
ᳵᳵ. ᳵᳵ ᳵᳵ ᳵᳵ ᳵᳵ ᳵᳵ ᳵᳵ ᳵᳵ ᳵᳵ
lel-ᳵᳵᳵᳵ-ᳵᳵ ᳵᳵ lel-ᳵᳵ ᳵᳵ jeᳵle-ᳵᳵᳵᳵᳵᳵᳵ
aaaaaaaaA, the language of Rymako

uruwi

|||||
aaaaaaaaaaaaaaaa
A complete grammar

21 November 2017

Dedicated to Isoraķatheð.

Branch: canon

Version: 0.1

Date: 2017-09-12

(C)opyright 2017 Uruwi. See README.md for details.

Contents

0.1	Introduction	5
1	Phonology and orthography	7
1.1	Phoneme inventory	7
1.2	Hacmisation	8
1.3	Phonotactics	9
1.4	Stress	9
1.5	Vowel harmony	9
1.6	The script of aaaaaaaaaA	10
2	Syntax	15
2.1	Basic word order	15
2.2	Questions	15
2.3	Multiple clauses	15
3	Nouns	17
3.1	Number	17
3.2	Case	17
3.3	Noun classes	18
3.3.1	Countable	18
3.3.2	Measurable	18
3.3.3	Uncountable	18
3.4	Definiteness	18
3.5	Declension table	18
3.5.1	Countable classes	18
3.5.2	Measurable classes	19
3.5.3	Uncountable classes	19
3.6	Pronouns	20
3.6.1	Last-clause pronouns	21
3.7	Compounding	21
3.8	Possession	22

4	Verbs	23
4.1	Aspect	24
4.2	Obliques	26
4.3	Conjunctions	26
4.4	Subordinate clauses	27
4.4.1	Conditions	27
4.5	Comparatives	27
4.6	Ditransitive-like constructions	28
4.7	Transitivisation	28
4.8	Clauses with nullary arguments	29
4.9	The copula	29
5	Descriptors	31
5.1	Conversion	31
6	Tree mode	33
6.1	Activation	33
6.2	Branch-switching	33
6.3	Anaphoric pronouns in joiner clauses	33
6.4	Errors	33
6.5	Example	34
7	Numerals	35
7.1	Single-digit numerals	35
7.2	Numerals up to $19 \cdot 17$	36
7.3	Numerals up to $13 \cdot 19 \cdot 17 = 4199$	36
7.4	Numerals up to and including $4199 \cdot (4199 + 1)/2 = 8817900$	37
7.5	Higher numerals	38
7.6	Cardinal and ordinal numerals	38
8	Derivational morphology	39
8.1	Abstraction	39
8.2	Dematuration	39
8.3	Verb-to-noun conversions	39
9	Names	41
9.1	Nominal names	41
9.2	Clausal names	42
10	Calendar	43
10.1	Tides	43
10.2	Months	45
10.3	Years	45
10.4	Eras	45
10.5	Subdivisions of the day	46
10.5.1	Traditional timekeeping	46
10.5.2	Modern timekeeping	46

0.1. INTRODUCTION	5
11 Miscellanea	49
11.1 Colour	49
A Listings of programs	51
A.1 workfiles/7/tides.sage	51
A.2 workfiles/7/bins.pl6	52
A.3 workfiles/7/conno.pl6	53
A.4 workfiles/7/count-days.pl6	54
B Arithmetic in base v	57
B.1 Operations on small numbers	57
B.1.1 Additions	57
B.1.2 Subtraction	57
B.1.3 Determining parity	57
B.1.4 Dividing by two	58
B.1.5 Multiplication	58
B.2 Operations on larger numbers	58
B.2.1 Addition	58
C Dictionary	61

0.1 | Introduction

1 | Phonology and orthography

1.1 | Phoneme inventory

Table 1.1: The consonants of aaaaaaaaaA.

	Bilabial	Alveolar	Palatal	Velar	Glottal
Nasal	m	n	ɲ	ŋ	
Plosive	p b	t d	c ɟ	k ɡ	ʔ
Fricative	f	s	ʃ	x	
(coarticulated)	θ x	ɸ x		ɸʃ	
Affricate		ts	tʃ		
Lateral fricative		ɬ			
Approximant		ɹ	j	w	
Lateral approximant		l			
Trill		r			

Table 1.2: The vowels of aaaaaaaaaA.

Spread	Half-rounded	Rounded
i	ɤ	y
u	ʊ	u
ɛ		œ
ʌ		ɔ
ä		

In addition to consonants and vowels, aaaaaaaaaA has rod signals, represented by numbers. Rod A is blue and held by one's dominant hand and B is red and held by one's non-dominant hand.

1. Rod A is raised to one's chest, while B is pointed down.
2. Rods A and B are crossed in the front.
3. Rod B is raised upwards in front of the nondominant arm, while rod A is lowered.

4. Rod A is pointed sideways near one's nondominant arm, while rod B is lowered.
5. Rods A and B are extended to the sides.
6. Rods A and B are extended, facing forward.
7. Rod A is raised forward, while B is pointed to the side.
8. Rod B is raised forward, while A is pointed to the side.

Lowering both rods is interpreted as an absence of a rod signal.

If the use of rods are unavailable, the numerals of the positions may be pronounced.

1.2 | Hacmisation

aaaaaaaaA uses the hacm script with superscript letters to indicate phonemes not found in Arka. The transcriptions can be found in Tables 1.3 and 1.4.

Table 1.3: The consonants of aaaaaaaaaA.

	Bilabial	Alveolar	Palatal	Velar	Glottal
Nasal	ɒ	n	n ^ɥ	n ^ɣ	
Plosive	d b	f ɳ	f ^ɥ ɳ ^ɥ	ɭ ɸ	.
Fricative	ɑ	ʃ	ʎ	h	
(coarticulated)	j ^h	ɑ ^h		ɑ ^ɭ	
Affricate		ʃʃ	ʎ		
Lateral fricative		s			
Approximant		ɹ	ɥ	o	
Lateral approximant		l			
Trill		r			

Table 1.4: The vowels of aaaaaaaaaA.

Spread	Half-rounded	Rounded
ɕ	ɕ ^ə	ɕ ^ɔ
ə ^ɕ	ə	ə ^ɔ
e		e ^ɔ
ɔ ^e		ɔ
ɪ		

Rod signs are represented by the hacm digits <1 ʃ ʈ ɳ ɥ ɳ Δ> attached to the end of the verbs they encompass. Proper words are preceded by a backslash <\>.

Vowels that are inferrable from context are sometimes omitted. For example, /æfan/ (to speak) is written <ɰean>, but /æfin/ (to spread), which is less common, is written <ɰeacn>, with the second vowel. Most of this grammar will leave all vowels written.

1.3 | Phonotactics

An onset consists of one of the following:

- any single consonant other than /l/ (the exceptions are <le> [lek] and related words),
- any obstruent followed by an approximant other than /l/,
- or any plosive followed by /r/,
- or any nasal followed by /j/ or /w/.

A nucleus consists of one vowel.

A coda consists of one of the following:

- nothing,
- a nasal,
- a voiceless plosive (excluding /ʔ/),
- /ɹ/, /s/ or /l/

1.4 | Stress

Stress falls on the last syllable with a coda, or otherwise the second-to-last syllable.

See table 1.5 for examples.

Table 1.5: Examples of stress locations.

Orthography	Location of stress (# from last)
ɒɪɹ	2
nɪ.cn	1
.əʔfəɔ	2
ɹɪɲedc ^a	3

1.5 | Vowel harmony

For the purposes of vowel harmony, vowels are divided into front and back vowels. /a/ is neutral. A root with neither front nor back vowels acts as if it has front vowels.

Table 1.6: Single consonants in the script.

Ʒ d	† f	↗ l	‡ j	⌘ a
⌘ n	⌚ ɒ	ss h	ʒ j ^h	⊙ u
△ r	? l/s	ℓ l	☆ o	⋈ φ
⌚ n ^φ	X ɔ	⊙ b	⌘ a ^l	⊗ a ^h
⌚ ɔ ^u	⌚ ɔ ^u	⊗ μ	⌘ n ^u	⊗ .

1.6 | The script of aaaaaaaaaA

aaaaaaaaaA also uses its own script, inspired by one of Uruwi's old childhood cyphers.

The consonants within a word are divided into pairs (plus one single consonant at the end if applicable). Thus, <ɔn^φɔɔ^e-μɔ> would have <ɔn^φ ɔμ>. These pairs then get a glyph that combines the glyphs for their constituent consonants.

The full table of consonant pairs can be found at tables 1.7 and 1.8. There are some general rules:

- Double consonants get their single-consonant glyphs with a ring below.
- d-coloured glyphs bear the characteristic middle bar of <Ʒ><d>: Ʒ d + ℓ l → ⌚ dl.
- l-coloured glyphs rest under the characteristic arrow of <↗><l>: ↗ l + † f → ↗ f.
- f-coloured glyphs rest under the characteristic hilt of <†><f>: † f + △ r → △ fr.
- j-coloured glyphs bear the characteristic bar-and-circle of <‡><j>: ‡ j + ⊙ u → ⊙ jb.
- a-coloured glyphs bear the characteristic double-swash of <⌘><a>: ⌘ a + ⌚ n^φ → ⌘ an^φ.
- ɒ-coloured glyphs bear the characteristic brook of <⌚><ɒ>: ⌚ ɒ + Ʒ d → ⌚ ɒd.
- j^h-coloured glyphs bear the characteristic arc of <ʒ><j^h>: ʒ j^h + ⌚ ɒ → ⌚ j^hɒ.
- u-coloured glyphs rest under the characteristic triangle of <⊙><u>: ⊙ u + ⋈ φ → ⊙ uφ.
- r-coloured glyphs rest under the characteristic overring of <△><r>: △ r + ⌘ a^l → △ ra^l.
- l-coloured glyphs rest to the left of the characteristic vertical line of <?l><l>: ?l l + ⌚ ɔ^u → ⌚ lɔ^u.

