lɐ˥–\ɾɹɪɥɔ–ɒɪ ɹcl lɐ˥–ɒɪ ɹcl ɹeᵓɾ(e–\ɥcᵊɒɪ˥ɔ

aaaaaaaaaA, the language of *Rymako*

uruwi

IIIIIIIIIIIIIII

aaaaaaaaaaaaaaaa

A complete grammar

September 23, 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	ɣ̟	y
ɯ	ʉ̟	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^{ɥ}	n^{φ}	
Plosive	d b	ɾ Ω	ɾ^{ɥ} Ω^{ɥ}	ɿ φ	.
Fricative	ɑ	ȷ	ɩ	h	
(coarticulated)	ȷ^{h}	ɑ^{h}		ɑ^{ɩ}	
Affricate		ɾȷ	ɾɩ		
Lateral fricative		s			
Approximant		μ	ɥ	o	
Lateral approximant		l			
Trill		ɧ			

Table 1.4: The vowels of aaaaaaaaaaA.

Spread	Half-rounded	Rounded
c	c^{ə}	c^{ɔ}
ə^{c}	ə	ə^{ɔ}
e		e^{ɔ}
ɔ^{e}		ɔ
ı		

Rod signs are represented by the hacm digits ⟨1 ˩ ʔ ⋒ ↟ ʕ 9 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, /ɹɛfan/ (to speak) is written ⟨μeɑn⟩, but /ɹɛfin/ (to spread), which is less common, is written ⟨μeɑcn⟩, 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)
ᴅɪjɪ	2
nɪ.cn	1
.əᶨʃəlɔ	2
lɪɲedcᵊ	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:

ɾlen ɸeɑcᵊʝhɪ ncᵓɣ
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:

ɸeɑcᵊʝhɪ ncᵓ ɾlel, ɥɪɾ ɸɔ.
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	$\lvert x \rvert = 1$
Dual	$\lvert x \rvert = 2$

Table 3.2: The continuous grammatical numbers of aaaaaaaaaaA.

Number	Constraint on $x \in \mathbb{R}$
Nullary	$x = 0$
Subsingular	$\lvert x \rvert < 1$
Supersingular	$1 \leq \lvert x \rvert < 2$
Plural	$\lvert x \rvert \geq 2$ 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: ⟨ᴅɪjɪ⟩ “person”				
Nominative	ᴅɪjɪ	ᴅɪjɪl̀e	ᴅɪjɪj	ᴅɪjɪl
Accusative	ᴅɪjɪn	ᴅɪjɪnᵠl̀e	ᴅɪjɪnᴄj	ᴅɪjɪnɪl
Sentient: ⟨ʲʰɪ.en⟩ “magician”				
Nominative	ʲʰɪ.en	ʲʰɪ.eɾ̀e	ʲʰɪ.eȷ	ʲʰɪ.el
Accusative	ʲʰɪ.eɥcn	ʲʰɪ.enᵠl̀e	ʲʰɪ.eɥcȷ	ʲʰɪ.eɥcl
(Note that the final consonant is preserved only in the integral nominative form.)				
Animate: ⟨ᴅənᵠɔ⟩ “rabbit”				
Nominative	ᴅənᵠɔ	ᴅənᵠɔl̀ᴄᵉ	ᴅənᵠɔȷ	ᴅənᵠɔ.əᶜ
Accusative	ᴅənᵠɔn	ᴅənᵠɔnɾ̀ᴄᵉ	ᴅənᵠɔn	ᴅənᵠɔnəᶜ
Animate: ⟨ȷcᵊl⟩ “fox”				
Nominative	ȷcᵊl	ȷcᵊɾ̀e	ȷcᵊȷ	ȷcᵊ.c
Accusative	ȷcᵊɥcn	ȷcᵊnᵠl̀e	ȷcᵊɥcȷ	ȷcᵊɥc
Inanimate: ⟨hɪɾdəᵓ⟩ “statue”				
Nominative	hɪɾdəᵓ	hɪɾdəᵓl̀ᴄᵉ	hɪɾdəᵓl	hɪɾdəᵓȷ
Accusative	hɪɾdəᵓᴅ	hɪɾdəᵓᴅdɔᵉ	hɪɾdəᵓdɔᵉ	hɪɾdəᵓɑəᶜ
Inanimate: ⟨.cᴅen⟩ “house”				
Nominative	.cᴅen	.cᴅeɾ̀e	.cᴅel̀	.cᴅeȷ
Accusative	.cᴅeɥcᴅ	.cᴅeɥcᴅde	.cᴅeɥcde	.cᴅeɥcɑc

3.5.2 | Measurable classes

Table 3.4: Declensions for measurable nouns.

