4DHS03KMW
ZYCTDZREVQ4KWPSN3HF25LXUGAJ609M
0ZL54R0NEM39PHYSGFVCAJ6XQ2U8DWT
20U2GD4YS5QPFVZRZMENVJCKX63KJ
32VWUCLTMKX043A56GQFPEYS8HNZJD9
438JTGQ0RC5VNS4DAYZ6UKPW2X9HML
546LM3GRZ2TFEND05SYXJ8P9AUKPQVC
65HKLA2QG6JZ0DTMU43P9FNEXWVSCR8
76RNW8KJLVPM5AXUH72CZS034FYD9TE
87777777777777777777777777777777
980SPK9UJEHTA26XFCLRYDG3VZ4QW5N

CDEFGHJKLMNPQRTUVWXYZ023456789
AAPJ5M6Q4EV0HCF2GWSNDXL93YTRKUZ
CC3KQWS07M2PT9HUFILRG5EDJAVZY4X6
DDW0XYNE3RTMK5JZ9VGCUS2QLH4FP67
EEULYZJV5TN2AM3GP6K0RHSWX7C4DF9
FFNZKAM4S357UH2QVCWYJP9TGDXL60E
GG7TJ3EK6PD42FV5YAMR90CHNLUWZQX
HHG64CW7RAQNXTU029LVK3JZF5EDSPM
JJC7PDY3GLXA6KZET5VH0WQ49US2NMR
KK9N35VAFDECS46MZQ2TPL07JXRUGWY
LLMQUR7XPSHEJD9TCYNA26V5WFKG0Z4
MMXDV6K2QZGUCWAF3S4PYTRLLEN975HJ
NN4H9PXJZ0LKVGYDR3BESCQAF7W2MT5U
PPQCLXTDJUR5G3NY7EZ4W2MA0SF69VH
QQD3M2FWCV6L704SKUHJEYXP5ZNTARG
RR62H70TX4CZDY19WNPMFKGVSA53UJQ
TSZVF4QHUKATLRWCM70EGJNY63DP295
UTFS79LNYC0GEZXPUDJ24AK6HQM5R3W
VUVMSHCRLF4YPX07QT956GZE2K3JWNA
WVRXZG36MNSJQ25KDFALT7HUY90CE4P
XWE52S4U06FFX9LCHAR73VZYDMGJNQTk
YX2WRT9YDH7V3EPN0ZJQSF6MU4AKLGC
ZY5UTNPZE7965VDJLGS3WH4F2RCQAXK0
0ZHRNJDDGV9PFM6E3XK5U7C4ST0WQYAL
205AWUHL92SDNP7R4XTKMVE3Q6GZCYF
32YE6FASWGR0UQ45HCDZNTXVJP9M73
4309DEZ5KXYQFAGVNM67LUWCPRH5J2T
54JGAQ2CH5M9R7SW60UZ3DPNKEYXFLV
65LPEVGMAYZ4WQK6J2F9XRU0DT7H3SN
76TYGK5F2J3HWSMAE4QXN97RZPL0VCD
87KFC0U9TQWJYNRSLSPX6A53G4MVEHD2
98888888888888888888888888888888

CDEFGHJKLMNPQRTUVWXYZ023456789
AAEZ7PUYRJ4MS08F6X3LGCWKVT52DQH
CCFX25KGTGSN4VJELMHZY38UPQRADW60
DDKLV74A8GJ03RUPHYF5EWMNZC2Q6XT
EEY0WC5GZQ276VLTNJHRXF5AM38UK4S
FFTJU8A3X6D2MQYR7S0GH5L4CZEBKPV
GGXMCTLJV25PD7ZHKA660Q3FYWSR8EUN
HHS2LZG64KE8PUJVC7D9QN0R35MXYTAW
JJQWTHZ475LFAPV6EUDM2S3XCNO8G8K
KKARMW2ELXQVHZ5CSGT8YP7D0FU4NJ3
LLRSKECZHMWD46TG2VJ30YA8NXFP57Q
MM7AXQSWKF3GLEN2RC5DP4JVYU6H0T8
NND80MQK5YXZTL2W3ECUA7V6RP4JSGF
PPCG4UDFYH6Q0XA8V3RET52WJLKN7SZ
QQ4P3S02W8RTECMNY5K7U6HJFDVZXLA
RRZ6AYF0S7PK2N3XUMQHVGEJDJTC8W4
TS6UR0XN2AYL5CQMFKW4DVZH87JG3EP
UT3Q5LEHJNKU74GZ6VXSR8F20YACDM
VU5T6D78FZVSX3PAJRYCLKN2HEWM40G
WVMKGGJH7DCTY8A64LPUNWQX0E2S3ZF5
XWYPQ2NCE3S3ZGK50TLAFU47X8D6MHR
YXJ7F3RQMU8CKW0SA2NV4HTGP6ZLY5D
ZYGVPF8X04UWNMR3DQSZTCE7HL5A26
0Z0NEGTV6WCAUDHJ574SMXYRKQ3FLP2
20VDYX3MNPFE5K5SQ8W267JGZA4HTRCU
32UFSNM5CR0HGTWKLXLEP8D643A7VQZY
43H48RYSQDA5W2X0PNMJ6ZLTUVGEFK7
542CH6VUPLZ3YF7DG8AW5NSQTKM0JRE
6583NKWLT0M6JHCEQZGFRADUSYP72VX
76N5ZVJDUEGRF847TAP2K0SLWQXHYC
87WEJ46PATHXRYDUZ8K2CQMG5NSV3L
98LHDAPR3V7NQSFY40XTZEK56GCWUMJ

# 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).

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

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

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

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

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

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

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

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

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

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

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

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