# lel–\ʃɹıɲɔ–ɒı ɟcl lel–ɒı ɟcl ɟeᵓʃle–\ɲcᵊɒıɔ

*aaaaaaaaaaaA*, the language of *Rymako*

uruwi

ıııııııııııııı

aaaaaaaaaaaaaaaaa

*A complete grammar*

October 20, 2017

*Dedicated to Isoraḳatheð.*

```
Branch: canon
Version: 0.1
Date: 2017-09-12
```

# Contents

# 0.1 | Introduction

# 1 | Phonology and orthography

## 1.1 | Phoneme inventory

Table 1.1: The consonants of aaaaaaaaaaA.

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

Table 1.2: The vowels of aaaaaaaaaaA.

| Spread | Half-rounded | Rounded |
|---|---|---|
| i | y̜ | y |
| ɯ | u̜ | u |
| ɛ |  | œ |
| ʌ |  | ɔ |
| ä |  |  |

In addition to consonants and vowels, aaaaaaaaaaA 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

aaaaaaaaaaA 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 aaaaaaaaaaA.

|  | Bilabial | Alveolar | Palatal | Velar | Glottal |
|---|---|---|---|---|---|
| Nasal | ᴅ | n | n� | nᵠ |  |
| Plosive | d b | ſ ῀ | ſᵁ ῀ᵁ | ꞁ φ | . |
| Fricative | ɑ | ɹ | ʅ | h |  |
| (coarticulated) | ɟʰ | ɑʰ |  | ɑˡ |  |
| Affricate |  | ʃ | ɭ |  |  |
| Lateral fricative |  | s |  |  |  |
| Approximant |  | ʯ | ɥ | o |  |
| Lateral approximant |  | l |  |  |  |
| Trill |  | ᴅ |  |  |  |

Table 1.4: The vowels of aaaaaaaaaaA.

| Spread | Half-rounded | Rounded |
|---|---|---|
| c | cᵊ | cᵓ |
| əᶜ | ə | əᵓ |
| e |  | eᵓ |
| ɔᵉ |  | ɔ |
| ı |  |  |

Rod signs are represented by the hacm digits ⟨꜀ ꜀ ? � ꜒ ꜏ ꜒ 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, /ɹɛ-fan/ (to speak) is written ⟨ɥeɑn⟩, but /ɹɛfin/ (to spread), which is less common, is written ⟨ɥeɑcn⟩, 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 ⟨lel⟩ [lɛk] 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) |
|---|---|
| ᴅıɟı | 2 |
| nı.cn | 1 |
| .əˤʃəlɔ | 2 |
| lıɟnedcə | 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.

# 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:

ʃen ɥeɑcᵊɟhı ncᵓჽ
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:

ɥeɑcᵊɟhı ncᵓ ʃel, ɥıʃ ɥɔ.
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

aaaaaaaaaaA has many grammatical numbers:

Table 3.1: The discrete grammatical numbers of aaaaaaaaaaA.

| 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 aaaaaaaaaaA.

| 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): ⟨ᴅɪjɪj⟩ "a person" becomes ⟨ᴅɪᴅɪjɪj⟩ "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: ⟨ᴅɪɟɪ⟩ "person" | | | | |
| Nominative | ᴅɪɟɪ | ᴅɪɟɪ̀ɭe | ᴅɪɟɪɟ | ᴅɪɟɪɭ |
| Accusative | ᴅɪɟɪn | ᴅɪɟɪnᵠɭe | ᴅɪɟɪnɕɟ | ᴅɪɟɪnɪɭ |
| Sentient: ⟨ɟʰɪ.en⟩ "magician" | | | | |
| Nominative | ɟʰɪ.en | ɟʰɪ.eɭe | ɟʰɪ.eɟ | ɟʰɪ.el |
| Accusative | ɟʰɪ.eɰɕn | ɟʰɪ.enᵠɭe | ɟʰɪ.eɰɕɟ | ɟʰɪ.eɰɕl |
| (Note that the final consonant is preserved only in the integral nominative form.) | | | | |
| Animate: ⟨ᴅənᵠɔ⟩ "rabbit" | | | | |
| Nominative | ᴅənᵠɔ | ᴅənᵠɔɭɔᵉ | ᴅənᵠɔɟ | ᴅənᵠɔ.əᶜ |
| Accusative | ᴅənᵠɔn | ᴅənᵠɔnɕᵉ | ᴅənᵠɔn | ᴅənᵠɔnəᶜ |
| Animate: ⟨ɟcᵊl⟩ "fox" | | | | |
| Nominative | ɟcᵊl | ɟcᵊɭe | ɟcᵊɟ | ɟcᵊ.c |
| Accusative | ɟcᵊɰɕn | ɟcᵊnᵠɭe | ɟcᵊɰɕɟ | ɟcᵊɰɕc |
| Inanimate: ⟨hɪɭəˀ⟩ "statue" | | | | |
| Nominative | hɪɭəˀ | hɪɭəˀɭɔᵉ | hɪɭəˀl | hɪɭəˀɟ |
| Accusative | hɪɭəˀᴅ | hɪɭəˀᴅdɔᵉ | hɪɭəˀdɔᵉ | hɪɭəˀɑəᶜ |
| Inanimate: ⟨.cᴅen⟩ "house" | | | | |
| Nominative | .cᴅen | .cᴅeɭe | .cᴅel | .cᴅeɟ |
| Accusative | .cᴅeɰɕᴅ | .cᴅeɰɕᴅde | .cᴅeɰɕde | .cᴅeɰɕɑc |

## 3.5.2 | Measurable classes

Table 3.4: Declensions for measurable nouns.

|  | Plural | Nullary | Subingular | Supersingular |
|---|---|---|---|---|
| Measure: ⟨ɰəᴅəᶜ⟩ "day (continuous)" | | | | |
| Nominative | ɰəᴅəᶜ | ɰəᴅəᶜɰəˀ | ɰəᴅəᶜɭ | ɰəᴅəᶜn |
| Accusative | ɰəᴅəᶜn | ɰəᴅəᶜɰəˀn | ɰəᴅəᶜnɕɔᵉ | ɰəᴅəᶜnəᶜn |
| Measure: ⟨ᴅel⟩ "volume" (in expressions such as ⟨ᴅel–ɥəɟɔᵉ⟩ "cupful") | | | | |
| Nominative | ᴅel | ᴅeɰɕɔ | ᴅeɰɕɭ | ᴅeɰɕn |
| Accusative | ᴅeɰɕn | ᴅeɰɕɔn | ᴅeɰɕnɭe | ᴅeɰɕncn |
| (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: ⟨ɑˈəɰəˀ⟩ "water" | |
| Nominative | ɑᵊɰəˀ |
| Accusative | ɑᵊɰəˀn |
| Fluid: ⟨ɷeɭeᴅ⟩ "nitrogen" | |