	Plural	Nullary	Subingular	Supersingular
Measure: ⟨ɥəᴅəᶜ⟩ “day (continuous)”				
Nominative	ɥəᴅəᶜ	ɥəᴅəᶜɥəᵓ	ɥəᴅəᶜɾ	ɥəᴅəᶜn
Accusative	ɥəᴅəᶜn	ɥəᴅəᶜɥəᵓn	ɥəᴅəᶜnɾ̀ᴄᵉ	ɥəᴅəᶜnəᶜn
Measure: ⟨ᴅel⟩ “volume” (in expressions such as ⟨ᴅel–ɥəȷɔᵉ⟩ “cupful”)				
Nominative	ᴅel	ᴅeɥcᵓ	ᴅeɥcɾ	ᴅeɥcn
Accusative	ᴅeɥcn	ᴅeɥcᵓn	ᴅeɥcnɾ̀e	ᴅeɥcncn
(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: ⟨ɑˡəɥəᵓ⟩ “water”	
Nominative	ɑᵊɥəᵓ
Accusative	ɑᵊɥəᵓn
Fluid: ⟨ɲel̀ᴇᴅ⟩ “nitrogen”	

	Mass
Nominative	ɑɐlɐɒ
Accusative	ɑɐlɐɒcn
(Here, the coda is preserved in the accusative as well.)	
Edible: ⟨ʃeɻ.cᵊ⟩ “beef”	
Nominative	ʃeɻ.cᵊ
Accusative	ʃeɻ.cᵊn
Edible: ⟨ɒɪn⟩ “rice”	
Nominative	ɒɪn
Accusative	ɒɪncn
Inedible: ⟨ɻəᵊʃɔ⟩ “gold”	
Nominative	ɻəᵊʃɔ
Accusative	ɻəᵊʃɔbe
Inedible: ⟨lɪrlɪj⟩ “stone”	
Nominative	lɪrlɪj
Accusative	lɪrlɪjde
Abstract: ⟨ɑʰəᵊɒɔ⟩ “empathy”	
Nominative	ɑʰəᵊɒɔ
Accusative	ɑʰəᵊɒɔnᵠ
Abstract: ⟨ɸcʝ⟩ “[the number] five”	
Nominative	ɸcʝ
Accusative	ɸcʝcnᵠ

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	ʃɪ	ʖeʃɪ	ɒe	ɑcɥc
Far	dɔ	ʖɔᵉdɔ	nəᶜ	bɥɪ
Other	nc	ʖenc	scᵓ	ʖɪɥc
Anaph. Sub.	ɥɪ	ʖeɥɪ	.cɒ	nᵠcɥc
Anaph. Obj.	ɥɔ	ʖɔᵉɥɔ	.əᶜɒ	nᵠəᶜɥɔ
Generic	.əᵓ			

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

3.6.1 | Last-clause pronouns

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

ɒel–ɥəȷɔᵉ–ɑˡəɥəᵓ–ɸcȷ
volume-cup-water-five
five cupfuls of water

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

ɒel–ɥəȷɔᵉ–ɑˡəɥəᵓ–ɸcȷ–ʃɪ
volume-cup-water-five-pr.near.integral
(arg1)'s five cupfuls of water

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

ɒɪȷɪ–ʖəᵓʃɪ
person-old
old people
(Compare to ɒɪȷɪ ʖəᵓʃɪ "person old-sentient".)

3.8 | Possession

“X’s Y” is translated as ⟨Y=ɒɪ ɟcl X⟩. The possessive construction is also used to create appositives.

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

ɒəɒənᵠɔɟ–ɑˡəɻəᵓ–ɒɪ ɟcl ɟʰɪ.eɟ
def~rabbit-sing-water=gen pos magician-sing
the magician’s water rabbit

In more casual speech, ⟨ɟcl⟩ 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 ⟨ɒlln⟩ "(S) eats (O)".

	Nonpast	Past
Near	ɒlln	ɒllɾ
Far	ɒlln	ɒlcᵊȷ
Other	ɒlɪ	ɒlcᵊ
Anaph. Sub.	ɒle	ɒlel
Anaph. Obj.	ɒlc.e	ɒlc.el
Generic	ɒlcↄ	ɒlcↄ

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

	Nonpast	Past
Near	ɥeɑcn	ɥeɑcɾ
Far	ɥeɑɪn	ɥeɑcᵊȷ
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	–ɒc / –ɒəᶜ	–ȷɪ	–lcᵊ / –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:

ɒɪlɪn sɪɾe ɾlɔɥɔ.
eat-near.nonpast fish flower
Fish eat flowers.

ɒɪlɪn sɪɾe ɾlɔɥɔ, ɒɪlɪn nɥɪɥɪ ɥɪ.
eat-near.nonpast fish flower, eat-near.nonpast cat pr.anaph_sub
Fish eat flowers, and cats eat fish.

ɒɪlɪn sɪɾe ɾlɔɥɔ, ɒɪle ɥɪɾíbe.
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.)

