

ᠮᠠᠨᠳᠡᠨᠠᠨ

Contents

ᠨᠠᠵᠤ

- 1. ᠬᠣᠭᠠᠨ
- 2. ᠢᠵᠢᠨᠠᠨᠠᠨᠠᠨᠠᠨ

English

- Homage
- Introduction
- Alphabet
- Grammar

Mandenkan

ᠮᠠᠨᠳᠡᠨᠠᠨ

// add logo

1. ᠬᠣᠭᠠᠨ

homage in nko

2. ᠢᠵᠢᠨᠠᠨᠠᠨᠠᠨᠠᠨ

ᠢᠵᠢᠨᠠᠨᠠᠨᠠᠨᠠᠨ

introduction in nko

Homage

In the name of Allah, most merciful and most gracious, I
embark upon this qu'est of self réflexion and observation. But,
as a
tree does not Spring without ils roots, thus do i begin by
honoring and paying hommage to those who have pave my
way.

I pay hommage to mousolo, womanhood, for we all dérive form
a woman. I pay hommage to my kindfolks, my household, my
clan, my tripes and mankind. I invoke the benevolent spirits of
my ancestors and the jinns for muse and Guidance for a sleep
that strays from ils pack is meal to Predators. May you guide
me on a righteous path.

To my readers, I appreciate you.

We shall explore numerous subjects in this book. From
philosophy,psychology,love and hatred

Introduction

Language is the voice of Man. it is the only aspect defining, uniting and differentiating men. All
voice is one and as stated in the holy quran, mankind spoke one language before it was confused
and broken into nations and tribes.

The language spoken by men is Mandekan, voice of the children of man and its speaker is a Mandeka, meaning one whose root lies in Man.

The table below shows the correspondence in mandenkan, voice of the child of man.

Table 1 Leafs of Man

ᐅᐅᐅ	Transliteration	Translation
ᐅᐅ	Man	Man
ᐅᐅᐅ		Men
ᐅᐅ	den	Child
ᐅᐅ		rererere
ᐅᐅ	ᐅᐅ	Voice
ᐅᐅᐅ		Voices
ᐅᐅᐅᐅ		Child of Man
ᐅᐅᐅᐅᐅ		Child of Men
ᐅᐅᐅᐅᐅᐅ		Children of Man
ᐅᐅᐅᐅᐅᐅᐅ		Children of Men
ᐅᐅᐅᐅᐅᐅᐅᐅ		The Children of Men
ᐅᐅ	ᐅᐅᐅᐅᐅᐅᐅᐅᐅᐅ	rererere
ᐅᐅ	ᐅᐅᐅᐅᐅᐅᐅᐅᐅᐅᐅ	rererere
ᐅᐅ	ᐅᐅᐅᐅᐅᐅᐅᐅᐅᐅᐅ	rererere
ᐅᐅ	ᐅᐅᐅᐅᐅᐅᐅᐅᐅᐅᐅᐅ	rererere
ᐅᐅᐅᐅᐅᐅᐅᐅ	Mandingdo	
ᐅᐅ	Father	
ᐅᐅ ᐅᐅᐅ	Father	

Definition

This is the placeholder for definition of therms such god, the god, a god, a god of, the god of
attributes, feature, qualities, entity, inheritence, encapsulation