|             | Mass |
|-------------|------|
| Nominative  | ɲeˈleɒ |
| Accusative  | ɲeˈleɒcn |
| (Here, the coda is preserved in the accusative as well.) | |
| Edible: ⟨ʃeɰ.cˀ⟩ "beef" | |
| Nominative  | ʃeɰ.cˀ |
| Accusative  | ʃeɰ.cˀn |
| Edible: ⟨ɒɪn⟩ "rice" | |
| Nominative  | ɒɪn |
| Accusative  | ɒɪncn |
| Inedible: ⟨ɰəˀʃɔ⟩ "gold" | |
| Nominative  | ɰəˀʃɔ |
| Accusative  | ɰəˀʃɔbe |
| Inedible: ⟨ʅɪɖɪʝ⟩ "stone" | |
| Nominative  | ʅɪɖɪʝ |
| Accusative  | ʅɪɖɪʝde |
| Abstract: ⟨ɑʰəˀɒɔ⟩ "empathy" | |
| Nominative  | ɑʰəˀɒɔ |
| Accusative  | ɑʰəˀɒɔnᵠ |
| Abstract: ⟨φcʝ⟩ "[the number] five" | |
| Nominative  | φcʝ |
| Accusative  | φcʝcnᵠ |

## 3.6 | Pronouns

Personal pronouns are not divided into first, second and third persons as in most languages. Instead, they fall into four categories which exhibit different behaviour depending on whether they occur as the first or second noun in the clause:

Table 3.6: Pronoun persons and their functions.

| Person | Role in first position | Role in second position |
|--------|------------------------|-------------------------|
| Near | The speaker. | The first argument of the sentence. |
| Far | The listener. | If the first argument is the speaker, then the listener. Otherwise, the speaker. |
| Other | A third entity. | An entity that is neither the speaker, the listener nor the first argument. |
| Generic | A generic entity (akin to "one"). | |
| Anaphoric Subject | The subject of the previous clause. | |
| Anaphoric Object | The object of the previous clause. | |

In wh-questions, the wh-word assumes the second position and the other argu-

ment 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 ⟨–ɔᵉn⟩ is suffixed for the accusative case.

| (continuous) (discrete) | Pl. / Sub. / Sup. Integral | Nullary Nullary | Singular | Dual |
|---|---|---|---|---|
| Near | ʃi | ʔeʃi | ᴅe | ɑcµc |
| Far | dɔ | ʔɔᵉdɔ | nəᶜ | bµı |
| Other | nc | ʔenc | scɔ | ʔıµc |
| Anaph. Sub. | µı | ʔeµı | .cᴅ | nᵠcµc |
| Anaph. Obj. | µɔ | ʔɔᵉµɔ | .əᶜᴅ | nᵠəᶜµɔ |
| Generic | .əᶜ | | | |

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

### 3.6.1 | Last-clause pronouns

The anaphoric pronoun ⟨beʝ⟩ (accusative: ⟨beʝen⟩) is grammatically an other pronoun, and it refers to the previous clause said. Likewise, ⟨bedcʝ⟩ (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.

ᴅel–µəʝɔᵉ–ɑˡəµəɔ–ᵠcʝ
volume-cup-water-five
five cupfuls of water

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

ᴅel–µəʝɔᵉ–ɑˡəµəɔ–ᵠcʝ–ʃi
volume-cup-water-five-PR.NEAR.INTEGRAL
(arg1)'s five cupfuls of water

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

ᴅıʝı–ʔəɔʃi
person-old
old people
(Compare to ᴅıʝı ʔəɔʃi "person old-SENTIENT".)

## 3.8 | Possession

"X's Y" is translated as ⟨Y=ᴅɪ ɟᴄl X⟩. The possessive construction is also used to create appositives.

    Observe that possession marks the head, and ⟨–ᴅɪ⟩ is a clitic, not an affix, as in the following example:

ᴅǝᴅǝnᵠɔɟ–ɑˈǝμǝᵓ–ᴅɪ ɟᴄl ɟʰɪ.eɟ
DEF~rabbit-SING-water=GEN POS magician-SING
the magician's water rabbit

    In more casual speech, ⟨ɟᴄl⟩ 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 ⟨ᴅɪlɪn⟩ "(S) eats (O)".

|              | Nonpast | Past    |
|--------------|---------|---------|
| Near         | ᴅɪlɪn   | ᴅɪlɪʃ   |
| Far          | ᴅɪlɪn   | ᴅɪlcᵊʝ  |
| Other        | ᴅɪlɪ    | ᴅɪlcᵊ   |
| Anaph. Sub.  | ᴅɪle    | ᴅɪlel   |
| Anaph. Obj.  | ᴅɪlc.e  | ᴅɪlc.el |
| Generic      | ᴅɪlcɔ   | ᴅɪlcɔ   |

Table 4.2: Person-tense conjugations for verbs, using ⟨ɥeɑcn⟩ "(S) spreads (O)".

|              | Nonpast | Past     |
|--------------|---------|----------|
| Near         | ɥeɑcn   | ɥeɑcʃ    |
| Far          | ɥeɑɪn   | ɥeɑcᵊʝ   |
| Other        | ɥeɑɪ    | ɥeɑcᵊ    |
| Anaph. Sub.  | ɥeɑe    | ɥeɑel    |
| Anaph. Obj.  | ɥeɑc.e  | ɥeɑc.el  |
| Generic      | ɥeɑcᵊ   | ɥeɑcᵊ    |

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 / –lɔᵉ | –hɪ              |
| Atelic | –ᴅc / –ᴅᵊᶜ    | –ʝɪ        | –lcᵉ / –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:

ᴅɪlɪn sɪɾle ʃɔʝɔ.
eat-NEAR.NONPAST fish flower
Fish eat flowers.

ᴅɪlɪn sɪɾle ʃɔʝɔ, ᴅɪlɪn nɥɪʝɪ ʝɪ.
eat-NEAR.NONPAST fish flower, eat-NEAR.NONPAST cat PR.ANAPH_SUB
Fish eat flowers, and cats eat fish.

ᴅɪlɪn sɪɾle ʃɔʝɔ, ᴅɪle ʝɪʃɪbe.
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.)