ɒɪlɪnle ɾlɔɥɔ sɪɾe.
eat-near.nonpast-neg flower fish
Flowers don't eat fish.

dɥɪ scᵓ hɥchɥcɑel, ɟenɪn ɾɪ beɟ.
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.

dɥɪhɪ scᵓ hɥchɥcɑel, ɟenɪn ɾɪ beɟ.
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	–ˀ	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	ʃcˀ	An action that was interrupted.
Perfect	–ɹ	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	ʃcʔ	A general truth or aphorism that the speaker considers to be false.
Deontic necessity	–ɱ	An action that the speaker insists on happening.
Epistemic necessity	ɲəᶜɒɱ	An action that the speaker infers that is happening.
Deontic potential	–ɬ	An action that the speaker permits to occur.
Epistemic potential	ɲəᶜɒɬ	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	ɲcˀ	Indicates that the subject comprises not only of what is explicitly mentioned, but also other things.
Nonexclusive object	ɲcʔ	Indicates that the object comprises not only of what is explicitly mentioned, but also other things.
Nonexclusive argument	ɲcɱ	Combination of both nonexclusive subject and nonexclusive object.

An example:

ʃuɱɾɒcˀ ɒe nc, ʃcnc.elʕ dənᵠɪdɔᵉ–ɥɪ.
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

aaaaaaaaaA 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”:

dɥcᵊ sc͡ɔ ɑɔᵉlɔᵉl, ɒɪlel sɪdʒcn.
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 ɟəɟəl–hɪ.ɪɥ, ncɒe hɪhɪɾdəᵓdɔᵉ.
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:

dɥcᵊle sc͡ɔ ɑɔᵉlɔᵉl, ɒɪlelʕ sɪdʒcn.
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 \ɥcᵊʝe \ɾɪɥəl, ɒɪlel ɾeɥ.cᵓn.
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 \ɥcᵊʝe \ɾɪɥəl, ɑcᵊ ɾeɥ.cᵓ sɪdʒc, ɒɪlcᵊ ɒɪɒɪʝɪl 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, ⟨ni.cn⟩ “wait, when” is also used for “if”:

ʃɐɥɪlehɪ, ni.cn ʃɪ bɐj, 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 ⟨ni.cn⟩, since our condition is an entire clause instead of a noun.

4.5 | 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:

ɾiɸɪɾ ɒe hɥchɥcɑen, nebel \ɥcᵊjen.
lose-near.past pr.near.sg def~book, give_to-anaph_sub.past Riᵘse-acc
I gave the book to Ryse.

4.6 | Transitivisation

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

ɾlcɾdcɾ ɑɥeɑɥenʸe.
fall-near.past def~coin
The coins fell.

ɒe ɸcɾlcɾdcᵊ ɑɥeɑɥenʸe
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, ⟨ɾlcɾdcn⟩ 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:

ɾlcɾdcɾ ɑɥeɑɥenʸe ɥɪʃɪ.
fall-near.past def~coin grass

The coins fell on the grass.

ᴅe ɸcſlcrdcᵃ ɑɥeɑɥenᵸe, ſlcrdel ɹıſibe.
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.

4.7 | 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:

ᴅəſınƚe ᴅıɹıƚe.
recall-near-neg person-null
No one knows.

ſjclſınƚe ᴅe ɹıɹſıᴅı ɹcl ɹʰı.eƚe.
want-near-neg pr.near.sg ring=gen pos magician-null
I don't want the rings of any magician.

4.8 | The copula

The copula ⟨ɹcn⟩ can take a noun as an object, in which case it can mean identity or membership. (Location is expressed with ⟨ın⟩ "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 ⟨ɹcn⟩ 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 ⟨hɐɒʃɪ⟩ “large” and ⟨ləˤʃɪ⟩ “old”.

Class or person	Declined form	
Sentient	hɐɒʃɪ	ləˤʃɪ
Animate	hɐɒʃɪ	ləˤʃɪ
Inanimate	hɐɒʃe	ləˤʃɔᵉ
Measure	hɐɒʃɪɟ	ləˤʃɪɟ
Fluid	hɐɒʃeɟ	ləˤʃɔᵉɟ
Edible	hɐɒʃc	ləˤʃəᶜ
Inedible	hɐɒʃeˤ	ləˤʃɔ
Abstract	hɐɒʃcᵊ	ləˤʃə
Near	hɐɒʃɪɥ	ləˤʃɪɥ
Far	hɐɒʃɪɥ	ləˤʃɪɥ
Other	hɐɒʃeɥ	ləˤʃeɥ
Anaph. Sub.	hɐɒʃɪɥ	ləˤʃɪɥ
Anaph. Obj.	hɐɒʃeɥ	ləˤʃeɥ
Generic	hɐɒʃcˤɥ	ləˤʃəˤɥ

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 ⟨–cnel⟩.

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 <nᵘᵎᑫ> is used, and disabled at the end of a sentence.

6.2 | Branch-switching

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

(N. B. <nᵘᵎᑫ> and <nᵘᵎΔ> 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 <nᵘᵎᑫ> or the right predecessor <nᵘᵎΔ>. This is done by marking the pronoun with <–ᑫ> or <–Δ>.