[illegible]

- μ -coloured glyphs rest to the left of the characteristic flare of $\langle \langle 3 \rangle \langle \mu \rangle$: $\langle 3 \mu \rangle + \mathbb{A} \ n^4 \rightarrow \mathbb{A}^3 \mu n^4$.
- n^4 -coloured glyphs bear the characteristic inner circle of $\langle \mathbb{A} \rangle \langle n^4 \rangle$: $\mathbb{A} \ n^4 + \mathbb{C} \ a^h \rightarrow \mathbb{C} \ n^4 a^h$.
- If all else has failed, the two consonants are superimposed. The default order is the same as the ordering used in table 1.6.
- In coloured-consonant pairs, the colourant is assumed to occur first unless the order is switched by an order reversal mark.
- A negative-sloping mark below a glyph means that the order of consonants is switched.

Thus in our case, we would have $\langle \text{𐌲} \text{𐌶} \rangle$. The next step is to add vowels. In our case, they would be paired as $\langle \text{𐌲} - \text{𐌶}^e - \text{𐌶} \rangle$. Note that it is possible for a pair to not have both vowels. The diacritics for the vowels are quite irregular, and they are shown in table 1.9.

Table 1.9: Vowel pairs in the script.

	∅	ı	c	c ^ə	c ^ɔ	e	e ^ɔ	ə ^c	ə	ə ^ɔ	ɔ ^e	ɔ
∅	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
ı	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
c	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
c ^ə	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
c ^ɔ	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
e	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
e ^ɔ	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
ə ^c	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
ə	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
ə ^ɔ	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
ɔ ^e	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲
ɔ	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲	𐌲

Thus, after adding vowels we get $\langle \text{𐌲} \text{𐌶} \rangle$.

Table 1.10: Miscellaneous symbols.

- 𐌲 1 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲 𐌲
- 𐌲 𐌲 period (circumfix)
- 𐌲 comma
- 𐌲 name mark (equiv. to $\langle \backslash \rangle$)

2 | Syntax

2.1 | Basic word order

The basic word order is VSO. Descriptors follow what they modify.

2.2 | Questions

Binary questions have the interrogative polarity marker and no change to syntax.

In wh-questions, the wh-word is pulled to the front (i. e. before the verb). This requires case marking for the wh-word:

flen peac^əjhi nc^ʔɕ
who-ACC speak-FAR.PAST-Q PR.FAR.SG
Whom did you speak to?

This applies only to questions, not interrogative-mood clauses that act as relative clauses:

peac^əjhi nc^ʔ flel, qif mu.
speak-FAR.PAST-Q PR.FAR.SG who, see-NEAR.PAST PR.ANAPH_OBJ.INT
I saw the person whom you talked to.

2.3 | Multiple clauses

A sentence might have multiple clauses. Each clause in a sentence follows the basic VSO order, and clauses are separated with commas.

3 | Nouns

Nouns are declined for number, case and definiteness.

3.1 | Number

aaaaaaaaA has many grammatical numbers:

Table 3.1: The discrete grammatical numbers of aaaaaaaaaA.

Number	Constraint on $x \in \mathbb{Z}$
Integral	none
Nullary	$x = 0$
Singular	$ x = 1$
Dual	$ x = 2$

Table 3.2: The continuous grammatical numbers of aaaaaaaaaA.

Number	Constraint on $x \in \mathbb{R}$
Nullary	$x = 0$
Subsingular	$ x < 1$
Supersingular	$1 \leq x < 2$
Plural	$ x \geq 2$ or x is unknown

3.2 | Case

In a clause with both the subject and object directly expressed in that order, both the subject and object are declined in the nominative case (and their roles are inferred through word order). In a clause where only one is present, or where both are expressed in the opposite order, the subject will receive the nominative case and the object will receive the accusative case.

3.3 | Noun classes

There are three overarching groups of noun classes.

3.3.1 | Countable

Nouns in these classes are declined for a discrete number.

1. Sentient – such as humans, AIs, deities.
2. Animate – nonsentient animals.
3. Inanimate – anything else.

3.3.2 | Measurable

Nouns in this class are declined for a continuous number.

4. Measure – all measurable nouns, especially units of measurement.

3.3.3 | Uncountable

Nouns in these classes are not declined for number, and require compounding with a countable or measurable noun in order to be quantified.

5. Fluid – liquids and gases.
6. Edible – edible (to humans) non-fluids.
7. Inedible – inedible (to humans) non-fluids.
8. Abstract – abstract ideas.

3.4 | Definiteness

The definite form of a noun is formed regularly by reduplicating the first syllable (without the coda): <𐤓𐤓𐤕> “a person” becomes <𐤓𐤓𐤕𐤓𐤕> “the person”.

3.5 | Declension table

3.5.1 | Countable classes

Note that noun declensions respect vowel harmony. For nouns with back vowels, replace the front vowels with the back vowels of the same height and rounding, and vice versa.

Table 3.3: Declensions for countable nouns.

	Integral	Nullary	Singular	Dual
--	----------	---------	----------	------

	Integral	Nullary	Singular	Dual
Sentient: <ɖɪɪ> “person”				
Nominative	ɖɪɪ	ɖɪɪle	ɖɪɪ	ɖɪɪl
Accusative	ɖɪɪn	ɖɪɪnʰle	ɖɪɪncɟ	ɖɪɪnɪl
Sentient: <ɟʰi.en> “magician”				
Nominative	ɟʰi.en	ɟʰi.efe	ɟʰi.eɟ	ɟʰi.el
Accusative	ɟʰi.eɸcn	ɟʰi.enʰle	ɟʰi.eɸcɟ	ɟʰi.eɸcl
(Note that the final consonant is preserved only in the integral nominative form.)				
Animate: <ɖənʰɔ> “rabbit”				
Nominative	ɖənʰɔ	ɖənʰɔɔʰe	ɖənʰɔɟ	ɖənʰɔ.əʰe
Accusative	ɖənʰɔn	ɖənʰɔnʰe	ɖənʰɔn	ɖənʰɔnəʰe
Animate: <ɟcʰɪ> “fox”				
Nominative	ɟcʰɪ	ɟcʰefe	ɟcʰɟ	ɟcʰe.c
Accusative	ɟcʰɸcn	ɟcʰɪnʰle	ɟcʰɸcɟ	ɟcʰɸc
Inanimate: <ɪɪɪə> “statue”				
Nominative	ɪɪɪə	ɪɪɪəɔʰe	ɪɪɪəɟ	ɪɪɪəɟ
Accusative	ɪɪɪəɖ	ɪɪɪəɖɔʰe	ɪɪɪəɖɔʰe	ɪɪɪəɖəʰe
Inanimate: <.cɖen> “house”				
Nominative	.cɖen	.cɖefe	.cɖel	.cɖeɟ
Accusative	.cɖeɸcɖ	.cɖeɸcɖe	.cɖeɸcɖe	.cɖeɸcac

3.5.2 | Measurable classes

Table 3.4: Declensions for measurable nouns.

	Plural	Nullary	Subingular	Supersingular
Measure: <ɸəɖəʰ> “day (continuous)”				
Nominative	ɸəɖəʰ	ɸəɖəʰɸəʰ	ɸəɖəʰɪ	ɸəɖəʰn
Accusative	ɸəɖəʰn	ɸəɖəʰɸəʰn	ɸəɖəʰnʰe	ɸəɖəʰnəʰn
Measure: <ɖel> “volume” (in expressions such as <ɖel-ɸəɟəʰ> “cupful”)				
Nominative	ɖel	ɖeɸcʰ	ɖeɸcɪ	ɖeɸcn
Accusative	ɖeɸcn	ɖeɸcʰn	ɖeɸcnʰe	ɖeɸcnɪn
(Note that the final consonant is preserved only in the plural nominative form.)				

3.5.3 | Uncountable classes

Notably, uncountable-class noun declensions do not respect vowel harmony.

Table 3.5: Declensions for measurable nouns.

	Mass
Fluid: <ɑʰəɸə> “water”	
Nominative	ɑʰəɸə
Accusative	ɑʰəɸən
Fluid: <neled> “nitrogen”	

In wh-questions, the wh-word assumes the second position and the other argument becomes the first.

If a clause has no explicit arguments, the first argument is understood to be the subject.

Table 3.7: Personal pronouns. <-n>, <-en> or <-ɔ̃n> is suffixed for the accusative case.

(continuous) (discrete)	Pl. / Sub. / Sup. Integral	Nullary Nullary	Singular	Dual
Near	fi	lefi	de	acɥc
Far	dɔ	ɔ̃dɔ	nə ^c	bɥi
Other	nc	lenc	sc ^{ɔ̃}	liɥc
Anaph. Sub.	ɥi	leɥi	.cd	n ^{ɔ̃} cɥc
Anaph. Obj.	ɥɔ	ɔ̃ɥɔ	.ə ^c ɔ̃	n ^{ɔ̃} ə ^c ɥɔ
Generic	.ə ^{ɔ̃}			

(For the observant readers: notice the similarity to Kavinan's system.)

3.6.1 | Last-clause pronouns

The anaphoric pronoun <bej> (accusative: <bejen>) is grammatically an other pronoun, and it refers to the previous clause said. Likewise, <bedcɥ> (accusative: <bedcn>) refers to the clause before the previous one.

3.7 | Compounding

Nouns can be compounded together in a head-initial manner. When that happens, only the leftmost noun is the one to be declined.

ɔ̃el-ɥəɔ̃-ə^ləɥə^{ɔ̃}-ɥcɥ
 volume-cup-water-five
 five cupfuls of water

Note that integral pronouns can modify other nouns, in which personal possession is indicated:

ɔ̃el-ɥəɔ̃-ə^ləɥə^{ɔ̃}-ɥcɥ-fi
 volume-cup-water-five-PR.NEAR.INTEGRAL
 (arg1)'s five cupfuls of water

Descriptors can also compound on nouns. This compounding is productive in aaaaaaaaA.