ᴅɪlɪnle ʃɔʝɔ sɪɾle.
eat-NEAR.NONPAST-NEG flower fish
Flowers don't eat fish.

dʝɪ scɔ hʝchʝcɑel, ʝenɪn ʃɪ beʝ.
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.

dʝɪhɪ scɔ hʝchʝcɑel, ʝenɪn ʃɪ beʝ.
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 | −ᒐ | An action that is currently going on. Also used to distinguish static actions as opposed to dynamic (e. g. *wear* as opposed to *put on*). |
| Interrupted | ʃclᒐ | An action that was interrupted. |
| Perfect | −⅃ | An action that has already finished. Changes present tense to immediate past. Also used to distinguish dynamic actions as opposed to static (e. g. *put on* as opposed to *wear*). |
| Gnomic | −ʔ | A general truth or aphorism, or an action done habitually. |
| Gnomic dubitative | ʃclʔ | A general truth or aphorism that the speaker considers to be false. |
| Deontic necessity | −ᴍ | An action that the speaker insists on happening. |
| Epistemic necessity | ʃəᶜᴅᴍ | An action that the speaker infers that is happening. |
| Deontic potential | −ꞓ | An action that the speaker permits to occur. |
| Epistemic potential | ʃəᶜᴅꞓ | An action that the speaker infers that might happen. |
| Unexpected | −ʕ | An action that is unexpected (akin to using "but"). |
| Comparative | deʕ | Indicates an action of greater intensity than what was described in the previous clause. |
| Nonexclusive subject | ʃcᒐ | Indicates that the subject comprises not only of what is explicitly mentioned, but also other things. |
| Nonexclusive object | ʃcʔ | Indicates that the object comprises not only of what is explicitly mentioned, but also other things. |
| Nonexclusive argument | ʃcᴍ | Combination of both nonexclusive subject and nonexclusive object. |

An example:

ʃ∪ᑎ∪ʃᴅcᒐ ᴅe ᴨc, lcᴨc.elʕ dənᵠɪdɔᵉ−μɪ.
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

aaaaaaaaaaA 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":

ɖμcᵊ sc˧ ɑɔᵉlɔᵉl, ɒıl̀el sıɖɟcn.
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:

nı.ı nc ɟəɟəl–hı.ıμ, ncɒe hıhıɾ̀ɸᵊdɔᵉ.
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:

ɖμcᵊle sc˧ ɑɔᵉlɔᵉl, ɒıl̀elˤ sıɖɟcn.
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 ⟨ɑcn⟩, and precedes the clause in which the two are used:

ɑcᵊ \μcᵊɟe \ɾ̀ıμəl, ɒıl̀el ʃeμ.c˧n.
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.

ɑcᵊ \μcᵊɟe \ɾ̀ıμəl, ɑcᵊ ʃeμ.c˧ sıɖɟc, ɒıl̀cᵊ ɒıɒıɟıl μı.
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, ⟨nɪ.cn⟩ "wait, when" is also used for "if":

ʃɛμɪlehɪ, nɪ.cn ʃɪ beɹ, ɑehcn.
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 ⟨nɪ.cn⟩, 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:

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

## 4.5 | Comparatives

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

Consider the following sentences:

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

Semantically, they can be translated to:

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

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

The heart of comparatives in aaaaaaaaaaA is the quadrivalent verb ⟨ɒɔμɪn *a b f* ⊒⟩. Thus:

ɒɪlcᵓhɪ ʃɔμɔɒ–ɹʰcn, ɒɔμɪn sɪɹe nɥɪμɪ μɔ neʃ.
eat-GENERIC-Q flower-ACC.INT-how_many, CMP-NEAR fish cat PR.ANAPH_OBJ.INT >
Fish eat more flowers than cats.

ɒɪlcᵓhɪ .əᵓ–ɹʰcn ʃɔμɔ, ɒɔμɪn sɪɹe nɥɪμɪ μɪ neʃ.
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 |
|---|---|
| > | neʃ |
| < | ɑɔᵉʝ |
| = | ʃeᵓnᵠ |
| ≥ | ʃᵤl |
| ≤ | ɒcʝ |
| ≠ | .cᵉʝ |
| ≈ | ʮeʝ |
| ≫ | ɑʰe |
| ≪ | ɾın |

## 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:

ʃıφıʃ ɒe hᵤchᵤcɑen, nebel \ᵤcᵊʝen.
lose-NEAR.PAST PR.NEAR.SG DEF~book, give_to-ANAPH_SUB.PAST Riᵘse-ACC
I gave the book to Ryse.

## 4.7 | Transitivisation

Verbs that are intransitively (i. e. have no object passed at this time) can be turned into a causative form with the prefix ⟨φc–⟩:

ʃcɾcʃ ɑʮeɑʮenᵞe.
fall-NEAR.PAST DEF~coin
The coins fell.

ɒe φcʃcɾcᵊ ɑʮeɑʮenᵞe
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, ⟨ſᴄᴅᴄɴ⟩ 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:

ſᴄᴅᴄſ ɋꞟeɋꞟenᴖe ꞟıſı.
fall-NEAR.PAST DEF~coin grass
The coins fell on the grass.

ᴅe φᴄſᴄᴅᴄᵊ ɋꞟeɋꞟenᴖe, ſᴄᴅel ꞟıſıbe.
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:

ᴅəſınle ᴅıɟıle.
recall-NEAR-NEG person-NULL
No one knows.

ſɹᴄlſınle ᴅe ɟıꞟſıᴅı ɟᴄl ɟʰı.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 ⟨ɟᴄɴ⟩ 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 ⟨ɟᴄɴ⟩ 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 ⟨heɒʃɪ⟩ "large" and ⟨ʃəˀʃɪ⟩ "old".

| Class or person | Declined form | |
|---|---|---|
| Sentient | heɒʃɪ | ʃəˀʃɪ |
| Animate | heɒʃɪ | ʃəˀʃɪ |
| Inanimate | heɒʃe | ʃəˀʃɔᵉ |
| Measure | heɒʃɪɟ | ʃəˀʃɪɟ |
| Fluid | heɒʃeɹ | ʃəˀʃɔᵉɹ |
| Edible | heɒʃc | ʃəˀʃəᶜ |
| Inedible | heɒʃeˀ | ʃəˀʃɔ |
| Abstract | heɒʃcᵊ | ʃəˀʃə |
| Near | heɒʃɪμ | ʃəˀʃɪμ |
| Far | heɒʃɪμ | ʃəˀʃɪμ |
| Other | heɒʃeμ | ʃəˀʃeμ |
| Anaph. Sub. | heɒʃɪμ | ʃəˀʃɪμ |
| Anaph. Obj. | heɒʃeμ | ʃəˀʃeμ |
| Generic | heɒʃcˀμ | ʃəˀʃəˀμ |

## 5.1 | Conversion

A noun can be converted to a descriptor by appending ⟨–ɟɪ⟩.

A descriptor can be converted to an abstract noun meaning "the nature of being ~" by replacing the final ⟨–ɪ⟩ with ⟨–cneʃ⟩.