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

6.4 | Errors

The following are ungrammatical:

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

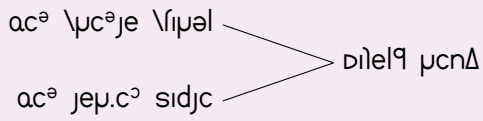
- Using the particle ⟨nʸɪΔ⟩ other than to close a corresponding ⟨nʸɪϥ⟩
- Using the unbranched anaphoric pronouns in clauses with two predecessors
- Using the branched anaphoric pronouns in clauses with one predecessor
- Starting a new branch with ⟨nʸɪϥ⟩ when the current branch is empty

6.5 | Example

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

ɑcᵊ \ɥcᵊɟe \ɾɪɥəl, nʸɪϥ ɑcᵊ ɟeɥ.cᵓ sɪdjc, nʸɪΔ ɒɪlelϥ ɥcnΔ.

The resulting tree is shown below:



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 9.8.

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	ɻəɒ
1	1	ɑcᵊl
2	ɹ	ʃɪȷ
3	ʔ	ɷʸeᵓn
4	𝍌	ɑˡɪɻ
5	ϟ	ɸcȷ
6	ʕ	ɒʏe
7	ϥ	ʃȷcȷ
8	Δ	dəᵓn
9	L	hed
10	F	bənᵠ
11	ⅎ	nəᶜ
12	ⅇ	leᵓ
13	♯	jʰcd
14	A	ʏɪn
15	∀	ɭel
16	Ψ	.ɪɻ

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	selcᵊ
34	J0	ɒɪlcᵊ
51	ʔ0	ɷʰeᶝlcᵊ
68	ℳ0	ɾɪɒlcᵊ
85	Ⱶ0	ɸcjʃcᵊ
102	ʕ0	ɒɥelcᵊ
119	ʠ0	ʃjcjʃcᵊ
136	Δ0	dəᶝnə
153	L0	helcᵊ
170	F0	bənᵠə
187	ⱷ0	nəᶜlə
204	ⱹ0	leᶝlcᵊ
221	ⱡ0	jʰcdlcᵊ
238	A0	ɥɪnᶜᵊ
255	∀0	ɭelɥcᵊ
272	Ψ0	.ɪlɥcᵊ
289	Ð0	jɪlscᵊ
306	Ψ0	heɒdcᵊ

Thus $y \cdot 17 + x$ is written ⟨x-y⟩.

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	hɪjɪʃɪn
646	J00	ɾɪjɪʃɪn
969	ʔ00	ɷʰeᶝjeʃɪn
1292	ℳ00	ɑˡɪɲɪʃɪn
1615	Ⱶ00	ɸcjɪʃɪn
1938	ʕ00	ɒɥejɪʃɪn
2261	ʠ00	ʃjcjɪʃɪn
2584	Δ00	dəᶝnɪɪʃɪn
2907	L00	heɑɪʃɪn
3230	F00	bənᵠɪʃɪn

base 10	base v	word
3553	ⅎ00	nəᶜɹɪɾɪn
3876	⅁00	lɐᵓɹɪɾɪn

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 ⟨Ꞗ:000⟩ and pronounced ⟨lə.ɔləᶜ–ɾɪɹ⟩, but the second ⟨lə.ɔləᶜ⟩ is one smaller than the first. In other words, ⟨Ꞗ:000⟩ = 4199 + (4199 − 1) = 8397.

Table 7.4: "Multiples" of ⟨lə.ɔləᶜ⟩.

"Multiple"	Difference from last	Total
(0)		0
1:000	4199	4199
Ꞗ:000	4198	8397
ʔ:000	4197	12594
ß:000	4196	16790
ᛋ:000	4195	20985
...		
⅁ㅍ∀:000	3	8817897
⅁ㅍΨ:000	2	8817899
1::000:000	1	8817900

Thus the nth "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

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

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 $\langle n_1{:}n_2 \rangle$, $n_1 + n_2$ 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	ɭəˌɔləᶜ
b_2	1::000:000	ɒɐɥcᵓʃɪ
b_3	1:::000:000::000:000	ʃɪlɥəᵓ
b_4		ɷencoɪ

7.6 | Cardinal and ordinal numerals

Cardinal numerals compound to their antecedents; ordinal numerals use the possessive ⟨–ɒɪ ɟcl⟩ construction:

ɒcɑɪ–ɷʰeᵓn
child-three
three children

ɒcɑɪjɒɪ ɟcl ɷʰeᵓn
child-sg=gen pos three
the third child

8 | 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.

8.1 | 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 8.1: Some examples of nominal names.

Name	Type
\ɥcᵊje	Personal (native)
\ʃɪɥəl	Personal (native)
\ɥeɒcn	Personal (foreign)
dɥeᵊɒe–\oɪlɪ.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.

8.2 | 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 8.2: Some examples of clausal names.