ɔ̃ɥi-lə^{ɔ̃}fi
 person-old

old people

(Compare to ႁႏ ႏႃႉ “person old-SENTIENT”.)

3.8 | Possession

“X’s Y” is translated as <Y=ႁႏ ႏႃႉ X>. The possessive construction is also used to create appositives.

Observe that possession marks the head, and <=ႁႏ> is a clitic, not an affix, as in the following example:

ႁႃႁႃႏႃႉ-ႃႉႃႃႃ-ႁႏ ႏႃႉ ႏႃႉ.ႃႉ

DEF~rabbit-SING-water=GEN POS magician-SING

the magician’s water rabbit

In more casual speech, <ႏႃႉ> may be dropped.

4 | Verbs

Verbs are conjugated for person of the subject, tense, polarity and tellicity, in two paradigms. Conjugation respects vowel harmony.

Table 4.1: Person-tense conjugations for verbs, using <ḍilɪn> “(S) eats (O)”.

	Nonpast	Past
Near	ḍilɪn	ḍilɪf
Far	ḍilɪn	ḍilc ^a ɟ
Other	ḍilɪ	ḍilc ^a
Anaph. Sub.	ḍile	ḍilel
Anaph. Obj.	ḍilc.e	ḍilc.el
Generic	ḍilc ^ɔ	ḍilc ^ɔ

Table 4.2: Person-tense conjugations for verbs, using <peacn> “(S) spreads (O)”.

	Nonpast	Past
Near	peacn	peacɪ
Far	peaɪn	peac ^a ɟ
Other	peaɪ	peac ^a
Anaph. Sub.	peae	peael
Anaph. Obj.	peac.e	peac.el
Generic	peac ^a	peac ^a

to which a suffix is added:

Table 4.3: Polarity-tellicity suffixes for verbs. The interrogative affix can also follow a negative affix.

	Positive	Negative	Interrogative
Telic	-	-le / -ɔ ^e	-hɪ
Atelic	-ɔc / -ɔc ^c	-ɟɪ	-lc ^a / -lə

Notes:

- “Negative atelic” means something akin to “unsuccessfully tried to avoid doing X”.
- The interrogative polarity, in addition to marking questions, is used to mark clauses that may or may not be true but are referred to later in the sentence.

Some examples:

ḏḏḏḏ ḏḏḏḏ ḏḏḏḏ.
eat-NEAR.NONPAST fish flower
Fish eat flowers.

ḏḏḏḏ ḏḏḏḏ ḏḏḏḏ, ḏḏḏḏ ḏḏḏḏ ḏḏḏḏ.
eat-NEAR.NONPAST fish flower, eat-NEAR.NONPAST cat PR.ANAPH_SUB
Fish eat flowers, and cats eat fish.

ḏḏḏḏ ḏḏḏḏ ḏḏḏḏ, ḏḏḏḏ ḏḏḏḏḏḏ.
eat-NEAR.NONPAST fish flower, eat-ANAPH_SUB.NONPAST grass-ACC
Fish eat flowers, and they eat grass.
(Grass is inedible to humans, but edible to fish.)

ḏḏḏḏḏḏ ḏḏḏḏ ḏḏḏḏ.
eat-NEAR.NONPAST-NEG flower fish
Flowers don’t eat fish.

ḏḏḏḏ ḏḏḏḏ ḏḏḏḏḏḏḏ, ḏḏḏḏ ḏḏḏḏ.
carry-OTHER.NONPAST PR.OTHER.SG DEF~book-SG, worry-NEAR.NONPAST PR.NEAR.INT
PR.LAST_CLAUSE
He has the book; that worries me.
or: That he has the book worries me.

ḏḏḏḏḏ ḏḏḏḏ ḏḏḏḏḏḏḏ, ḏḏḏḏ ḏḏḏḏ.
carry-OTHER.NONPAST-INTERROGATIVE PR.OTHER.SG DEF~book-SG, worry-NEAR.NONPAST
PR.NEAR.INT PR.LAST_CLAUSE
He might have the book; that worries me.
or: That he might have the book worries me.

4.1 | Aspect

Verbs can also be marked for aspect, either using a rod sign directly on the verb, or a particle with a rod sign, placed anywhere between the verb it modifies and the next verb.

Table 4.4: Aspect markers. Those with hyphens are attached to verb. Those without hyphens are placed as separate particles anywhere after the verb.

Aspect name	Marking	Meaning
Imperfect	–1	An action that is currently going on. Also used to distinguish static actions as opposed to dynamic (e. g. <i>wear</i> as opposed to <i>put on</i>).
Interrupted	1c11	An action that was interrupted.
Perfect	–1	An action that has already finished. Changes present tense to immediate past. Also used to distinguish dynamic actions as opposed to static (e. g. <i>put on</i> as opposed to <i>wear</i>).
Gnomic	–ʔ	A general truth or aphorism, or an action done habitually.
Gnomic dubitative	1c1ʔ	A general truth or aphorism that the speaker considers to be false.
Deontic necessity	–Ꭰ	An action that the speaker insists on happening.
Epistemic necessity	1əᵑᵑᎠ	An action that the speaker infers that is happening.
Deontic potential	–Ꭱ	An action that the speaker permits to occur.
Epistemic potential	1əᵑᵑᎡ	An action that the speaker infers that might happen.
Unexpected	–ɿ	An action that is unexpected (akin to using “but”).
Comparative	deɿ	Indicates an action of greater intensity than what was described in the previous clause.
Nonexclusive subject	1c1	Indicates that the subject comprises not only of what is explicitly mentioned, but also other things.
Nonexclusive object	1cʔ	Indicates that the object comprises not only of what is explicitly mentioned, but also other things.
Nonexclusive argument	1cᎠ	Combination of both nonexclusive subject and nonexclusive object.

An example:

11111c1 de nc, 1cnc.ɛ1ʃ dənʔidoᵑ–ɲ.
 fight-NEAR.PAST-ATELIC-IMPERFECT PR.NEAR.SG PR.OTHER.INT, shoot-ANAPH_OBJ.PAST-
 UNEXPECTED knee-SG.ACC-PR.ANAPH_SUB.INT
 I tried to fight them, but they shot my knee.

4.2 | Obliques

aaaaaaaaA lacks oblique arguments. Instead, equivalent expressions employ serial verb constructions. For instance, “he ate soup with a spoon” would be reduced to “he held a spoon and ate soup”:

dɥc^a sc^o aɔ^oɔ^ol, ɒlɛl sɪdʒɒn.
 INST-OTHER.PAST PR.OTHER.SG spoon-SG, eat-ANAPH_SUB.PAST soup-ACC
 He held a spoon and ate soup.
 or: He ate soup with a spoon.

Likewise:

nɪ.ɪ nc jəjəl-hɪ.ɪp, ncɒɛ hɪhɪɾə^odɔ^o.
 TEMPORAL-OTHER PR.OTHER.INT DEF~day-SG-spring, dance-ANAPH_SUB DEF~statue-SG.ACC
 They will wait until the spring equinox and dance around the statue.
 or: They will dance around the statue on the spring equinox.

A similar construction can be used for the negation of obliques:

dɥc^ale sc^o aɔ^oɔ^ol, ɒlɛlɪ sɪdʒɒn.
 INST-OTHER.PAST-NEG PR.OTHER.SG spoon-SG, eat-ANAPH_SUB.PAST-UNEXPECTED soup-ACC
 He did not hold a spoon, but ate soup.
 or: He ate soup without a spoon.

4.3 | Conjunctions

Conjunctions such as “and” are treated like obliques. For instance, “and” is represented by the verb <acɒ>, and precedes the clause in which the two are used:

ac^a ɥc^aje ɥɪpəl, ɒlɛl fɛp.c^on.
 and-OTHER.PAST Ryse Tarul, eat-ANAPH_SUB.PAST beef-ACC
 Ryse and Tarul ate beef.

Sufficiently complex nesting may be unrepresentable using only anaphoric referents. The easiest way to resolve this issue is to use definite nouns in place of anaphoric referents.

ac^a ɥc^aje ɥɪpəl, ac^a fɛp.c^o sɪdʒɒ, ɒɪlc^a ɒɪɪɪɪɪɪ ɥɪ.
 and-OTHER.PAST Ryuse Tarul, and-OTHER.PAST beef soup, eat-OTHER.PAST DEF~person-DU PR.ANAPH_SUB.CONT
 [They,] Ryse and Tarul ate beef and soup.

4.4 | Subordinate clauses

Ideas such as “if” or “because” are also expressed with verbs. For example, $\langle \text{ni.cn} \rangle$ “wait, when” is also used for “if”:

feɣilehi, ni.cn fi bej, æhcn.
 rain-OTHER-NEG-Q, wait-NEAR PR.NEAR.INT ANAPH_CLAUSE, play-NEAR
 If it doesn't rain, we will play.

Note the clausal argument to $\langle \text{ni.cn} \rangle$, since our condition is an entire clause instead of a noun.

4.4.1 | Conditions

Conditional ideas whose English translations contain “if” can also be expressed in a more concise way, but this usage can sometimes sound colloquial:

feɣilehi, æhcn.
 rain-OTHER-NEG-Q, play-NEAR
 If it doesn't rain, we will play.

4.5 | Comparatives

The comparative is a function $\text{cmp} : A \times A \times (A \rightarrow \mathbb{R}) \times (A \times A \rightarrow \{0, 1\}) \rightarrow \{0, 1\}$, where $\text{cmp}(a, b, f, \sqsupset) = f(a) \sqsupset f(b)$.

Consider the following sentences:

Fish eat flowers more than cats.
 More fish eat flowers than cats.