# 6 | Tree mode

As mentioned in section 4.3, anaphoric referents in a linked-list sentence are some-
times insufficient for expressing even simple sentence structures. While the easiest
method of resolving this issue is using definite nouns, aaaaaaaaaaA 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 ⟨nᕼı٩⟩ is used, and
disabled at the end of a sentence.

## 6.2 | Branch-switching

The aforementioned particle ⟨nᕼı٩⟩ marks the beginning of the right branch of the
tree. The right branch is ended by the particle ⟨nᕼı∆⟩, which causes the next clause
to join the left and right branches.

(N. B. ⟨nᕼı٩⟩ and ⟨nᕼı∆⟩ can occur only between clauses. If the particles are rep-
resented 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 ⟨nᕼı٩⟩ or the right predecessor ⟨nᕼı∆⟩.
This is done by marking the pronoun with ⟨−٩⟩ or ⟨−∆⟩.

Likewise, verbs can be modified with ⟨−٩⟩ or ⟨−∆⟩ to indicate which branch the
subject came from.

## 6.4 | Errors

The following are ungrammatical:

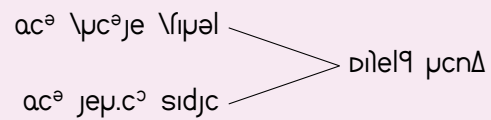- Using the particle ⟨nᕼı∆⟩ or the branched anaphoric pronouns when tree mode
  is disabled

- Using the particle ⟨ɲꞱᴵΔ⟩ other than to close a corresponding ⟨ɲꞱᴵꟼ⟩
- 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 ⟨ɲꞱᴵꟼ⟩ when the current branch is empty

## 6.5 | Example

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

ɑcᵊ \µcᵊɾe \ɾɪµǝl, ɲꞱᴵꟼ ɑcᵊ ɾeµ.cᵓ sɪdɾc, ɲꞱᴵΔ ᴅɪ̀lelꟼ µcɲΔ.

The resulting tree is shown below:

ɑcᵊ \µcᵊɾe \ɾɪµǝl
ɑcᵊ ɾeµ.cᵓ sɪdɾc
ᴅɪ̀lelꟼ µcɲΔ

# 7 | Numerals

aaaaaaaaaaA 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 | ꭑəᗡ |
| 1 | 1 | ɑcᵊl |
| 2 | ┘ | ʃıɟ |
| 3 | Ƨ | ɒ�ꟼeᵓn |
| 4 | Ⰿ | ɑˡıꭑ |
| 5 | Ⱶ | φcɹ |
| 6 | ſ | ᗡɥe |
| 7 | �q | ʃɔcɹ |
| 8 | Δ | dəᵓn |
| 9 | L | hed |
| 10 | F | bənᵠ |
| 11 | Ⅎ | nəᶜ |
| 12 | Ⱡ | leᵓ |
| 13 | ♯ | ɟʰcd |
| 14 | A | ɥın |
| 15 | Ɐ | ʔel |
| 16 | Ψ | .ıꭑ |

Note that digits above 9 use capital hacm letters.

31

## 7.2 | Numerals up to 19 · 17

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

Table 7.2: Multiples of 17, up to 18 · 17.

| base 10 | base v | word |
|---|---|---|
| 17 | 10 | selcə |
| 34 | ɟ0 | ᴅɪlcə |
| 51 | ʔ0 | ɒ⁴eˀlcə |
| 68 | ₥0 | ʃɪᴅlcə |
| 85 | ⱶ0 | φcɟʃcə |
| 102 | ϛ0 | ᴅɥelcə |
| 119 | 90 | ʃɟcɟʃcə |
| 136 | Δ0 | dəˀnə |
| 153 | L0 | helcə |
| 170 | F0 | bənᵠə |
| 187 | ⅎ0 | nəᶜlə |
| 204 | £0 | leˀlcə |
| 221 | ⧻0 | ɟʰcdlcə |
| 238 | A0 | ɥɪncə |
| 255 | Ɐ0 | ʔelɥcə |
| 272 | Ψ0 | .ɪlɥcə |
| 289 | Ð0 | ɟɪlscə |
| 306 | Ψ0 | heᴅdcə |

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

## 7.3 | Numerals up to 13 · 19 · 17 = 4199

These are represented by three digits. The multiples of 19 · 17 are listed below:

Table 7.3: Multiples of 19 · 17, up to 12 · 19 · 17.

| base 10 | base v | word |
|---|---|---|
| 323 | 100 | hɪɟɪʃɪn |
| 646 | ɟ00 | ʃɪɟɪʃɪn |
| 969 | ʔ00 | ɒ⁴eˀɟeʃɪn |
| 1292 | ₥00 | ɑˡɪɥɪʃɪn |
| 1615 | ⱶ00 | φcɟɪʃɪn |
| 1938 | ϛ00 | ᴅɥeɟɪʃɪn |
| 2261 | 900 | ʃɟcɟɪʃɪn |
| 2584 | Δ00 | dəˀnɪʃɪn |
| 2907 | L00 | heɑɪʃɪn |
| 3230 | F00 | bənᵠɪʃɪn |

| base 10 | base v | word |
|---:|---:|---|
| 3553 | ７00 | nəᶜjıſın |
| 3876 | £00 | leᵓjıſın |

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

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

The numeral for 4199 is ⟨lə.ɔləᶜ⟩, written as ⟨1:000⟩.

Likewise, two ⟨lə.ɔləᶜ⟩ is written as ⟨⅃:000⟩ and pronounced ⟨lə.ɔləᶜ–ſıj⟩, but the second ⟨lə.ɔləᶜ⟩ is one smaller than the first. In other words, ⟨⅃:000⟩ = 4199 + (4199 − 1) = 8397.

Table 7.4: "Multiples" of ⟨lə.ɔləᶜ⟩.

| "Multiple" | Difference from last | Total |
|---|---:|---:|
| (0) | | 0 |
| 1:000 | 4199 | 4199 |
| ⅃:000 | 4198 | 8397 |
| ?:000 | 4197 | 12594 |
| ℿ:000 | 4196 | 16790 |
| ⊦:000 | 4195 | 20985 |
| ... | | |
| £Ψ�V:000 | 3 | 8817897 |
| £Ψ५:000 | 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

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

And likewise, for some given $y$, the largest "multiple" of ⟨lə.ɔləᶜ⟩ not smaller than $y$ has the index

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

In other words, for any numeral ⟨$n_1$:$n_2$⟩, $n_1 + n_2$ must be less than 4199.