Name	Type	Literal meaning
[AEɤΨϵᵌ dɳlʃɹ ʃɔɑɔl]	Place	The trees *covered* the ground
[ɒəʃɪ ɸɤEᵌÐE ɥcᵊl–\lɐɯɥcᵓ]	Place	The *city* remembers the Šedrŷ star
[ɑcᵊ ɭɥɪʃə–ɒɪ ɟcl ɟəɥ ɟʰɪ.en–ɒɪ ɟcl ɭɪɑcᵓ, ɸoeɟc.el ɸɤEᵌÐEÐ]	Place	The *city* was founded by the warrior of the sun and the wizard of the moon
[ɒeɟɹ \Ψ∃TF϶ ɟcᵊl–selcᵊ]	Personal	*Gulto* takes care of 17 foxes

An example of usage:

ɲɪ.ɪ bɪnen–bəɟ, [ɒəʃɪ dɥeᵓɒe ɥcᵊl–\lɐɯɥcᵓ], ɟʰɪ ɒɪdɪɟɪɟ .cɒ.
wait-other year-future, (name), go-other def~person-sg pr.anaph_sub.sg
He will go to Muta Pröme Ryk-Šedrŷ next year.

9 | 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 9.1: 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

9.1 | Tides

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

$$y = y_s + y_m \tag{9.1}$$

$$y_s = A_s \cdot (1 + A_{sa} \cdot \cos(\tau \cdot t)) \cdot \cos(2 \cdot \tau \cdot t) \tag{9.2}$$

$$y_m = A_m \cdot \left(1 + A_{ma} \cdot \cos\left(\frac{\tau \cdot t}{l_m}\right)\right) \cdot \cos\left(\frac{2 \cdot \tau \cdot (1 - l_m) \cdot t}{l_m}\right) \tag{9.3}$$

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 9.6 for a Sage program to do so.

As the calendar used by aaaaaaaaaA 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* ⟨ɥəɒəᶜ⟩ (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:

$$\begin{aligned}
l_m/l_t &\approx 29.80148 && (9.4)\\
&\approx 4053/136 && (9.5)\\
l_y/l_m &\approx 9.80299 && (9.6)\\
&\approx 7215/736 && (9.7)
\end{aligned}$$

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.

9.2 | 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 9.2: The months of the year.

#	Name
0	ɑəᵓbəᵓ
1	ɥcᵊ.eɒ
2	ʃɹɪɸɔl
3	ɒəɥəɹɪ
4	ʅdeɥɪɒ
5	ʃɔᵉɒəɾ
6	nᵠəɥəᶜɾ
7	lcbclɪ
8	bɔɒəᵓʅ
9*	.cɹeᵓɥɪ

Days within a month are indexed from one.

9.3 | Years

The lengths of the year follow a 736-year cycle as specified in Figure 9.1. The code used to generate this table can be found in Section 9.7.

9.4 | Eras

Years are grouped further into *eras* ⟨ʅɪɥɪ⟩, 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 9.3: The months of the year.

Name	Crossover date	Days between	Cumulative
ʅɪɥɪ–ɥəlɔᵉ	𝍐 ʅdeɥɪɒ LΔJ	889726	889726
ʅɪɥɪ–ləᵓɾɪ	1J ʃɹɪɸɔl ϛΛϛ	642508	1532234
ʅɪɥɪ–ʅɪɹnedcᵊ	Δ lcbclɪ ɹʔɬ	207366	1739600
ʅɪɥɪ–nchel	(to present)		

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

Figure 9.1: Table of year lengths in a cycle.

```
   0123456789 |     0123456789 |     0123456789 |     0123456789
 0 XXXXX9XXXX |  19 XX9XXXX9XX |  38 9XXXX9XXXX |  57 XXX9XXXX9X
 1 9XXXX9XXXX |  20 XXX9XXXX9X |  39 9XXXX9XXXX |  58 XXX9XXXX9X
 2 9XXXX9XXXX |  21 XXX9XXXX9X |  40 9XXXXX9XXX |  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 XXXXX9XXXX |  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
```

9.5 | Subdivisions of the day

Lek-Tsaro has two systems for subdividing the day.

9.5.1 | Traditional timekeeping

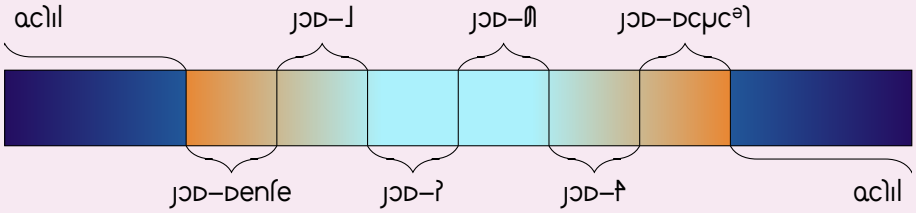
The traditional system of timekeeping relies on subdivisions of the solar day. As shown in Figure 9.2, the period between sunrise and sunset are divided into six equally-sized parts ⟨ɟɔɒ⟩, and the night is considered a separate category. This implies that the length of the "hours" depends on the seasons.