Semantically, they can be translated to:

$$\text{cmp}(\text{fish}, \text{cats}, a \mapsto (\# \text{ of flowers eaten by } a), >) \quad (4.1)$$

$$\text{cmp}(\text{fish}, \text{cats}, a \mapsto (\# \text{ of } a \text{ that eat flowers}), >) \quad (4.2)$$

The heart of comparatives in aaaaaaaaA is the quadrivalent verb $\langle \text{ɔʔɔɔɔ} \ a \ b \ f \ \sqsupset \rangle$. Thus:

ɔɔɔɔɔɔɔɔ ɔʔɔɔɔ-ɔʔɔɔɔ, ɔʔɔɔɔ ɔɔɔɔ ɔʔɔɔɔ ɔʔɔɔ ɔʔɔɔ.
 eat-GENERIC-Q flower-ACC.INT-how_many, CMP-NEAR fish cat PR.ANAPH_OBJ.INT >
 Fish eat more flowers than cats.

ɔɔɔɔɔɔɔ .əʔ-ɔʔɔɔɔ ɔʔɔɔɔ, ɔʔɔɔɔ ɔɔɔɔ ɔʔɔɔɔ ɔʔɔɔ ɔʔɔɔ.
 eat-GENERIC-Q PR.GENERIC-how_many flower, CMP-NEAR fish cat PR.ANAPH_SUB.INT
 >

More fish eat flowers than cats.

Note that we place a clause whose argument is the generic pronoun before the comparative clause. From the *doran*-clause, we refer to the function using the anaphoric pronoun referring to the position of the return value.

Table 4.5: Comparators in aaaaaaaaaA.

□	Comparator
>	nef
<	ac ^ə
=	fe ^ə n ^ə
≥	f ^ə il
≤	dcj
≠	.c ^ə
≈	pej
≫	a ^h e
≪	din

4.6 | Ditransitive-like constructions

In English, some verbs such as *give* take two objects: the item being given and the recipient of the item. Since clauses in aaaaaaaaaA can take only one object, translating such verbs requires multiple clauses:

fɪpɪf de hɪpɪpɪcɔen, nebel \pɪc^əjen.
 lose-NEAR.PAST PR.NEAR.SG DEF~book, give_to-ANAPH.SUB.PAST Ri^use-ACC
 I gave the book to Ryse.

4.7 | Transitivity

Verbs that are intransitively (i. e. have no object passed at this time) can be turned into a causative form with the prefix <φC->:

fɪlɪɪf aɪpɪpɪn^ue.
 fall-NEAR.PAST DEF~coin
 The coins fell.

de φɪlɪɪɪf aɪpɪpɪn^ue
 PR.NEAR.SG TRANS-fall-OTHER.PAST DEF~coin
 I dropped the coins.

Note that the word order changes to SVO. In addition, the verb is conjugated for its object, rather than the subject as expected. If the following clause uses an anaphoric

subject, it refers to the object of the current clause.

Moreover, the verb does not need to be one that can never take an object. In the above example, <fclrcn> means “(S) falls on (O)”. However, if the verb in question is taking an object, it cannot be transitivised directly and a more roundabout way is required:

fclrcf aqeaqen^{4e} pıfı.
fall-NEAR.PAST DEF~coin grass
The coins fell on the grass.

de fclrc^ə aqeaqen^{4e}, fclrel pıfıbe.
PR.NEAR.SG TRANS-fall-OTHER.PAST DEF~coin, fall-ANAPH.SUB.PAST grass-ACC
I dropped the coins; they fell on grass.
or: I dropped the coins on grass.

4.8 | Clauses with nullary arguments

A clause with one or more arguments that are nullary or modified by nullary-number nouns (either through compounding or possession) will have a negative verb as well:

dəfınle dıjıle.
recall-NEAR-NEG person=NULL
No one knows.

fjclfınle de jıfıdı jcl j^hı.ele.
want-NEAR-NEG PR.NEAR.SG ring=GEN POS magician=NULL
I don't want the rings of any magician.

4.9 | The copula

The copula <jcn> can take a noun as an object, in which case it can mean identity or membership. (Location is expressed with <lın> “be at”.) With no object at all, it is used to denote existence.

It can also accept a descriptor, in which case the descriptor is attached before <jcn> in the dictionary form.

5 | Descriptors

Descriptors act as adjectives or adverbs. They follow what they modify, and are inflected for the noun class or verbal person of their antecedents.

Table 5.1: Descriptor declensions, using the descriptors <hedfi> “large” and <leʔfi> “old”.

Class or person	Declined form	
Sentient	hedfi	leʔfi
Animate	hedfi	leʔfi
Inanimate	hedfe	leʔfɔ ^e
Measure	hedfiɲ	leʔfiɲ
Fluid	hedfej	leʔfɔ ^e j
Edible	hedfc	leʔfə ^c
Inedible	hedfeʔ	leʔfɔ
Abstract	hedfc ^a	leʔfə
Near	hedfiɥ	leʔfiɥ
Far	hedfiɥ	leʔfiɥ
Other	hedfeɥ	leʔfeɥ
Anaph. Sub.	hedfiɥ	leʔfiɥ
Anaph. Obj.	hedfeɥ	leʔfeɥ
Generic	hedfc ^ɔ ɥ	leʔfə ^ɔ ɥ

5.1 | Conversion

A noun can be converted to a descriptor by appending <-ɲ>.

A descriptor can be converted to an abstract noun meaning “the nature of being ~” by replacing the final <-i> with <-cneɪ>.

6 | Tree mode

As mentioned in section 4.3, anaphoric referents in a linked-list sentence are sometimes insufficient for expressing even simple sentence structures. While the easiest method of resolving this issue is using definite nouns, aaaaaaaaaA also provides a mode where sentences are not linked lists of clauses, but rather (binary) trees.

6.1 | Activation

Tree mode is enabled automatically when the treeing particle <ᵿᵿ> is used, and disabled at the end of a sentence.

6.2 | Branch-switching

The aforementioned particle <ᵿᵿ> marks the beginning of the right branch of the tree. The right branch is ended by the particle <ᵿᵿΔ>, which causes the next clause to join the left and right branches.

(N. B. <ᵿᵿ> and <ᵿᵿΔ> can occur only between clauses. If the particles are represented by left and right brackets, respectively, then the brackets should match.)

6.3 | Anaphoric pronouns in joiner clauses

In clauses that join two branches, anaphoric pronouns require marking whether the antecedent occurs in the left predecessor <ᵿᵿ> or the right predecessor <ᵿᵿΔ>. This is done by marking the pronoun with <-ᵿ> or <-Δ>.

Likewise, verbs can be modified with <-ᵿ> or <-Δ> to indicate which branch the subject came from.

6.4 | Errors

The following are ungrammatical:

- Using the particle <ᵿᵿΔ> or the branched anaphoric pronouns when tree mode is disabled

- Using the particle $\langle n^4\Delta \rangle$ other than to close a corresponding $\langle n^4\eta \rangle$
- Using the unbranched anaphoric pronouns in clauses with two predecessors
- Using the branched anaphoric pronouns in clauses with one predecessor
- Starting a new branch with $\langle n^4\eta \rangle$ when the current branch is empty

6.5 | Example

The second example in section 4.3 can be expressed as follows:

$ac^a \setminus \mu c^a \eta \setminus \eta \mu \theta$, $n^4\eta \ ac^a \ \eta \mu.c^a \ \text{sidjc}$, $n^4\Delta \ \text{d}\eta \text{el}\eta \ \mu c n \Delta$.

The resulting tree is shown below:



7 | Numerals

aaaaaaaaA uses a mixed-base system for its numerals. Numerals are abstract nouns. A Perl 6 program to convert numerals can be found in Section A.3.

7.1 | Single-digit numerals

Here are the numerals for $n < 17$:

Table 7.1: The cardinal numbers from 0 – 16.

base 10	base v	word
0	0	μəɒ
1	1	ac ^ə l
2	ɹ	ɹɹ
3	ʔ	ɒ ⁴ e ^ɔ n
4	ɒ	ɒ ⁴ ɹμ
5	ɹ	ɹcɹ
6	ɹ	ɒɹe
7	ɹ	ɹcɹ
8	Δ	də ^ɔ n
9	L	hed
10	F	bən ^ɸ
11	7	nə ^c
12	£	le ^ɔ
13	‡	j ^h cd
14	A	ɹɹn
15	V	ɹel
16	ψ	.ɹμ

Note that digits above 9 use capital hacm letters.

7.2 | Numerals up to $19 \cdot 17$

These are represented by two digits. The multiples of 17 are shown below:

Table 7.2: Multiples of 17, up to $18 \cdot 17$.

base 10	base v	word
17	10	selc ^a
34	J0	ɔilc ^a
51	ʔ0	ɔ ⁴ e ³ lc ^a
68	00	ʃɔlc ^a
85	ʔ0	ʔcʃc ^a
102	ʃ0	ɔ ⁴ elc ^a
119	90	ʃcʃc ^a
136	Δ0	də ³ nə
153	L0	helc ^a
170	F0	bən ⁴ ə
187	70	nə ^c lə
204	£0	le ³ lc ^a
221	‡0	j ^h cdlc ^a
238	A0	yinc ^a
255	V0	lelyc ^a
272	ψ0	.ilyc ^a
289	Ð0	jɪlsc ^a
306	ʰ0	heɔdc ^a

Thus $y \cdot 17 + x$ is written $\langle x-y \rangle$.

7.3 | Numerals up to $13 \cdot 19 \cdot 17 = 4199$

These are represented by three digits. The multiples of $19 \cdot 17$ are listed below:

Table 7.3: Multiples of $19 \cdot 17$, up to $12 \cdot 19 \cdot 17$.

base 10	base v	word
323	100	hɪjɪn
646	J00	ʃɪjɪn
969	ʔ00	ɔ ⁴ e ³ jɪn
1292	000	ɔ ⁴ ɪjɪn
1615	ʔ00	ʔcɪjɪn
1938	ʃ00	ɔ ⁴ eɪjɪn
2261	900	ʃcɪjɪn
2584	Δ00	də ³ nɪjɪn
2907	L00	heɪjɪn
3230	F00	bən ⁴ ɪjɪn

base 10	base v	word
3553	700	nə ^c jɪɪn
3876	£00	le ³ jɪɪn

Thus $(z \cdot 19 \cdot 17) + (y \cdot 17) + x$ is written $\langle z-x-y \rangle$.