Table 2 Definitions

מִצָּח	Transliteration	Translation
Attr	Man	Man
מִצָּחִים		Men

Mathematics note

```
#imports
from binarytree import Node
```

```
root = Node('Word')
left = root.left = Node('attribute')
right = root.right = Node('attribute')

root.left.left = Node('quality')
root.left.right = Node('quality')

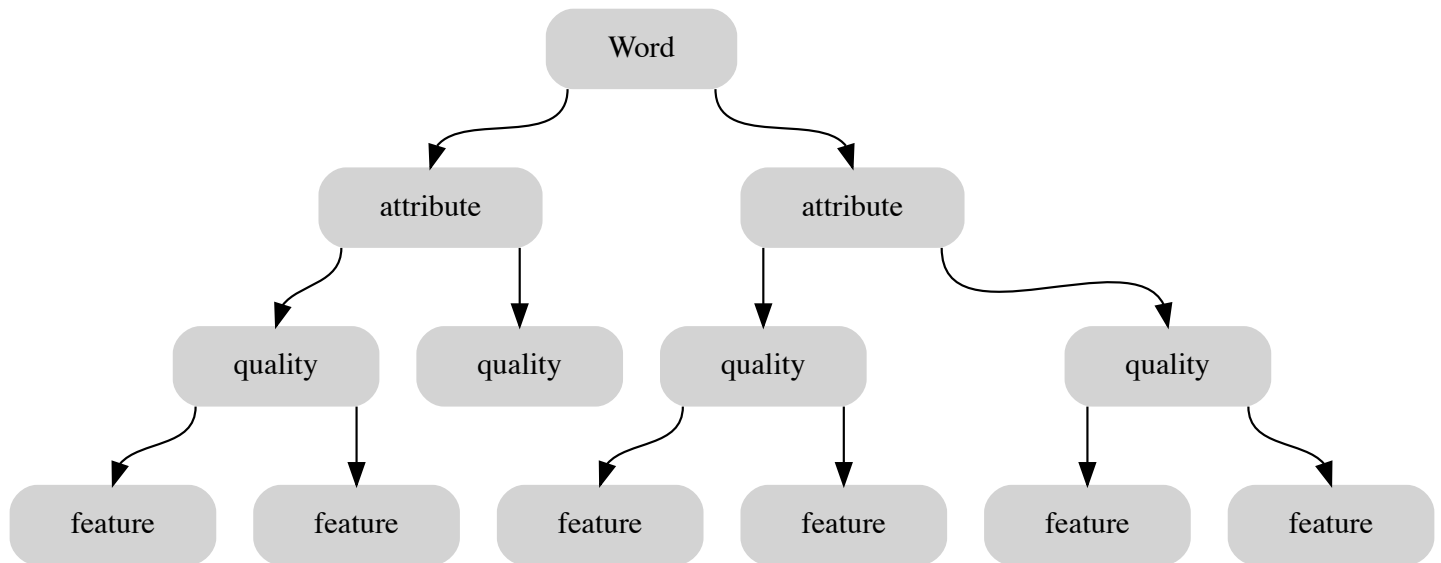
root.right.left = Node('quality')
root.right.right = Node('quality')

root.right.left.left = Node('feature')
root.right.left.right = Node('feature')

root.right.right.left = Node('feature')
root.right.right.right = Node('feature')

root.left.left.left = Node('feature')
root.left.left.right = Node('feature')

root.graphviz()
```



```
from treelib import Node, Tree
tree = Tree()
tree.create_node("Harry", "harry") # root node
tree.create_node("Jane", "jane", parent="harry")
tree.create_node("Bill", "bill", parent="harry")
tree.create_node("Diane", "diane", parent="jane")
tree.create_node("Mary", "mary", parent="diane")
tree.create_node("Mark", "mark", parent="jane")
print(tree)
```

```
Harry
├── Bill
└── Jane
    ├── Diane
    │   └── Mary
    └── Mark
```

Alphabet

this is the index page for alphabet

add table of contents

tone

Intonation is the rise and fall of the voice in speaking. It is the accuracy of pitch in playing or singing, or on stringed instrument such as a tamtam.

We distinguish two categories of tones: elevated and elongated. Elevation of tone is either low or high ;its elongation is either descending or ascending.

As for types of elevated tones: their combination corresponds to a wave. the same applies to elongation.

The decibel of the tones corresponds to its magnitude on a wave.

its vertical magnitude corresponds to its height and the horizontal corresponds to its width. that is to say that elongation corresponds to width of the wave and elevation to the its heigh. therefore any other tone is a combination of the proportions of the wave height and width. please see mathematical explanation for further details.

Mandenkan distinguishes four tones: high and low, ascending and descenting. From these four are all others tones produced

Tone and **intonation** are both aspects of how we use pitch in speech, but they serve different purposes and are used in different linguistic contexts. Here's a breakdown of each:

Tone

- **Definition:** Tone refers to the use of pitch to distinguish word meaning.
- **Usage:** In **tonal languages** (e.g., Mandarin, Thai, Yoruba), tone is an essential part of a word's identity. The same sequence of sounds can have different meanings depending on the pitch or pitch pattern used.
- **Example:** In Mandarin, the syllable "ma" can mean different things depending on the tone used:
 - **mā** (high level tone) = "mother"
 - **má** (rising tone) = "hemp"
 - **mǎ** (falling-rising tone) = "horse"
 - **mà** (falling tone) = "scold"

Intonation

- **Definition:** Intonation refers to the variation of pitch across a sentence or phrase, used to convey meaning or emotion, or to signal the type of sentence (e.g., question, statement, command).
- **Usage:** Intonation occurs in **non-tonal languages** (e.g., English) and can change the nuance or intention behind a sentence, but it doesn't change the literal meaning of individual words.
- **Example:** In English, you can say the same words with different intonations:
 - "You're coming." (statement, falling intonation)
 - "You're coming?" (question, rising intonation)
 - "You're *coming*!" (emphasis, expressive intonation)

Key Differences:

- **Tone** affects individual words and their meanings.
- **Intonation** affects the overall meaning or mood of a sentence, without changing the dictionary meaning of the words.