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

9.5.2 | 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 9.2: Hours in traditional timekeeping.



tidal day is divided into 23 equal parts ⟨lɪn⟩, each of which is divided into 80 equal parts ⟨ɟcne⟩, which are each divided into 40 equal parts ⟨bɪɾdəl⟩.

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

Listings of programs

9.6 | workfiles/7/tides.sage

```
# How many values to output

limit = int(sys.argv[1]) if len(sys.argv) > 1 else 1000

# :P

tau = 2 * pi

t = var("t")

# Constants

A_s = 0.675; A_sa = 0.0532; A_m = 1.267; A_ma = 0.176; l_m = 30.80152

# Solar component
y_s2 = A_s * (1 + A_sa * cos(tau * t)) * cos(2 * tau * t)
# Lunar component
y_m2 = A_m * (1 + A_ma * cos(tau * t / l_m)) * cos(2 * tau * t / l_m -
    2 * tau * t)
y = y_s2 + y_m2
yp = diff(y, t)

# High and low tides occur at values of t where dy/dt = 0.

i = 0
time = 0
print(0)
while i < limit:
  try:
    time2 = find_root(yp == 0, time + 0.000000001, time + 0.35)
    print(time2)
    time = time2
    i += 1
  except:
    time += 0.01
```

workfiles/7/tides.sage

9.7 | workfiles/7/bins.pl6

```
# CONSTANTS

constant \MONTHS_PER_YEAR_CYCLE = 7215;
constant \YEARS_PER_YEAR_CYCLE = 736;
constant \AVG_MONTHS_PER_YEAR = MONTHS_PER_YEAR_CYCLE /
    YEARS_PER_YEAR_CYCLE;

# COMPUTATION
# For each year, take as many months as are needed
# in order to cycle to the next.

my $c = 0;
my @k;

for 0 ..^ YEARS_PER_YEAR_CYCLE -> $i {
  my $need = 1 - ($c - floor($c));
  my $objs = ceiling($need * AVG_MONTHS_PER_YEAR);
  @k[$i] = $objs;
  $c += $objs / AVG_MONTHS_PER_YEAR;
}

# DISPLAY

my \cols = 4;
my $len = @k.elems;

say ("    0123456789" xx cols).join(" | ");

my \total-rows = ceiling($len / 10);
my \rows = ceiling(total-rows / cols);

for 0 ..^ rows -> $j {
  for 0 ..^ cols -> $p {
    print(" | ") if $p != 0;
    my $q = $j + rows * $p;
    next if $q >= total-rows;
    printf("%3d ", $q);
    for 0 ..^ 10 {
      my $i = 10 * $q + $_;
      if $i >= $len { print " "; }
      else {
        print "0123456789XE".substr(@k[$i], 1);
      }
    }
  }
  say "";
}
```

workfiles/7/bins.pl6

9.8 | 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) {
```

```
  # base case
  if $i == 0 {
    return convert-small-fwd($n, $pad);
  }
  # recursive
  my $super = untriangle($n, @powers[$i - 1]);
  my $infra = $n - triangle($super, @powers[$i - 1]);
  if $super == 0 && !$pad {
    return convert-large-fwd-h($infra, $i - 1, False);
  }
  return
    convert-large-fwd-h($super, $i - 1, $pad) ~
    (":" x $i) ~
    convert-large-fwd-h($infra, $i - 1, True);
}

sub convert-large-fwd($n, $pad = False) {
  my $i = 0;
  ++$i while @powers[$i] <= $n;
  convert-large-fwd-h($n, $i, $pad);
}

sub convert-large-back($s) {
  # Find the longest run of colons
  my @matches = ($s ~~ m:g/":"+/); #/"
  if (!@matches) {
    return convert-small-back($s);
  }
  my $longest-match = @matches.max(*.chars);
  my $i = (~$longest-match).chars;
  my $left = $s.substr(0, $longest-match.from);
  my $right = $s.substr($longest-match.to);
  my $sup = convert-large-back($left);
  my $inf = convert-large-back($right);
  return triangle($sup, @powers[$i - 1]) + $inf;
}

multi MAIN(Int :$fwd) {
  say convert-large-fwd($fwd);
}
multi MAIN(Str :$back) {
  say convert-large-back($back);
}
```

workfiles/7/conno.pl6

9.9 | workfiles/7/count-days.pl6

```
# Count the number of days between 1/0/0 and D/M/Y, inclusive.