## 7.5 | Higher numerals

The *bases* of higher numerals $b_i$ can be derived from the recurrence relation

$$b_i = \begin{cases} 4199 & \text{if } i = 1 \\ \dfrac{b_{i-1} \cdot (b_{i-1} + 1)}{2} & \text{otherwise} \end{cases} \tag{7.3}$$

Then $b_i$ acts as a new triangular base. Equations 7.1 and 7.2 can be generalised to the following:

$$y(n, b_i) = \sum_{i=1}^{n} (b_i + 1 - n) \tag{7.4}$$

$$= \frac{1}{2} \cdot (n \cdot (2 \cdot b_i + 1 - n)) \tag{7.5}$$

$$N(y, b_i) = \left\lfloor \frac{1}{2} \cdot \left( 2 \cdot b_i - \sqrt{4 \cdot b_i^2 + 4 \cdot b_i + 1 - 8 \cdot y} \right) \right\rfloor \tag{7.6}$$

It follows that $y(n_1, b_i) + n_2$ is represented as $\langle n_1 :^i n_2 \rangle$ ($i$ colons), and such a numeral must satisfy $n_1 + n_2 < b_i$.

Here are the names of the bases themselves:

Table 7.5: Names of higher bases.

| Base | base v | word |
|------|-------:|------|
| $b_1$ | 1:000 | ꞁə.ɔꞁəᶜ |
| $b_2$ | 1::000:000 | ᴅeμcˀʃı |
| $b_3$ | 1:::000:000::000:000 | ʃıꞁμəˀ |
| $b_4$ | | ꞔencoı |

## 7.6 | Cardinal and ordinal numerals

Cardinal numerals compound to their antecedents; ordinal numerals use the posses-sive ⟨–ᴅɪ ꞁcꞁ⟩ construction:

ᴅcɑɪ–ꞔ�³n
child-three
three children

ᴅcɑɪꞁᴅɪ ꞁcꞁ ꞔᵟeˀn
child-SG=GEN POS three
the third child

# 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 ⟨–ne⟩ or ⟨–nɔᵉ⟩ to the noun.
Examples:

- ⟨hμcɑen⟩ *book* → ⟨hμcɑene⟩ *literature*

- ⟨hɔlı⟩ *cart* → ⟨hɔlınɔᵉ⟩ *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 diminuitive). ⟨ʃc–⟩ or ⟨ʃəᶜ–⟩ are prepended to nouns that begin in ⟨ɟ⟩ or ⟨l⟩, or ⟨ɟc–⟩ or ⟨ɟəᶜ–⟩ otherwise.
Examples:

- ⟨ᴅıɟı⟩ *person* → ⟨ɟcᴅıɟı⟩ *child*

- ⟨ʃɔμəᵓ⟩ *fruit* → ⟨ɟəᶜʃɔμəᵓ⟩ *unripe fruit*

- ⟨ɟeɾılcn⟩ *essay* → ⟨ʃcɟeɾılcn⟩ *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

- ɑ ↔ ɟʰ

- ɟ ↔ ɑʰ (in any position other than in a coda)

- h ↔ ɑˡ

- ʮ ↔ ɭ (in any position other than after a fricative in an onset)

- s ↔ l (in any position other than in a coda)

- ɥ ↔ o

For instance, ⟨ʮɪ.cʃɪn⟩ would be inverted to ⟨ɭɪ.əᶜɑɪn⟩.
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 | –eᵓnᵠ / –ɔnᵠ |
| Patient | –eᵓd / –ɔd |
| Location | –eᵓʮ / –ɔʮ |
| Instrument | –ɪʃɹɪ |

Table 8.2: An example with ⟨nəᵓbɪn⟩ *to steal*.

| Name | Derivation | Meaning |
|---|---|---|
| Agent | ncᵓdeᵓnᵠ | thief |
| Patient | ncᵓdeᵓd | stolen goods |
| Location | ncᵓdeᵓʮ | site of theft |
| Instrument | ncᵓdɪʃɹɪ | 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 |
|---|---|
| \ɥcᵊɹe | Personal (native) |
| \ʃiɰəl | Personal (native) |
| \ɰeɒcn | Personal (foreign) |
| dɰeˤɒe–\oɪɭɪ.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.

## 9.2 | Clausal names

These names comprise of one or more clauses. Due to the nature of clausal names, they are all considered native. Most of these names refer to places; personal clausal names are almost always nicknames or such. Orthographically, they are put into square brackets ⟨[]⟩.

Clausal names are used by saying them as their own clauses, then using an anaphoric pronoun to backreference the entity described by the name in question. The type of anaphoric pronoun used varies from name to name. It might be the anaphoric subject pronoun, the object pronoun or the last-clause pronoun.

We call the *referent* the subject, the object or the verb of the last clause, respectively depending on the type of anaphoric pronoun used to refer to the name. If the referent is a noun, it must be declined in the integral number without definiteness.

Here, as common in maps and such, the referent will be capitalised. However, other contexts that make the type of anaphoric pronoun to use clear do not use this type of capitalisation.

Table 9.2: Some examples of clausal names.

| Name | Type | Literal meaning |
|---|---|---|
| [AEЖѰЄꟻ d�étʃ[ɪ ʃɔɑɔ]] | Place | The trees *covered* the ground |
| [ᗡəʃɪ ФЖEꟳÐE µcᵊꟸ–\leɑµcᵓ] | Place | The *city* remembers the Šedrŷ star |
| [ɑcᵊ ꟸµ[ʃə–ᗡɪ ɹcl ɹəµ ʃʰɪ.en–ᗡɪ ɹcl ꟸɪɑcᵓ, φoeɹc.el ФЖEꟳÐEÐ] | Place | The *city* was founded by the warrior of the sun and the wizard of the moon |
| [ᗡeɹɪ \ѰꟼTFꟺ ɹcᵊl–selcᵊ] | Personal | *Gulto* takes care of 17 foxes |

An example of usage:

nɪ.ɪ bɪnen–bəɹ, [ᗡəʃɪ dµeᵓᗡe µcᵊꟸ–\leɑµcᵓ], ʃʰɪ ᗡɪᗡɪɹɪɹ .ᴄᴅ.
wait-OTHER year-future, (name), go-OTHER DEF~person-SG PR.ANAPH_SUB.SG
He will go to Muta Pröme Ryk-Šedrŷ next year.

# 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:

$$y = y_s + y_m \tag{10.1}$$
$$y_s = A_s \cdot (1 + A_{sa} \cdot \cos(\tau \cdot t)) \cdot \cos(2 \cdot \tau \cdot t) \tag{10.2}$$
$$y_m = A_m \cdot \left(1 + A_{ma} \cdot \cos\left(\frac{\tau \cdot t}{l_m}\right)\right) \cdot \cos\left(\frac{2 \cdot \tau \cdot (1 - l_m) \cdot t}{l_m}\right) \tag{10.3}$$

where:

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

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 aaaaaaaaaaA 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* ⟨ʮǝɒǝᶜ⟩ ($l_t$) – 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:

$$l_m/l_t \approx 29.80148 \tag{10.4}$$
$$\approx 4053/136 \tag{10.5}$$
$$l_y/l_m \approx 9.80299 \tag{10.6}$$
$$\approx 7215/736 \tag{10.7}$$

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.