7.4 | Numerals up to and including $4199 \cdot (4199 + 1)/2 = 8817900$

The numeral for 4199 is $\langle \text{ə.ɔlə}^c \rangle$, written as $\langle 1:000 \rangle$.

Likewise, two $\langle \text{ə.ɔlə}^c \rangle$ is written as $\langle \text{J}:000 \rangle$ and pronounced $\langle \text{ə.ɔlə}^c\text{-fɪj} \rangle$, but the second $\langle \text{ə.ɔlə}^c \rangle$ is one smaller than the first. In other words, $\langle \text{J}:000 \rangle = 4199 + (4199 - 1) = 8397$.

Table 7.4: “Multiples” of $\langle \text{ə.ɔlə}^c \rangle$.

“Multiple”	Difference from last	Total
(0)		0
1:000	4199	4199
J:000	4198	8397
ɿ:000	4197	12594
0:000	4196	16790
ɸ:000	4195	20985
...		
£.ɸɸ:000	3	8817897
£.ɸɸ:000	2	8817899
1::000:000	1	8817900

Thus the n th “multiple” differs from the $(n-1)$ th multiple by $(4199 + 1 - n)$ (given $1 \leq n \leq 4199$), and the sum of the first n “multiples” is

$$\begin{aligned}
 y(n) &= \sum_{i=1}^n (4200 - i) \\
 &= \frac{1}{2} \cdot (8399 \cdot n - n^2)
 \end{aligned} \tag{7.1}$$

And likewise, for some given y , the largest “multiple” of $\langle \text{ə.ɔlə}^c \rangle$ not smaller than y has the index

$$N(y) = \left\lceil \frac{1}{2} \cdot (8399 - \sqrt{70543201 - 8 \cdot y}) \right\rceil \tag{7.2}$$

In other words, for any numeral $\langle n_1:n_2 \rangle$, $n_1 + n_2$ must be less than 4199.

8 | Derivational morphology

The following methods are used to derive related terms from existing ones.

8.1 | Abstraction

Abstraction is a derivation that takes a non-abstract noun and returns the abstract noun representing the concept of the argument. This formation appends $\langle -ne \rangle$ or $\langle -n\omega^e \rangle$ to the noun.

Examples:

- $\langle h\mu c\alpha en \rangle$ *book* \rightarrow $\langle h\mu c\alpha ene \rangle$ *literature*
- $\langle h\omega l \rangle$ *cart* \rightarrow $\langle h\omega l n\omega^e \rangle$ *transportation*

Note that any double letters collapse into a single.

8.2 | Dematuration

Dematuration is a derivation that takes a noun and returns a noun of the same class that represents an immature form of the argument (not necessarily a diminutive). $\langle f\epsilon - \rangle$ or $\langle f\epsilon^c - \rangle$ are prepended to nouns that begin in $\langle j \rangle$ or $\langle l \rangle$, or $\langle j\epsilon - \rangle$ or $\langle j\epsilon^c - \rangle$ otherwise.

Examples:

- $\langle \mathcal{D}l j \rangle$ *person* \rightarrow $\langle j\epsilon \mathcal{D}l j \rangle$ *child*
- $\langle f\epsilon j\omega \mu \epsilon^3 \rangle$ *fruit* \rightarrow $\langle j\epsilon^c f\epsilon j\omega \mu \epsilon^3 \rangle$ *unripe fruit*
- $\langle j\epsilon r l \epsilon n \rangle$ *essay* \rightarrow $\langle f\epsilon j\epsilon r l \epsilon n \rangle$ *draft*

8.3 | Verb-to-noun conversions

Verb-to-noun conversions involve an operation called *inversion*; this operation swaps certain phonemes of a word:

- front vowels ↔ back vowels
- voiceless plosives ↔ voiced plosives (in any position other than in a coda)
- $f \leftrightarrow \beta$, $d \leftrightarrow f^h$ (in coda position)
- $a \leftrightarrow j^h$
- $j \leftrightarrow a^h$ (in any position other than in a coda)
- $h \leftrightarrow a^l$
- $\mu \leftrightarrow n$ (in any position other than after a fricative in an onset or in a coda)
- $s \leftrightarrow l$ (in any position other than in a coda)
- $\mu \leftrightarrow l$ (in coda position)
- $y \leftrightarrow o$

For instance, $\langle \mu i.c f i n \rangle$ would be inverted to $\langle n i . a ^ c n i n \rangle$.

All other phonemes are unchanged.

Since all of the conversions below are straightforward, only their names will be mentioned.

Table 8.1: Verb-to-noun conversions, from the inversion of the verb stem.

Name	Affix
Agent	$-e^o n^o$ / $-o n^o$
Patient	$-e^o d$ / $-o d$
Location	$-e^o \mu$ / $-o \mu$
Instrument	$-i f j$

Table 8.2: An example with $\langle n a ^ o b i n \rangle$ to *steal*.

Name	Derivation	Meaning
Agent	$n c ^ o d e ^ o n^o$	thief
Patient	$n c ^ o d e ^ o d$	stolen goods
Location	$n c ^ o d e ^ o \mu$	site of theft
Instrument	$n c ^ o d i f j$	tool used for theft

9 | Names

Names fall into two grammatical categories:

- *Nominal names* act as nouns. They are usually single words.
- *Clausal names* are entire clauses. These names usually refer to places, although a few people have clausal names. In extreme cases, such a name can span multiple clauses.

9.1 | Nominal names

These names act as nouns, and they are preceded by a backslash <\>. If the name spans multiple words (as common in foreign names), spaces are escaped by backslashes. No distinction is made between native and foreign names.

Only personal names can stand on their own, and even then, only given or full names. Other names must modify a common noun describing the nature of what is named, in the integral number without definiteness.

Table 9.1: Some examples of nominal names.

Name	Type
\pɕ ^ə je	Personal (native)
\ɸɪpəl	Personal (native)
\pɛɓcn	Personal (foreign)
dpe ^ɔ ɓe-\oɪɪ.c	Place (foreign)

Native names will usually respect vowel harmony. Children of parents who work in professions demanding physical labour (e. g. bricklaying) will usually have names with back vowels. In contrast, those born to parents of professions that do not demand physical strength (e. g. computer programming) will usually bear names with front vowels.

10 | Calendar

Domain II, which contains *Rymako*, has a day that is 26.99410 hours long. Other figures are given in terms of local days:

Table 10.1: Astronomical measures for Domain II.

Period	Length in local days
Local (synodic) day	1.00000
Sidereal day	0.99699
Tropical year (l_y)	301.94714
Sidereal year	302.03719
Synodic month (l_m)	30.80152
Sidereal month	27.95032

10.1 | Tides

In Domain II, the offset of the sea level due to the tide can be modeled by the following equations:

$$y = y_s + y_m \quad (10.1)$$

$$y_s = A_s \cdot (1 + A_{sa} \cdot \cos(\tau \cdot t)) \cdot \cos(2 \cdot \tau \cdot t) \quad (10.2)$$

$$y_m = A_m \cdot \left(1 + A_{ma} \cdot \cos\left(\frac{\tau \cdot t}{l_m}\right)\right) \cdot \cos\left(\frac{2 \cdot \tau \cdot (1 - l_m) \cdot t}{l_m}\right) \quad (10.3)$$

where:

$$\begin{aligned}
\tau &= 2 \cdot \pi \\
A_s &\approx 0.675 \\
A_{sa} &\approx 0.0532 \\
A_m &\approx 1.267 \\
A_{ma} &\approx 0.176 \\
y &= \text{offset of sea level in metres} \\
t &= \text{time since HAT in local synodic days}
\end{aligned}$$

An exact solution to $dy/dt = 0$ is not known to exist. However, the solutions to this equation can be found numerically. Consult Section A.1 for a Sage program to do so.

As the calendar used by aaaaaaaaA uses the high and low tides to count time, it is not synchronised even with days. The basic unit of time in the calendar is the *tidal day* $\langle \mu\theta\delta\theta^c \rangle$ (l_t) – the amount of time between a high tide and the second high tide thereafter, which is, on average, 1.03356 local synodic days, but can vary considerably. Thus:

$$l_m/l_t \approx 29.80148 \quad (10.4)$$

$$\approx 4053/136 \quad (10.5)$$

$$l_y/l_m \approx 9.80299 \quad (10.6)$$

$$\approx 7215/736 \quad (10.7)$$

This suggests that:

1. most months will have 30 days, but every 136 months, 27 months will have only 29.
2. most years will have 10 months, but every 736 years, 145 years will have only 9.

Figure 10.1: Table of year lengths in a cycle.

0123456789		0123456789		0123456789		0123456789
0 XXXXX9XXXX		19 XX9XXXX9XX		38 9XXXX9XXXX		57 XX9XXXX9X
1 9XXXX9XXXX		20 XXX9XXXX9X		39 9XXXX9XXXX		58 XXX9XXXX9X
2 9XXXX9XXXX		21 XXX9XXXX9X		40 9XXXX9XXXX		59 XXX9XXXX9X
3 9XXXX9XXXX		22 XXX9XXXX9X		41 X9XXXX9XXX		60 XXX9XXXX9
4 9XXXX9XXXX		23 XXX9XXXX9X		42 X9XXXX9XXX		61 XXX9XXXX9
5 9XXXX9XXXX		24 XXX9XXXX9X		43 X9XXXX9XXX		62 XXX9XXXX9
6 9XXXX9XXXX		25 XXX9XXXX9X		44 X9XXXX9XXX		63 XXX9XXXX9
7 X9XXXX9XXX		26 XXX9XXXX9		45 X9XXXX9XXX		64 XXX9XXXX9
8 X9XXXX9XXX		27 XXX9XXXX9		46 X9XXXX9XXX		65 XXX9XXXX9
9 X9XXXX9XXX		28 XXX9XXXX9		47 XX9XXXX9XX		66 XXX9XXXXX
10 X9XXXX9XXX		29 XXX9XXXX9		48 XX9XXXX9XX		67 9XXXX9XXXX
11 X9XXXX9XXX		30 XXX9XXXX9		49 XX9XXXX9XX		68 9XXXX9XXXX
12 X9XXXX9XXX		31 XXX9XXXX9		50 XX9XXXX9XX		69 9XXXX9XXXX
13 X9XXXX9XX		32 XXX9XXXX9		51 XX9XXXX9XX		70 9XXXX9XXXX
14 XX9XXXX9XX		33 XXX9XXXXX		52 XX9XXXX9XX		71 9XXXX9XXXX
15 XX9XXXX9XX		34 9XXXX9XXXX		53 XX9XXXX9X		72 9XXXX9XXXX
16 XX9XXXX9XX		35 9XXXX9XXXX		54 XXX9XXXX9X		73 9XXXX9
17 XX9XXXX9XX		36 9XXXX9XXXX		55 XXX9XXXX9X		
18 XX9XXXX9XX		37 9XXXX9XXXX		56 XXX9XXXX9X		

9: 9 months

X: 10 months

10.5 | Subdivisions of the day

Lek-Tsaro has two systems for subdividing the day.