# CONSTANTS

constant \MONTHS_PER_YEAR_CYCLE = 7215;
constant \YEARS_PER_YEAR_CYCLE = 736;
```

```
constant \AVG_MONTHS_PER_YEAR = MONTHS_PER_YEAR_CYCLE /
    YEARS_PER_YEAR_CYCLE;
constant \MONTHS_PER_MONTH_CYCLE = 136;
constant \DAYS_PER_MONTH_CYCLE = 4053;

# COMPUTATION
# For each year, take as many months as are needed
# in order to cycle to the next.

my $c = 0;
my @k = (0);

for 0 ..^ YEARS_PER_YEAR_CYCLE -> $i {
  my $need = 1 - ($c - floor($c));
  my $objs = ceiling($need * AVG_MONTHS_PER_YEAR);
  @k[$i + 1] = $objs;
  $c += $objs / AVG_MONTHS_PER_YEAR;
}

my @cumk = [\+] @k;

sub months-before-year($year) {
  my $whole-cycles = $year div YEARS_PER_YEAR_CYCLE;
  my $remainder = $year % YEARS_PER_YEAR_CYCLE;
  return $whole-cycles * MONTHS_PER_YEAR_CYCLE + @cumk[$remainder];
}

my @m = (0);

for 0 ..^ MONTHS_PER_MONTH_CYCLE -> $i {
  @m.push: ($i % 5 == 2) ?? 29 !! 30;
}

my @cumm = [\+] @m;

sub days-before-month($month) {
  my $whole-cycles = $month div MONTHS_PER_MONTH_CYCLE;
  my $remainder = $month % MONTHS_PER_MONTH_CYCLE;
  return $whole-cycles * DAYS_PER_MONTH_CYCLE + @cumm[$remainder];
}

sub days-before-date($d2, $m, $y) {
  my $d = $d2 - 1; # d is 0-indexed
  my $bm = months-before-year($y) + $m;
  return days-before-month($bm) + $d;
}