## 10.2 | Months

Months follow a 136-month cycle wherein the $5n+2$-numbered months (zero-indexed, $n \in \mathbb{N}$) have 29 days and the other months have 30.

The names of months, on the other hand, are determined from their positions relative to the first month of the year:

Table 10.2: The months of the year.

| # | Name |
|---|------|
| 0 | ɑəˤbəˤ |
| 1 | µcᵉ.ɐ |
| 2 | ʃɹiɸɔl |
| 3 | ɒəµəʝi |
| 4 | ʔɹɛµiɒ |
| 5 | ʃɔᵉɒəʃ |
| 6 | nᵠəµəᶜʃ |
| 7 | lcbclɪ |
| 8 | bɔɒəˤʔ |
| 9* | .cjeˤµi |

Days within a month are indexed from one.

## 10.3 | Years

The lengths of the year follow a 736-year cycle as specified in Figure 10.1. The code used to generate this table can be found in Section A.2.

## 10.4 | Eras

Years are grouped further into *eras* ⟨lɪɥɪ⟩, which change on major historical events. The start of a new era resets the month and year cycle. Eras can also start in the middle of a year of the previous era; thus, the start of the year is different for each era. The *crossover date* of an era is the date of the era that coincides with the first day of the next; in other words, it is the date immediately after the last day of the era.

Table 10.3: The months of the year.

| Name | Crossover date | Days between | Cumulative |
|------|----------------|--------------|------------|
| lɪɥɪ–µəlɔᵉ | ʍ ʔɹɛµiɒ ʟɅʃ | 889726 | 889726 |
| lɪɥɪ–ləˤʃi | 1ʃ ʃɹiɸɔl ʕɅʕ | 642508 | 1532234 |
| lɪɥɪ–lɪʝnedcᵊ | Δ lcbclɪ ʃʔᵲ | 207366 | 1739600 |
| lɪɥɪ–nchel | (to present) | | |

The first day of ⟨lɪɥɪ–nchel⟩ coincides with the founding of the (not yet named).

Figure 10.1: Table of year lengths in a cycle.

```
    0123456789 |        0123456789 |        0123456789 |        0123456789
  0 XXXX9XXXX  |  19 XX9XXXX9XX  |  38 9XXXX9XXXX  |  57 XXX9XXXX9X
  1 9XXXX9XXXX |  20 XXX9XXXX9X  |  39 9XXXX9XXXX  |  58 XXX9XXXX9X
  2 9XXXX9XXXX |  21 XXX9XXXX9X  |  40 9XXXX9XXXX  |  59 XXX9XXXX9X
  3 9XXXX9XXXX |  22 XXX9XXXX9X  |  41 X9XXXX9XXX  |  60 XXXX9XXXX9
  4 9XXXX9XXXX |  23 XXX9XXXX9X  |  42 X9XXXX9XXX  |  61 XXXX9XXXX9
  5 9XXXX9XXXX |  24 XXX9XXXX9X  |  43 X9XXXX9XXX  |  62 XXXX9XXXX9
  6 9XXXX9XXXX |  25 XXX9XXXX9X  |  44 X9XXXX9XXX  |  63 XXXX9XXXX9
  7 X9XXXX9XXX |  26 XXX9XXXXX9  |  45 X9XXXX9XXX  |  64 XXXX9XXXX9
  8 X9XXXX9XXX |  27 XXXX9XXXX9  |  46 X9XXXX9XXX  |  65 XXXX9XXXX9
  9 X9XXXX9XXX |  28 XXXX9XXXX9  |  47 XX9XXXX9XX  |  66 XXXX9XXXXX
 10 X9XXXX9XXX |  29 XXXX9XXXX9  |  48 XX9XXXX9XX  |  67 9XXXX9XXXX
 11 X9XXXX9XXX |  30 XXXX9XXXX9  |  49 XX9XXXX9XX  |  68 9XXXX9XXXX
 12 X9XXXX9XXX |  31 XXXX9XXXX9  |  50 XX9XXXX9XX  |  69 9XXXX9XXXX
 13 X9XXXXX9XX |  32 XXXX9XXXX9  |  51 XX9XXXX9XX  |  70 9XXXX9XXXX
 14 XX9XXXX9XX |  33 XXXX9XXXX   |  52 XX9XXXX9XX  |  71 9XXXX9XXXX
 15 XX9XXXX9XX |  34 9XXXX9XXXX  |  53 XX9XXXXX9X  |  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 ⟨ꖴꖊꚞ⟩, 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

Figure 10.2: Hours in traditional timekeeping.
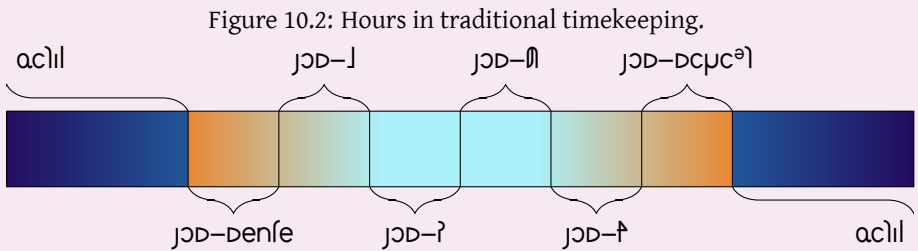


tidal day is divided into 23 equal parts ⟨lın⟩, each of which is divided into 80 equal parts ⟨jcne⟩, which are each divided into 40 equal parts ⟨bıɾəl⟩.

Of course, having 23 ⟨lın⟩ per tidal day requires predicting the next two high tides. For that reason, each day's ⟨lın⟩ are based on the length of the *previous* tidal day, such that each day might have more or less than 23 ⟨lın⟩.

# 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 -
        2 * tau * t)
19 y = y_s2 + y_m2
20 yp = diff(y, t)
21
22 # High and low tides occur at values of t where dy/dt = 0.
23
24 i = 0
25 time = 0
26 print(0)
27 while i < limit:
28   try:
29     time2 = find_root(yp == 0, time + 0.000000001, time + 0.35)
30     print(time2)
31     time = time2
32     i += 1
33   except:
34     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