Tonal languages have both **tone** and **intonation**, while non-tonal languages like English rely primarily on **intonation**.

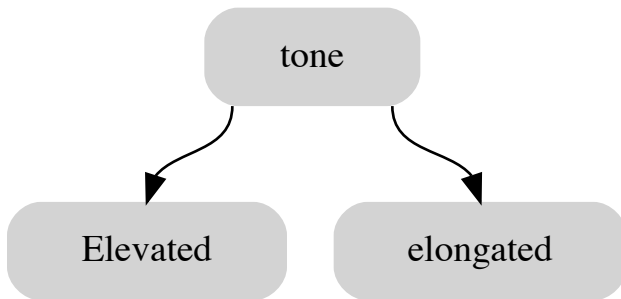
alphabet's tone in mathematics

```
# imports
from binarytree import Node
```

types of tone:

We distinguish two types of tone: Elevated and elongated

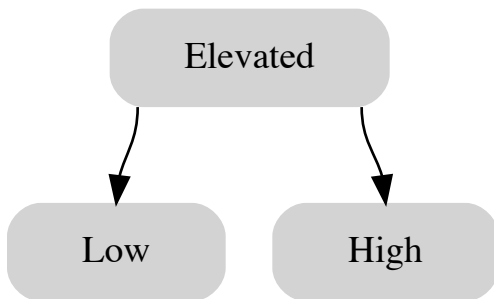

```
tone = Node('tone')
elevated = tone.left = Node('Elevated')
elongated = tone.right = Node('elongated')
tone.graphviz()
```



Elevated tone

Elevated tone is divided into two categories: low and high

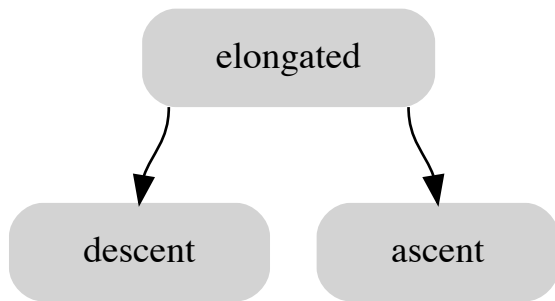
```
elevated = Node('Elevated')
left = elevated.left = Node('Low')
right = elevated.right = Node('High')
elevated.graphviz()
```



Elongated tone

Elongated tone is divided into two categories: Descent and Ascent

```
elongated = Node('elongated')
descent = elongated.left = Node('descent')
ascent = elongated.right = Node('ascent')
elongated.graphviz()
```



```
### intonation
```

Intonation corresponds to the combination of of elevated and elongated tones. it is a combination of low and high elevated tones and descending and ascending elongated tones. it corresponds to a wave of variable frequencies.

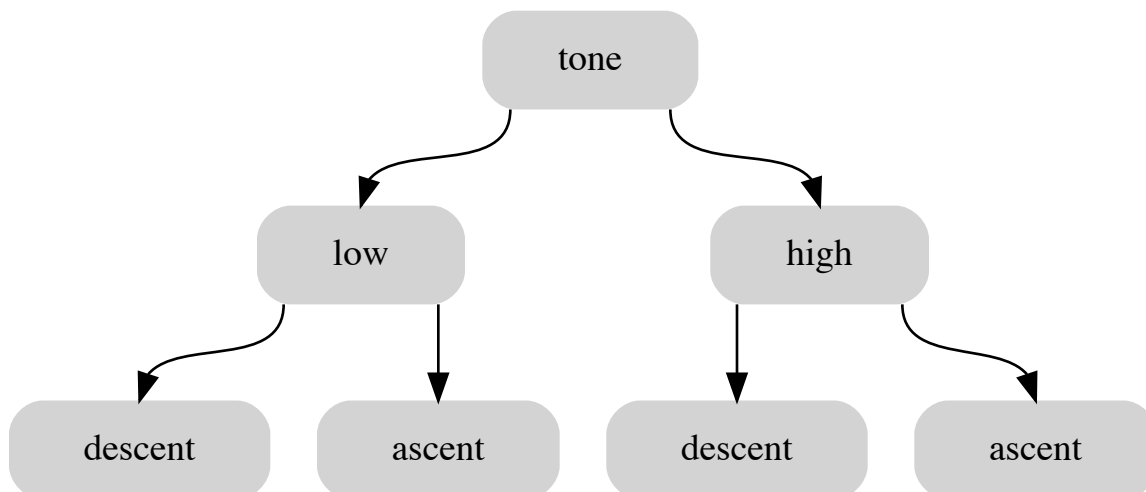
```

tone = Node('tone')
tone.left = Node('low')
tone.right = Node('high')

tone.left.left = Node('descent')
tone.left.right = Node('ascent')

tone.right.left = Node('descent')
tone.right.right = Node('ascent')

tone.graphviz()
  