sub MAIN($d2, $m, $y) {
  say days-before-date($d2, $m, $y);
}
```

workfiles/7/count-days.pl6

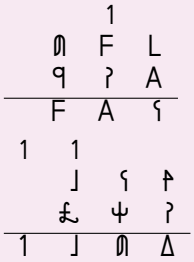
Arithmetic in base *v*

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

9.10 | Operations on small numbers

9.10.1 | Additions

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



9.10.2 | Subtraction

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

9.10.3 | 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.

9.10.4 | 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 ⟨ᴅʏɪɴᴇ⟩ "one half". Thus, in hacm:

- ᴅ11 = L
- ᴅ101 = LL
- ᴅ110 = F0

9.10.5 | Multiplication

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

9.11 | Operations on larger numbers

9.11.1 | 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

$$x_a :^i y_a + x_b :^i y_b = (x_a + 1) :^i y_a + (x_b - 1) :^i y_b + (x_a - x_b + 1) \tag{9.8}$$
$$x_a :^i y_a + x_b :^i y_b = (x_a + x_b) :^i y_a + 0 :^i y_b + x_a \cdot x_b \tag{9.9}$$
$$= (x_a + x_b) :^i (y_a + y_b) + x_a \cdot x_b \tag{9.10}$$

Romanisation

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

Table 9.4: The consonants of aaaaaaaaaaA.

	Bilabial	Alveolar	Palatal	Velar	Glottal
Nasal	m	n	ñ	ŋ	
Plosive	p b	t d	ť ď	k g	’
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 9.5: 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 ⟨\⟩.

Ŋaŋaŋa

Dictionary

.

.ᴄᴅᴇɴ *ninanim* house
.ᵊʃəlɔ *nabst* sadness, grief

ʃ

ʃʟiᖶin *v* (S) fights (O)
ʃʟᴄrdᴄn *v* (S) falls on (O)
ʃʟɔɻɔ *ninanim* flower
ʃʟɔᵉdɪ *desc* sufficient, wanted, wished-for
ʃʟeᖶɪ *desc* complete, full, mature
ʃʟel *nsent* who?
ʃɻɪdɔ *nabst* power, magic, motivation
ʃɻᴄlʃɪn *v* (S) wants (O)
ʃrdɪɻᴄn *v* (S) answers to (O)
ʃɪɸɪn *v* (S) loses, frees (O); (O) escapes
ʃɪbɻɪ *desc* heavy
ʃɔɑɔᖶ *ninanim* tree
ʃedᴄn *v* (S) buys (O)
ʃeɻ.cᵊ *nedib* beef
ʃeɻɪn *v* rain (S = other)
ʃəᖶɪ *nsent* coward, knave

ᖶ

ᖶɻɪʃʰəᵊ *nsent* warrior
ᖶoɔᵉlɪ *desc* potent, powerful not in a physical sense
ᖶiᖶɪn *v* (S) makes a loud noise
ᖶɪɻnedcᵊ *ninanim* mirror
ᖶɪn *v* (S) is at (O), locational verb
ᖶɪn *nmeas* subdivision of the day cf Grammar / Calendar / Subdivisions of the day / Modern timekeeping
ᖶɪɑcᵊ *ninanim* moon
ᖶɪɻɪ *ninanim* era
ᖶɪrdɪɻ *nined* stone
ᖶcᖶɪ *desc* all, every
ᖶɔn *desc* whole, entire
ᖶəᵊʃɪ *desc* old

l

lɻeᵊncn *v* (S) hunts for (O)
lᴄnɪn *v* (S) shoots an arrow to (O)

ɻ

ɻɪɻʃɪ *ninanim* ring
ɻcn *v* (S) is (O)
ɻᴄncn *v* (S) attaches to, loves (O)
ɻᴄne *nmeas* subdivision of the day cf Grammar / Calendar / Subdivisions of the day / Modern timekeeping

ɲcᵊl *nanim* fox
ɲɔɒ *nmeas* subdivision of the day cf Grammar / Calendar / Subdivisions of the day / Traditional timekeeping
ɲe.ɪn *v* (S) knows (O) answers (last clause)
ɲenɪn *v* (S) is worried by (O)
ɲeɑˡc *nabst* daytime
ɲeᵓʃɭe *ninanim* land, country
ɲəʃɪ *desc* many, again
ɲəɥ *ninanim* day, sun

ɲʰ

ɲʰɪ.en *nsent* magician
ɲʰɪn *v* (S) goes toward (O)
ɲʰɪɥcn *v* (S) creates (O)
ɲʰeᵓncn *v* (S) befriends (O)

n

nɥɪɥɪ *nanim* cat
nɪ.cn *v* (S) waits for/until (O), temporal verb, if
ncɒcn *v* (S) dances around (O)
nchel *ninanim* group, organisation, order
neɒ *desc* male
nebɪn *v* (S) gives something to (O)
neɥɑcn *v* (S) hides from (O)
nel *nabst* nature, temperament, disposition
neldɪ *nsent* mind, brain
nəᵓnɪn *v* (S) kills (O), (O) dies

ɑ

ɑɥenᵠe *ninanim* coin
ɑcn *v* (S) joins (O), and
ɑcnɑɪ *desc* early
ɑcᵊl *desc* female
ɑɔᵉlɔᵉ *ninanim* spoon
ɑehcn *v* (S) plays with (O)
ɑəᵓ *ninanim* event, occurrence

ɑˡ

ɑˡe *ninanim* what
ɑˡəɥəᵓ *nfluid* water

ɑʰ

ɑʰəᵓɒɔ *nabst* empathy

ɒ

ɒiɭɪn *v* (S) eats (O)
ɒɪɟɪ *nsent* person
ɒɪn *nedib* rice
ɒcɑɪ *nsent* child (offspring)
ɒcɥcᵊl *nabst* evening
ɒɔʃɪn *v* (S) produces (O)
ɒeɟcn *v* (S) raises, takes care of, tends to (O)
ɒenʃe *nabst* morning
ɒedcn *v* (S) succeeds at (O), (S) does something to (O)
ɒel *nmeas* volume in expressions such as ɒel–ɥəɟɔᵉ "cupful"
ɒelɭɪ *desc* similar
ɒelɭcn *v* (S) imitates (O)
ɒəʃɪn *v* (S) recalls (O)
ɒənᵠɔ *nanim* rabbit

ɑ

ɑɪbə *nabst* life, existence
ɑelɒ *nfluid* nitrogen

ɸ

ɸɥəᵓnɪn *v* (S) laughs at (O)
ɸoeɟɪn *v* (S) founds (O)
ɸcʃɭɪ *desc* well (not sick)
ɸcɟ *nabst* five

d

dɾɪʃɹ *nabst* ground, floor
dɾɐn *nanim* owl
dɥcn *v* hold, carry, instrumental verb
dɥeᶝɒe *ninanim* city
dɪʃɥɪn *v* (S) sits at (O)
dɪʃɪn *v* (S) dislikes, objects to, disapproves of (O)
deɥeᶝ *nedib* noodles
də.ɔn *nanim* large animal
dənᵠɪ *ninanim* knee

b

bɪne *ninanim* year
bɪɾdəl *nmeas* subdivision of the day cf Grammar / Calendar / Subdivisions of the day / Modern timekeeping
bəɹ *nabst* future, next (time period)

h

hɥcɑen *ninanim* book
hɪ.ɪɥ *nabst* spring (season)
hɪɾdəᶝ *ninanim* statue
hcɥcn *v* (S) is named (O)
heɒʃɪ *desc* large
həᶝɒɪ *desc* evil, malicious

ɥ

ɥɪn *v* (S) sees (O), because ɥɪn[O=jəɥ] (“see the sun”) = “wish”

ɾ

ɾɪjɪn *v* discipline, punish, constrain
ɾcn *v* (S) allows (O)

ɥ

ɥɪʃɪ *nined* grass
ɥɪʃɪbɪ *ninanim* blade of grass
ɥcᵊl *ninanim* star
ɥeɑɪn *v* (S) speaks to (O), (S) asks (O)
ɥeɑcn *v* (S) spreads (O)
ɥeᶝjɪ *desc* friendly, kind, considerate, nice
ɥəlɪ *desc* late
ɥəlɔᵉ *nabst* nighttime
ɥəjɔᵉ *ninanim* cup
ɥəɒəᶜ *nmeas* tidal day
ɥəᶝʃɔ *nined* gold

s

sɪdjc *nfluid* soup
sɪɾe *nanim* fish
seɑᵠɪn *v* (S) perceives (O) non-visually

l

lel *nabst* language
lelebɪ *ninanim* a language