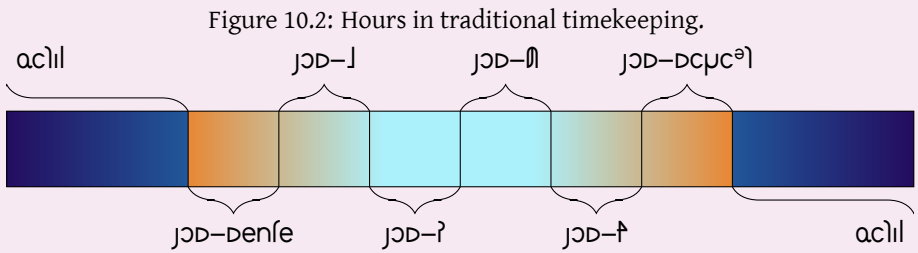
10.5.1 | Traditional timekeeping

The traditional system of timekeeping relies on subdivisions of the solar day. As shown in Figure 10.2, the period between sunrise and sunset are divided into six equally-sized parts <ᵛᵛᵛ>, and the night is considered a separate category. This implies that the length of the “hours” depends on the seasons.

The traditional timekeeping system is vague – typically, the most precise interval used is a quarter or eighth of an “hour”.

10.5.2 | Modern timekeeping

The need for precise schedules necessitated another standard for subdividing the day. The modern system is based on the tidal day, rather than the solar day. In theory, each



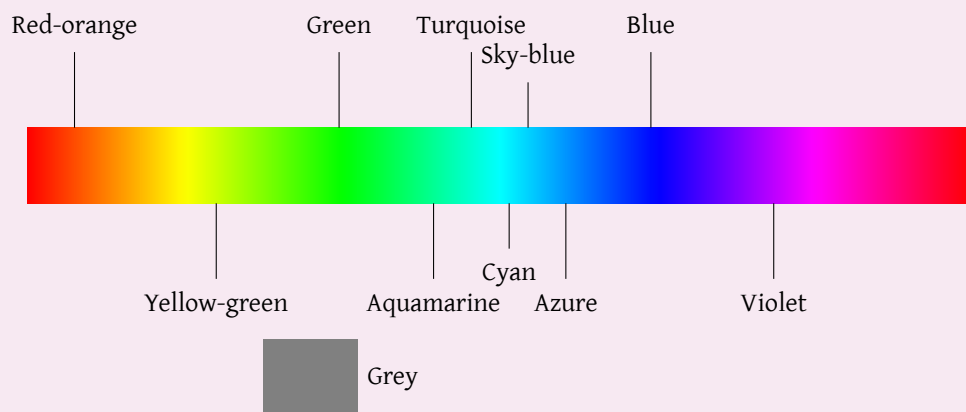
tidal day is divided into 23 equal parts $\langle \text{לח} \rangle$, each of which is divided into 80 equal parts $\langle \text{כנע} \rangle$, which are each divided into 40 equal parts $\langle \text{בדח} \rangle$.

Of course, having 23 $\langle \text{לח} \rangle$ per tidal day requires predicting the next two high tides. For that reason, each day's $\langle \text{לח} \rangle$ are based on the length of the *previous* tidal day, such that each day might have more or less than 23 $\langle \text{לח} \rangle$.

11 | Miscellanea

11.1 | Colour

aaaaaaaaA has eleven non-derived colour terms:



Note that “grey” refers generically to a loss of chroma. There is no distinction between a decrease in saturation and a decrease in value.

aaaaaaaaA works with colour *transitions*, not static colours, and uses abstract nouns to represent them. See table 11.1.

Table 11.1: Colour transitions in aaaaaaaaaA. Each row represents a different starting colour; each column represents a different ending colour.

	RO	YG	Gn	Aq	Tu	Cy	SB	Az	Bl	Vi	Gy
RO	hij	deI	n ⁴ en	ail	j ^h cμ	dəl	be ³ f	ɒil	ɑː ^h ed	fɛn ^ʰ	ɒc
YG	ɒə ^f	μə ³ ɒ	hɔ ^e d	ɑc ^ə I	n ^ʰ c ³ j	fəl	n ⁴ ip	ɑ ^h ef	fje ³ n ⁴	yed	də ³
Gn	f ⁴ ɔ ^e n	ɑː ^h f ⁴	seI	ɾəɒ	ʰə ³ f	nə ^c I	bɔn ⁴	le ³ f ⁴	jc ^ə I	j ^h ə ³ j	ɔ
Aq	j ^h ip	fəμ	μc ^ə ɒ	ocl	f ⁴ əl	.ə ^c I	fɔμ	ɔμɔj	noed	fɾid	be ³
Tu	ɑə ^c I	n ^ʰ ə ³ j	leI	n ⁴ c ^ə μ	ɒyɛn	μc ^j	hɔɒ	f ⁴ el	joc ^f	nə ³ μ	ɾə
Cy	bc ^ə μ	nɔ ^e f	nc ³ μ	.c ^f	ɾə ^c j	bɾel	fɔc ³ I	n ⁴ il	ɑμcn	bə ^c f	nɔ
SB	dɔI	n ⁴ il	de ³ n ⁴	fle ³ I	ɑː ^h e ³ ɒ	fə ³ μ	oid	j ^h μɔd	den	ɾcd	fjc ^ə
Az	ɒip	jɔ ^e I	ɔd	ʰɾe ³ j	nɔɔ ^e μ	n ⁴ il	ɑpe ³ f ⁴	ɑc ^j	fɔμ	μəd	oi
Bl	he ^f	fjɔn ⁴	ɑ ^h əμ	n ⁴ ɔ ^e f ⁴	ɑ ^h yə ^c I	j ^h μə ^c n	bɔ ^e n	nə ^c I	hil	ɔc ^ə n ^ʰ	yə ³
Vi	nɔ ^e n ^ʰ	ɔɔ ^e f ⁴	ɑc ³ j	n ⁴ il ⁴	fɔ ³ I	dc ³ I	μə ^c f ⁴	ɾc ^ə f ⁴	ʰən ^ʰ	hɔn	ai
Gy	nɔ	bc ^ə	le	də	μe	jc	fjɪ	yɪ	ɑ ^h e	j ^h c	.c ^j

A | Listings of programs

A.1 | workfiles/7/tides.sage

```
1 # How many values to output
2
3 limit = int(sys.argv[1]) if len(sys.argv) > 1 else 1000
4
5 # :P
6
7 tau = 2 * pi
8
9 t = var("t")
10
11 # Constants
12
13 A_s = 0.675; A_sa = 0.0532; A_m = 1.267; A_ma = 0.176; l_m = 30.80152
14
15 # Solar component
16 y_s2 = A_s * (1 + A_sa * cos(tau * t)) * cos(2 * tau * t)
17 # Lunar component
18 y_m2 = A_m * (1 + A_ma * cos(tau * t / l_m)) * cos(2 * tau * t / l_m -
19             2 * tau * t)
20 y = y_s2 + y_m2
21 yp = diff(y, t)
22
23 # High and low tides occur at values of t where dy/dt = 0.
24
25 i = 0
26 time = 0
27 print(0)
28 while i < limit:
29     try:
30         time2 = find_root(yp == 0, time + 0.000000001, time + 0.35)
31         print(time2)
32         time = time2
33         i += 1
34     except:
35         time += 0.01
```

workfiles/7/tides.sage

A.2 | workfiles/7/bins.pl6

```

1  # CONSTANTS
2
3  constant \MONTHS_PER_YEAR_CYCLE = 7215;
4  constant \YEARS_PER_YEAR_CYCLE = 736;
5  constant \AVG_MONTHS_PER_YEAR = MONTHS_PER_YEAR_CYCLE /
    YEARS_PER_YEAR_CYCLE;
6
7  # COMPUTATION
8  # For each year, take as many months as are needed
9  # in order to cycle to the next.
10
11 my $c = 0;
12 my @k;
13
14 for 0 ..^ YEARS_PER_YEAR_CYCLE -> $i {
15     my $need = 1 - ($c - floor($c));
16     my $objs = ceiling($need * AVG_MONTHS_PER_YEAR);
17     @k[$i] = $objs;
18     $c += $objs / AVG_MONTHS_PER_YEAR;
19 }
20
21 # DISPLAY
22
23 my \cols = 4;
24 my $len = @k.elems;
25
26 say ("    0123456789" xx cols).join(" | ");
27
28 my \total-rows = ceiling($len / 10);
29 my \rows = ceiling(total-rows / cols);
30
31 for 0 ..^ rows -> $j {
32     for 0 ..^ cols -> $p {
33         print(" | ") if $p != 0;
34         my $q = $j + rows * $p;
35         next if $q >= total-rows;
36         printf("%3d ", $q);
37         for 0 ..^ 10 {
38             my $i = 10 * $q + $_;
39             if $i >= $len { print " "; }
40             else {
41                 print "0123456789XE".substr(@k[$i], 1);
42             }
43         }
44     }
45     say "";
46 }