```



intonation is the fall and rise of the voice in speaking.

```
$ y = a + b x $
```

```
Cell In[7], line 1
    $ y = a + b x $
    ^
SyntaxError: invalid syntax
```

alphabet's tone in physics

alphabet's tone chemistry

Probability & Statistics

```
from itertools import permutations
from math import factorial
```

```
per = permutations('ABCD', 3)
for val in per:
    print(*val)
```

A B C
A B D
A C B
A C D
A D B
A D C
B A C
B A D
B C A
B C D
B D A
B D C
C A B
C A D
C B A
C B D
C D A
C D B
D A B
D A C
D B A
D B C
D C A
D C B

```
import scipy
import numpy as np
from scipy.special import perm

k = np.array([3, 4])
n = np.array([10, 10])
perm(n, k)
```

array([720., 5040.])

```
perm(6, 2, exact=True)
```

30

```
from sympy import *
x, y, z = symbols("x y z")
```

```
expr = cos(x) + 1
expr.subs(x, y)
```

$$\cos(\left(y\right)) + 1$$

```
# Permutation without repetition
```

The permutations without repetition of tones high, low and wavy along with elongated, descending, and ascending gives using formula.

$y = a + b \times x$

$${}_n P_r = \frac{n!}{(n-r)!}$$

```
import scipy
import numpy as np
from scipy.special import perm

perm(3, 2, exact = True)
```

6

This results in six unique intonations.

```
Cell In[9], line 1
    This results in six unique intonations.
          ^
SyntaxError: invalid syntax
```

The permutation with repetition gives:

```
import scipy
import numpy as np
from scipy.special import perm
perm(2, 3)
```

0.0

The combination gives

```
from scipy.special import comb
my_comb = comb(6, 2, exact = True)
print("The combination of 6 and 2 is ")
print(my_comb)
```

```
The combination of 6 and 2 is
15
```

alphabet's tone in computer science

letter

Letter is the grain of speech, the root. it consists of vowels, nasalizers and numbers. the consonant is not a letter for it consists of one, two, or three successive letters. in nko script consonants are the basis of grammar.

Similar to hebrew and arabic, a letter is always capitalized: upper and lower cases are absent. in calligraphy all are of uniform dimension but this uniformity is irrelevant in penmanship.

The vowel is a letter utterable on a continuous note on the vocal track. it is produced with open mouth, allowing air to freely flow outwardly with minimal constriction of the vocal chords. they have the following properties: continuous ascent and descent, elongation and nasalization. they can be voiced in either high or low tone.

high and short tones are the tones in nko. neutral, intermediate, tone is a combination, a sinusoidal wave.

vowel

this is the index page for vowel

index

this is the index page for consonant

Number

this is the index page for number

Character ௐ

this is the index page for ௐ

Character T

this is the index page for the character T

Grammar

this is the index page for grammar

add table of contents

Noun

The noun, ௐ, is the foundation of grammar for it is the designation of every partical of speech. by it, do we distinguish between people and elements of nature.

Kante distinguishes two types of nouns: The Proper, ௐ, and Common noun, ௐ.

The pronounciation of a name is either known or unknow. We say that it is proper if know, ௐ, and improper if unknow, ௐ.

The final tone is serene, LHL , if proper, and is either abrupt, LHF , or serene if improper.

The Name is a proper noun: that is to say, it is know. The pronoun is an improper noun for it unknow.

pronoun

this is the index page for pronoun

adjective

this is the index page for adjective

table of content

this is the page for intermediate adjective

attachable

this is the page for attachable adjective

unattachable

this is the page for unattachable adjective

intermediate

this is the page for intermediate adjective

predicate

this is the page for predicate

verb

this is the page for verb

particle

this is the page for particle

interrogation

this is the page for interrogation

adverb

this is the page for adverb