```
my $digits-str = "0123456789TKXSNVFMD";
my @digits = $digits-str.comb;

sub convert-small-fwd($n, $pad = False) {
  die "$n must be < 4199" if $n >= 4199;
  my $a = $n div (19 * 17);
  my $b = ($n div 17) % 19;
  my $c = $n % 17;
  return
    (!$pad && $a == 0 ?? "" !! @digits[$a]) ~
    (!$pad && $b == 0 && $a == 0 ?? "" !! @digits[$b]) ~
    @digits[$c];
}

sub convert-small-back($s) {
  die "$s must be 3 chars or fewer" if $s.chars > 3;
  my $c = $digits-str.index($s.substr(* - 1, 1) // "0");
  my $b = $digits-str.index($s.substr(* - 2, 1) // "0");
  my $a = $digits-str.index($s.substr(* - 3, 1) // "0");
  return $c + 17 * ($b + 19 * $a);
}

sub triangle($n, $p) {
  return ($n * (2 * $p + 1 - $n)) div 2;
}

sub sqrt-floor($y) {
  die "$y is negative" if $y < 0;
  return $y if $y < 2;
  my $small = sqrt-floor($y +> 2) +< 1;
  my $large = $small + 1;
  return $small if $large * $large > $y;
  return $large;
}
sub sqrt-ceil($y) {
  my $n = sqrt-floor($y);
  return $n if $n * $n == $y;
  return $n + 1;
}

sub untriangle($y, $p) {
  return (2 * $p + 1 - sqrt-ceil(4 * $p * $p + 4 * $p - 8 * $y + 1))
    div 2;
}

my @powers = (4199);

for 0 .. 10 {
  my $p = @powers[* - 1];
  @powers.push: $p * ($p + 1) div 2;
}

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:

```
      1
   𝕸  F  L
   ꟼ  ?  A
  ──────────
   F  A  ꟗ

 1    1
      ﹂  ꟗ  ꟼ
      £  ѱ  ?
  ──────────
 1  ﹂  𝕸  Δ
```

### B.1.2 | Subtraction

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

```
    ꟗ   1?.
    Я   ʒ  A
    𝕸   F  L
  ──────────
    ﹂  £  ꟼ
```

### 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 ⟨ᴅ⟩, short for ⟨ᴅyɪne⟩ "one half". Thus, in hacm:

- ᴅ11 = L

- ᴅ101 = LL

- ᴅ110 = 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

$$x_a \;:^i\; y_a + x_b \;:^i\; y_b = (x_a + 1) \;:^i\; y_a + (x_b - 1) \;:^i\; y_b + (x_a - x_b + 1) \tag{B.1}$$
$$x_a \;:^i\; y_a + x_b \;:^i\; y_b = (x_a + x_b) \;:^i\; y_a + 0 \;:^i\; y_b + x_a \cdot x_b \tag{B.2}$$
$$= (x_a + x_b) \;:^i\; (y_a + y_b) + x_a \cdot x_b \tag{B.3}$$

# 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 aaaaaaaaaaA.

|  | Bilabial | Alveolar | Palatal | Velar | Glottal |
|---|---|---|---|---|---|
| Nasal | m | n | ñ | ŋ | |
| Plosive | p b | t d | ťɗ | k g | ʼ |
| Fricative | f | s | š | h | |
| (coarticulated) | þh | fh | | fš | |
| Affricate | | ts | tš | | |
| Lateral fricative | | ł | | | |
| Approximant | | r | j | w | |
| Lateral approximant | | l | | | |
| Trill | | ř | | | |

Table B.2: The vowels of aaaaaaaaaaA.

| Spread | Half-rounded | Rounded |
|---|---|---|
| i | y | ŷ |
| ï | u | û |
| 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 ⟨\⟩.
 ⟨ŋ⟩ 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

ʃəʃⅼ *nsent*    coward, knave
ʃəᶜμɔ *nined*    blood vessels

## | .

.cᴅen *ninanim*    house
.əˀʃəlɔ *nabst*    sadness, grief

## | ʃ

ʃιμc」 *ninanim*    river
ʃιʃⅼιn *v*    (S) fights (O)
ʃcᴅcn *v*    (S) falls on (O)
ʃɔμɔ *ninanim*    flower
ʃɔᵉdι *desc*    sufficient, wanted, wished-for
ʃeⅼι *desc*    complete, full, mature
ʃel *nsent*    who?
ʃιdɔ *nabst*    power, magic, motivation
ʃɔcⅼcn *v*    (S) stabs, stings (O)
ʃɔcⅼʃιn *v*    (S) wants (O), benefactive
ʃɔμəˀ *ninanim*    fruit
ʃᴅιｊcn *v*    (S) answers to (O)
ʃιφιn *v*    (S) loses, frees (O); (O) escapes
ʃιbμι *desc*    heavy
ʃɔɑɔʃ *ninanim*    tree
ʃɔɑɔʃ *nined*    wood
ʃedcn *v*    (S) buys (O)
ʃeμ.cˀ *nedib*    beef
ʃeμιn *v*    rain (S = other)

## | ⅼ

ⅼμιʃ⁴əˀ *nsent*    warrior
ⅼɔɔᵉⅼι *desc*    potent, powerful    not in a physical sense
ⅼι.cdι *nabst*    south
ⅼιⅼιn *v*    (S) makes a loud noise
ⅼιｊnedcᵊ *ninanim*    mirror
ⅼιn *v*    (S) is at (O), locational verb
ⅼιn *nmeas*    subdivision of the day    cf Grammar / Calendar / Subdivisions of the day / Modern timekeeping
ⅼιɒcˀ *ninanim*    moon
ⅼιчι *ninanim*    era
ⅼιᴅιｊ *nined*    stone
ⅼcⅼι *desc*    all, every
ⅼɔn *desc*    whole, entire
ⅼəˀ.ιn *v*    (S) needs (O)
ⅼəˀⅼι *desc*    old

## | ⎩

⎩μeˀncn *v*    (S) hunts for (O)
⎩cnιn *v*    (S) shoots an arrow to (O)
⎩cⅼ *nabst*    nature, disposition

## | ɟ

ɟıʋʃı *ninanim*     ring
ɟcn *v*     (S) is (O)
ɟcncn *v*     (S) attaches to, loves (O)
ɟcne *nmeas*     subdivision of the day cf Grammar / Calendar / Subdivisions of the day / Modern timekeeping
ɟcᵊl *nanim*     fox
ɟɔᴅ *nmeas*     subdivision of the day cf Grammar / Calendar / Subdivisions of the day / Traditional timekeeping
ɟɔᵉɑʋı *ninanim*     table
ɟe.ın *v*     (S) knows (O) answers (last clause)
ɟenın *v*     (S) is worried by (O)
ɟeɑˡc *nabst*     daytime
ɟenıˡcn *ninanim*     essay
ɟeᵊʃe *ninanim*     land, country
ɟəʃı *desc*     many, again
ɟəʋ *ninanim*     day, sun