```

workfiles/7/bins.pl6

A.3 | workfiles/7/conno.pl6

```

1 my $digits-str = "0123456789TKXSNVFM";
2 my @digits = $digits-str.comb;
3
4 sub convert-small-fwd($n, $pad = False) {
5     die "$n must be < 4199" if $n >= 4199;
6     my $a = $n div (19 * 17);
7     my $b = ($n div 17) % 19;
8     my $c = $n % 17;
9     return
10         (!$pad && $a == 0 ?? "" !! @digits[$a]) ~
11         (!$pad && $b == 0 && $a == 0 ?? "" !! @digits[$b]) ~
12         @digits[$c];
13 }
14
15 sub convert-small-back($s) {
16     die "$s must be 3 chars or fewer" if $s.chars > 3;
17     my $c = $digits-str.index($s.substr(* - 1, 1) // "0");
18     my $b = $digits-str.index($s.substr(* - 2, 1) // "0");
19     my $a = $digits-str.index($s.substr(* - 3, 1) // "0");
20     return $c + 17 * ($b + 19 * $a);
21 }
22
23 sub triangle($n, $p) {
24     return ($n * (2 * $p + 1 - $n)) div 2;
25 }
26
27 sub sqrt-floor($y) {
28     die "$y is negative" if $y < 0;
29     return $y if $y < 2;
30     my $small = sqrt-floor($y +> 2) +< 1;
31     my $large = $small + 1;
32     return $small if $large * $large > $y;
33     return $large;
34 }
35
36 sub sqrt-ceil($y) {
37     my $n = sqrt-floor($y);
38     return $n if $n * $n == $y;
39     return $n + 1;
40 }
41
42 sub untriangle($y, $p) {
43     return (2 * $p + 1 - sqrt-ceil(4 * $p * $p + 4 * $p - 8 * $y + 1))
44         div 2;
45 }
46
47 my @powers = (4199);
48
49 for 0 .. 10 {
50     my $p = @powers[* - 1];
51     @powers.push: $p * ($p + 1) div 2;
52 }
53
54 sub convert-large-fwd-h($n, $i, $pad = False) {

```

```

53 # base case
54 if $i == 0 {
55     return convert-small-fwd($n, $pad);
56 }
57 # recursive
58 my $super = untriangle($n, @powers[$i - 1]);
59 my $infra = $n - triangle($super, @powers[$i - 1]);
60 if $super == 0 && !$pad {
61     return convert-large-fwd-h($infra, $i - 1, False);
62 }
63 return
64     convert-large-fwd-h($super, $i - 1, $pad) ~
65     (":" x $i) ~
66     convert-large-fwd-h($infra, $i - 1, True);
67 }
68
69 sub convert-large-fwd($n, $pad = False) {
70     my $i = 0;
71     ++$i while @powers[$i] <= $n;
72     convert-large-fwd-h($n, $i, $pad);
73 }
74
75 sub convert-large-back($s) {
76     # Find the longest run of colons
77     my @matches = ($s =~ m:g/"+"+); #/"
78     if (!@matches) {
79         return convert-small-back($s);
80     }
81     my $longest-match = @matches.max(*.chars);
82     my $i = (~$longest-match).chars;
83     my $left = $s.substr(0, $longest-match.from);
84     my $right = $s.substr($longest-match.to);
85     my $sup = convert-large-back($left);
86     my $inf = convert-large-back($right);
87     return triangle($sup, @powers[$i - 1]) + $inf;
88 }
89
90 multi MAIN(Int :$fwd) {
91     say convert-large-fwd($fwd);
92 }
93 multi MAIN(Str :$back) {
94     say convert-large-back($back);
95 }

```

workfiles/7/conno.pl6

A.4 | workfiles/7/count-days.pl6

```

1 # Count the number of days between 1/0/0 and D/M/Y, inclusive.
2
3 # CONSTANTS
4
5 constant \MONTHS_PER_YEAR_CYCLE = 7215;
6 constant \YEARS_PER_YEAR_CYCLE = 736;

```

```

7 constant \AVG_MONTHS_PER_YEAR = MONTHS_PER_YEAR_CYCLE /
  YEARS_PER_YEAR_CYCLE;
8 constant \MONTHS_PER_MONTH_CYCLE = 136;
9 constant \DAYS_PER_MONTH_CYCLE = 4053;
10
11 # COMPUTATION
12 # For each year, take as many months as are needed
13 # in order to cycle to the next.
14
15 my $c = 0;
16 my @k = (0);
17
18 for 0 ..^ YEARS_PER_YEAR_CYCLE -> $i {
19   my $need = 1 - ($c - floor($c));
20   my $objs = ceiling($need * AVG_MONTHS_PER_YEAR);
21   @k[$i + 1] = $objs;
22   $c += $objs / AVG_MONTHS_PER_YEAR;
23 }
24
25 my @cumk = [\+] @k;
26
27 sub months-before-year($year) {
28   my $whole-cycles = $year div YEARS_PER_YEAR_CYCLE;
29   my $remainder = $year % YEARS_PER_YEAR_CYCLE;
30   return $whole-cycles * MONTHS_PER_YEAR_CYCLE + @cumk[$remainder];
31 }
32
33 my @m = (0);
34
35 for 0 ..^ MONTHS_PER_MONTH_CYCLE -> $i {
36   @m.push: ($i % 5 == 2) ?? 29 !! 30;
37 }
38
39 my @cumm = [\+] @m;
40
41 sub days-before-month($month) {
42   my $whole-cycles = $month div MONTHS_PER_MONTH_CYCLE;
43   my $remainder = $month % MONTHS_PER_MONTH_CYCLE;
44   return $whole-cycles * DAYS_PER_MONTH_CYCLE + @cumm[$remainder];
45 }
46
47 sub days-before-date($d2, $m, $y) {
48   my $d = $d2 - 1; # d is 0-indexed
49   my $bm = months-before-year($y) + $m;
50   return days-before-month($bm) + $d;
51 }
52
53 sub MAIN($d2, $m, $y) {
54   say days-before-date($d2, $m, $y);
55 }

```

workfiles/7/count-days.pl6

B | Arithmetic in base v

This chapter describes algorithms for performing arithmetic operations in Lek-Tsaro's number system.

B.1 | Operations on small numbers

B.1.1 | Additions

If both addends are smaller than 4199, then it is sufficient to use mixed-base addition:

$$\begin{array}{r}
 1 \\
 0 F L \\
 9 ? A \\
 \hline
 F A ? \\
 \\
 1 1 \\
 J ? \uparrow \\
 \mathfrak{L} \psi ? \\
 \hline
 1 J 0 \Delta
 \end{array}$$

B.1.2 | Subtraction

If both of the operands are smaller than 4199, then it is sufficient to use mixed-base subtraction.

$$\begin{array}{r}
 ? 1? \\
 \mathfrak{J} ? A \\
 0 F L \\
 \hline
 J \mathfrak{L} \uparrow
 \end{array}$$

B.1.3 | Determining parity

A number less than 4199 is even iff the sum of its digits in base v is even – that is, either none of its digits are odd, or if exactly two are.

B.1.4 | Dividing by two

If a number's base- v representation contains only even digits, then divide each digit by two.

If the representation has two odd digits, then take advantage of the identities

$$11_v/2 = 9_v$$

$$101_v/2 = 99_v$$

$$110_v/2 = T0_v$$

This operation is written as $\langle D \rangle$, short for $\langle D \psi ne \rangle$ “one half”. Thus, in hacm:

- $D11 = L$
- $D101 = LL$
- $D110 = F0$

B.1.5 | Multiplication

With the previous two operations, it is now possible to use peasant multiplication to multiply small numbers.

B.2 | Operations on larger numbers

B.2.1 | Addition

For some $i \in \mathbb{N}$, and two numbers number $a = x_a :^i y_a$ and $b = x_b :^i y_b$, we take advantage of the fact that

$$x_a :^i y_a + x_b :^i y_b = (x_a + 1) :^i y_a + (x_b - 1) :^i y_b + (x_a - x_b + 1) \quad (B.1)$$

$$x_a :^i y_a + x_b :^i y_b = (x_a + x_b) :^i y_a + 0 :^i y_b + x_a \cdot x_b \quad (B.2)$$

$$= (x_a + x_b) :^i (y_a + y_b) + x_a \cdot x_b \quad (B.3)$$

Romanisation

In this text, the romanisation is used only to transcribe names into English. Whenever possible, the hacmisation should be used.

Table B.1: The consonants of aaaaaaaaA.

	Bilabial	Alveolar	Palatal	Velar	Glottal
Nasal	m	n	ɲ	ŋ	
Plosive	p b	t d	tʃ dʃ	k g	ʔ
Fricative	f	s	ʃ	h	
(coarticulated)	ɸh	fh		fʃ	
Affricate		ts	tʃ		
Lateral fricative		ɬ			
Approximant		r	j	w	
Lateral approximant		l			
Trill		ʀ			

Table B.2: The vowels of aaaaaaaaA.

Spread	Half-rounded	Rounded
i	y	ɥ
ĩ	u	ũ
e		ö
ë		o
a		

Rod signs are represented by the Arabic digits <1 2 3 4 5 6 7 8> attached to the end of the verbs they encompass. Proper words are preceded by a backslash <\>.

<ɲ> should be capitalised as <N> only if one can depend on the majuscule glyph appearing like an N with a hook. Otherwise, it should be spelled <Ng>.