## | ɟʰ

ɟʰı.en *nsent*     magician
ɟʰın *v*     (S) goes toward (O)
ɟʰıʋcn *v*     (S) creates (O)
ɟʰcn *nabst*     how many?
ɟʰeᵊncn *v*     (S) befriends (O)

## | n

nyıʋı *nanim*     cat
nı.cn *v*     (S) waits for/until (O), temporal verb, if
nı.eʋcn *v*     (S) covers, spans (O)
ncᴅcn *v*     (S) dances around (O)
nchel *ninanim*     group, organisation, order
ncᵊ *ninanim*     point     ncᵊ–ᴅyıne halfway point
nelcn *v*     (S) swims in (O)
neᴅ *desc*     male
nebın *v*     (S) gives something to (O)

neʋɑcn *v*     (S) hides from (O)
neʋcʃı *desc*     sudden
nel *nabst*     nature, temperament, disposition
neldı *nsent*     mind, brain
nəᵊnın *v*     (S) kills (O), (O) dies
nəᵊbın *v*     (S) steals from (O)
nəᵊbıʋɔᵉn *nsent*     thief

## | nᵠ

nᵠɔʋın *v*     (S) thinks, ponders about (O)

## | u

uɔᵉʋɔ *nfluid*     poison

## | ɑ

ɑʋen⁴e *ninanim*     coin
ɑcɟcn *v*     (S) obeys (O)
ɑcn *v*     (S) joins (O), and
ɑcnɑı *desc*     early
ɑcʋcn *v*     touch
ɑcʋe *nabst*     what
ɑcᵊl *desc*     female
ɑɔᵉlɔᵉ *ninanim*     spoon
ɑehcn *v*     (S) plays with (O)
ɑəᵊ *ninanim*     event, occurrence

## | ɑˡ

ɑˡıʋe *nabst*     quote, words, speech
ɑˡe *ninanim*     what
ɑˡəʋəᵊ *nfluid*     water

## | ɑʰ

ɑʰəᵊᴅɔ *nabst*     empathy

# ɔ

ɔɥɪne *nabst*   one half
ɔiʃɪn *v*   (S) eats (O)
ɔɪʃɪ *nsent*   person
ɔɪn *nedib*   rice
ɔcɑɪ *nsent*   child (offspring)
ɔcɥɒeˀ *nanim*   tongue
ɔcɥcˀʃ *nabst*   evening
ɔɔʃɪn *v*   (S) produces (O)
ɔɔʃɪn *v*   (S) is destroyed to make, for (O)
ɔɔˀɥəɒ *nanim*   scorpion
ɒeʃcn *v*   (S) gives birth to (O), (O) is born   (S) is not necessarily the mother; this can be either parent
ɒeʃcn *v*   (S) raises, takes care of, tends to (O)
ɒenʃe *nabst*   morning
ɒeɑɥɪn *v*   (S) stands on, is on (O)
ɒeφɪn *v*   (S) drowns in (O), (O) fills (S)
ɒedɪ *desc*   in return
ɒedcn *v*   (S) succeeds at (O), (S) does something to (O)
ɒeɥcᵉ *ninam*   opposite side
ɒeʃ *nmeas*   volume   in expressions such as ɒeʃ–ɥəʃɔᵉ "cupful"
ɒeʃʃɪ *desc*   similar
ɒeʃʃcn *v*   (S) imitates (O)
ɒəʃɪn *v*   (S) recalls (O)
ɒənᵠɔ *nanim*   rabbit

# Ω

ɑɪnʃɪn *v*   (S) is inside (O)
ɑɪɑɔ *ninanim*   back (body part)
ɑɪbə *nabst*   life, existence
ɑeʃeɒ *nfluid*   nitrogen
ɑeɒɪn *v*   (S) sleeps
ɑəˀɥɔ *ninanim*   pathway

# φ

φɥəˀnɪn *v*   (S) laughs at (O)

φoeʃɪn *v*   (S) founds (O)
φcʃʃɪ *desc*   well (not sick)
φcʃ *nabst*   five

# d

dʃɪʃʃɪ *nabst*   ground, floor
dʃen *nanim*   owl
dɥcn *v*   hold, carry, instrumental verb
dɥeˀɒe *ninanim*   city
dɪʃɥɪn *v*   (S) sits at (O)
dɪʃɪn *v*   (S) dislikes, objects to, disapproves of (O)
dɪʃe *ninanim*   landmass, domain
deɥɪn *v*   (S) wears, experiences (O)
deɥeˀ *nedib*   noodles
də.ɔn *nanim*   large animal
dənᵠɪ *ninanim*   knee

# b

bɪne *ninanim*   year
bɪʃəʃ *nmeas*   subdivision of the day   cf Grammar / Calendar / Subdivisions of the day / Modern timekeeping
bcʃcn *v*   (S) walks to (O)
bəʃ *nabst*   future, next (time period)
bəˀnᵠɪn *v*   (S) succumbs to their impulses

# h

hɥcɑen *ninanim*   book
hɪ.ɪɥ *nabst*   spring (season)
hɪʃnɔ *ninanim*   nose
hɪʃde *ninanim*   leaf
hɪʃəˀ *ninanim*   statue
hcn *v*   (S) claims that (O)
hcɥcn *v*   (S) is named (O)
hcˀeɒɪ *nedib*   food
hɔʃɪ *ninanim*   cart
heɑɪn *v*   (S) asks for, requests (O)
heɒʃɪ *desc*   large
həˀɒɪ *desc*   evil, malicious

## Ч

Чın   *v*    (S) sees (O), because Чın[O=ɿəμ] ("see the sun") = "wish"

## ꓷ

ꓷıɟın *v*    discipline, punish, constrain
ꓷcn *v*    (S) allows (O)

## μ

μı.cɾın *v*    (S) is beside (O)
μıɾı *nined*    grass
μıɾıbı *ninanim*    blade of grass
μıɹın *v*    (S) climbs, rises in (O)
μcᵊɹ *ninanim*    star
μeɑın *v*    (S) speaks to (O), (S) asks (O)
μeɑcn *v*    (S) spreads (O)
μeꟼı *ninanim*    place
μeᵓɟı *desc*    friendly, kind, considerate, nice

μəlı *desc*    late
μəlɔᵉ *nabst*    nighttime
μəjɔᵉ *ninanim*    cup
μəꓷəᶜ *nmeas*    tidal day
μəᵓɾɔ *nined*    gold

## s

sıdjc *nfluid*    soup
sıꓷe *nanim*    fish
seΩᵁın *v*    (S) perceives (O) non-visually

## o

oelje *nined*    forest

## l

leꟼ *nabst*    language
leꟼebı *ninanim*    a language