C | Dictionary

<p> <i>.cɒn ninanim</i> house <i>.əʔfəɔ nabst</i> sadness, grief </p>	<p> <i>fəpɪn v</i> rain (S = other) <i>fəʈɪ nsent</i> coward, knave <i>fəʔpɔ nined</i> blood vessels </p>
<p> <i>ḥ</i> </p> <p> <i>ḥɪpɔ ninanim</i> river <i>ḥɪɪɪn v</i> (S) fights (O) <i>ḥɪɪɪn v</i> (S) falls on (O) <i>ḥɪpɔ ninanim</i> flower <i>ḥɪʔdɪ desc</i> sufficient, wanted, wished-for <i>ḥɪʔdɪ desc</i> complete, full, mature <i>ḥɪʔdɪ nsent</i> who? <i>ḥɪpɔ nabst</i> power, magic, motivation <i>ḥɪɪɪn v</i> (S) stabs, stings (O) <i>ḥɪɪɪn v</i> (S) wants (O), benefactive <i>ḥɪpɔʔ ninanim</i> fruit <i>ḥɪɪɪn v</i> (S) answers to (O) <i>ḥɪpɔ nsent</i> child (young person) <i>ḥɪpɪn v</i> (S) loses, frees (O); (O) escapes <i>ḥɪpɪ desc</i> heavy <i>ḥɪpɔ ninanim</i> tree <i>ḥɪpɔ nined</i> wood <i>ḥɪɪɪn v</i> (S) buys (O) <i>ḥɪpɔʔ nedib</i> beef </p>	<p> <i>ḥɪpɪʔ nsent</i> warrior <i>ḥɪpɪʔ desc</i> potent, powerful not in a physical sense <i>ḥɪpɪ nabst</i> south <i>ḥɪɪɪn v</i> (S) makes a loud noise <i>ḥɪpɪʔ ninanim</i> mirror <i>ḥɪɪɪn v</i> (S) is at (O), locational verb <i>ḥɪpɪ nmeas</i> subdivision of the day cf Grammar / Calendar / Subdivisions of the day / Modern timekeeping <i>ḥɪpɪʔ ninanim</i> moon <i>ḥɪpɪ ninanim</i> era <i>ḥɪpɪ nined</i> stone <i>ḥɪpɪ desc</i> all, every <i>ḥɪpɪ desc</i> whole, entire <i>ḥɪʔ.ɪn v</i> (S) needs (O) <i>ḥɪʔɪ desc</i> old </p>
<p> <i>ḥ</i> </p> <p> <i>ḥɪpɪʔncn v</i> (S) hunts for (O) <i>ḥɪpɪn v</i> (S) shoots an arrow to (O) <i>ḥɪpɪ nabst</i> nature, disposition </p>	

J

jɿɿɿ *ninanim* ring
 jcn v (S) is (O)
 jcncn v (S) attaches to, loves (O)
 jcne *nmeas* subdivision of the day
 cf Grammar / Calendar / Subdivisions of
 the day / Modern timekeeping
 jc^əl *nanim* fox
 jɔɔ *nmeas* subdivision of the day
 cf Grammar / Calendar / Subdivisions of
 the day / Traditional timekeeping
 jɔ^əɿɿ *ninanim* table
 je.in v (S) knows (O) answers (last
 clause)
 jenin v (S) is worried by (O)
 jea^lc *nabst* daytime
 jerɿɿcn *ninanim* essay
 je^əɿle *ninanim* land, country
 jəɿ *desc* many, again
 jəɿ *ninanim* day, sun

J^h

j^hɿ.en *nsent* magician
 j^hin v (S) goes toward (O)
 j^hɿɿcn v (S) creates (O)
 j^hcn *nabst* how many?
 j^he^əncn v (S) befriends (O)
 j^hən *nabst* how much?

n

nɿɿɿ *nanim* cat
 ni.cn v (S) waits for/until (O), tem-
 poral verb, if
 ni.eɿcn v (S) covers, spans (O)
 ncɔcn v (S) dances around (O)
 nchel *ninanim* group, organisation,
 order
 nc^ə *ninanim* point nc^ə-ɔɿine
 halfway point
 nelcn v (S) swims in (O)
 neɔ *desc* male

nebin v (S) gives something to (O)
 nepacn v (S) hides from (O)
 nepɿɿ *desc* sudden
 nel *nabst* nature, temperament,
 disposition
 neldɿ *nsent* mind, brain
 nə^ənɿn v (S) kills (O), (O) dies
 nə^əbin v (S) steals from (O)
 nə^əbɿɔ^ən *nsent* thief

n^ə

n^əɔɿɿn v (S) thinks, ponders about (O)

u

uɔ^əɔ *nfluid* poison

a

aɿen^{4e} *ninanim* coin
 acjcn v (S) obeys (O)
 acn v (S) joins (O), and
 acnɿ *desc* early
 acɿcn v touch
 acɿe *nabst* what
 ac^əl *desc* female
 aɔ^əɿɔ^ə *ninanim* spoon
 aehcn v (S) plays with (O)
 aə^ə *ninanim* event, occurrence

a^l

a^lɿe *nabst* quote, words, speech
 a^le *ninanim* what
 a^ləɿə^ə *nfluid* water

a^h

a^hə^əɔɔ *nabst* empathy

D

- ԾԿԻՆ *nabst* one half
 ԾԼԻՆ *v* (S) eats (O)
 ԾԼԻ *nsent* person
 ԾԼՆ *nedib* rice
 ԾՇԱ *nsent* child (offspring)
 ԾՇԸԵՅ *nanim* tongue
 ԾՇԸԵՂ *nabst* evening
 ԾՇԼԻՆ *v* (S) produces (O)
 ԾՇԼԻՆ *v* (S) is destroyed to make, for
 (O)
 ԾՇԸԸԾ *nanim* scorpion
 ԾԵԼՇՆ *v* (S) gives birth to (O), (O) is
 born (S) is not necessarily the mother;
 this can be either parent
 ԾԵԼՇՆ *v* (S) raises, takes care of,
 tends to (O)
 ԾԵՆԲ *nabst* morning
 ԾԵՊԻՆ *v* (S) stands on, is on (O)
 ԾԵՓԻՆ *v* (S) drowns in (O), (O) fills
 (S)
 ԾԵԼԻ *desc* in return
 ԾԵԾՇՆ *v* (S) succeeds at (O), (S)
 does something to (O)
 ԾԵԸԵՆ *nanim* opposite side
 ԾԵԼ *nmeas* volume in expressions
 such as ԾԵԼ-ԿՅՉԵՅ “cupful”
 ԾԵԼԻ *desc* similar
 ԾԵԼԼՇՆ *v* (S) imitates (O)
 ԾԵԼԻՆ *v* (S) recalls (O)
 ԾԵՆՓՇ *nanim* rabbit

Ռ

- ՌԻՆԼԻՆ *v* (S) is inside (O)
 ՌՈՇ *ninanim* back (body part)
 ՌՈԲ *nabst* life, existence
 ՌԵԼԵԾ *nfluid* nitrogen
 ՌԵԾԻՆ *v* (S) sleeps
 ՌԵՅՇ *nanim* pathway

Փ

- ՓԿԵՅՆԻՆ *v* (S) laughs at (O)

- ՓԵԵԼԻՆ *v* (S) founds (O)
 ՓԵԼԼԻ *desc* well (not sick)
 ՓԵԼՇ *nabst* five

d

- ԺԼԻՇԻ *nabst* ground, floor
 ԺԼԵՆ *nanim* owl
 ԺՄՇՆ *v* hold, carry, instrumental
 verb
 ԺՄԵՅԵ *ninanim* city
 ԺԼՄԻՆ *v* (S) sits at (O)
 ԺԼԻՆ *v* (S) dislikes, objects to, disap-
 proves of (O)
 ԺԼԵ *ninanim* landmass, domain
 ԺԵՄԻՆ *v* (S) wears, experiences (O)
 ԺԵՄԵՅ *nedib* noodles
 ԺԵ.ՇՆ *nanim* large animal
 ԺԵՆՓԻ *ninanim* knee

b

- ԲԻՆ *ninanim* year
 ԲԻԺԵԼ *nmeas* subdivision of the day
 cf Grammar / Calendar / Subdivisions of
 the day / Modern timekeeping
 ԲՇԼՇՆ *v* (S) walks to (O)
 ԲԵՇ *nabst* future, next (time period)
 ԲԵՅՆՓԻՆ *v* (S) succumbs to their im-
 pulses
 ԲԵՅԾՇ *nsent* adult person

h

- ԻՄՇՈԵՆ *ninanim* book
 ԻԼ.ԼՄ *nabst* spring (season)
 ԻԼՇՆ *ninanim* nose
 ԻԼԺԵ *ninanim* leaf
 ԻԼԺԵՅ *ninanim* statue
 ԻՇՆ *v* (S) claims that (O)
 ԻՇՄՇՆ *v* (S) is named (O)
 ԻՇՅԻՆ *nedib* food
 ԻՇԼԻ *ninanim* cart
 ԻԵԱՆ *v* (S) asks for, requests (O)

hedʃɪ desc large
 həʔɪ desc evil, malicious

q

qɪn v (S) sees (O), because
 qɪn[ɔ=jəp] (“see the sun”) = “wish”

r

rɪjɪn v discipline, punish, constrain
 rɪcn v (S) allows (O)

p

pɪ.cɪn v (S) is beside (O)
 pɪfɪ nined grass
 pɪfɪbɪ ninanim blade of grass
 pɪlɪn v (S) climbs, rises in (O)
 pɪcɪcn v (S) is (O) old
 pɪc^əɪ ninanim star
 pɪcɪn v (S) speaks to (O), (S) asks
 (O)
 pɪcɪcn v (S) spreads (O)

pɛɪɪ ninanim place
 pɛ^əɪɪ desc friendly, kind, consider-
 ate, nice
 pɛɪɪ desc late
 pɛɪ^ə nabst nighttime
 pɛɪ^ə ninanim cup
 pɛɪ^ə nmeas tidal day
 pɛ^əɪɪ nined gold

s

sɪdɪc nfluid soup
 sɪɪɪ nanim fish
 sɛn^ɪɪn v (S) perceives (O) non-
 visually

o

oɛɪɪ nined forest

l

lɛɪ nabst language
 lɛɛɛɪ ninanim a language