

a music composition language

FARM 2016 (Nara, Japan)

<https://github.com/alda-lang/alda>

Dave Yarwood • @dave_yarwood

**First, some
background.**

↙ (that's me!)



File Home Note Input Notations Text Play Layout Appearance Parts Review View

Clef Key Signature Time Signature Barline Common

Slur Crescendo Decrescendo Trill

Segno Coda Do...py Inv...ent

Type Add Note Names

To and From Rests Over Rests Stemlets

Graphic Adjust Color

Bracket Brace Sub-bracket

Ideas

major

Score Library All

Rock Organ Rhythm 4 4/4 120bpi

Rock Organ Rhythm 5 4/4 120bpi

Rock Organ Rhythm 1 4/4 120bpi

Rock Organ Rhythm 2 4/4 120bpi

Rock Organ Rhythm 3 4/4 120bpi

Rock Electric Guitar Distorted Solo 9 4/4 120bpi

Rock Electric Guitar Distorted Solo 8 4/4 120bpi

Rock Electric Guitar Distorted Solo 5 4/4 120bpi

Rock Electric Guitar Distorted Solo 7 4/4 120bpi

Full Score

Playback

00:00'00.0" 1 100

Navigator

2

Fretboard

Auto Maple Guitar

Mixer

CPU

MASTER

Tpt. Tbn. E. Gtr. E. Pro. (a) E. Pro. (b) Bass Dr. Tamb. Vln. 1 Vln. 2 Vla. Vc. Click DLSMcd DLSMcd FX 1 FX 2 FX 3 FX 4

Keyboard

Electric Stage Piano

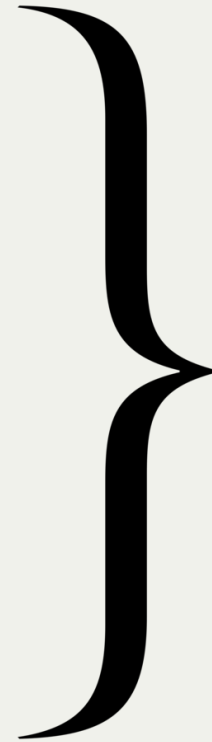
Page 1 of 6 Bars: 25 Bar 3 (4), beat 4 to bar 4 (5), beat 5 Timecode: 7.2"-10.1" Notes: F#4, A4, D5 Harmony: Db13/E Edit Passage Concert Pitch

125.00%



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>HTML</title>
</head>
<body>
  <div class="sort-of-important">
    <p>I'm a paragraph!</p>
  </div>
</body>
</html>
```

```
(defun fibonacci (N)
  "Compute the N'th Fibonacci number."
  (if (or (zerop N) (= N 1))
      1
      (+ (fibonacci (- N 1)) (fibonacci (- N 2))))))
```



abc

```
X:1
T:The Legacy Jig
M:6/8
L:1/8
R:jig
K:G
GFG BAB | gfg gab | GFG BAB | d2A AFD |
GFG BAB | gfg gab | age edB | 1 dBA AFD :|2 dBA ABd | :
efe edB | dBA ABd | efe edB | gdB ABd |
efe edB | d2d def | gfe edB | 1 dBA ABd :|2 dBA AFD | ]
```

LilyPond

```
\score {
  \new Staff {
    \key d \major
    \numericTimeSignature
    \time 2/4
    <cs' d'' b''>16 <cs' d'' b''>8.
    ~
    <cs' d'' b''>8 [ <b d'' a''> ]
  }
}
```

Music Macro Language (MML)

```
#timebase 480
#title "The M.GAKKOU KOUKA"
#copyright "Music Composed by Kenkichi Motoi 2009"
A t160
A o3l4 V12
A @1 ed8ce8 gg8er8 aa8>c<a8 g2r
A aa8ga8 >cc8d<r8 ee8de8 c2r
A dd8dd8 dd8dr8 ed8ef8 g2r
A aa8ga8 >cc8<ar8 >dc8de8 d2<r
A >ee8dc8< ab8>cc8< gg8ea8 g2r
A >cc8<ge8 cd8ea8 gg8de8 c2r
```

Clojure

clojure.org



```
(defn fibonacci
  "Compute the nth Fibonacci number."
  [n]
  (if (or (zero? n) (= n 1))
      1
      (apply + (map fibonacci [(- n 1) (- n 2)]))))
```

```
(let [counter (atom 0)]
  (dotimes [_ 10]
    (future (swap! counter inc)))
  (Thread/sleep 500)
  @counter)
```

```
(import '[javax.sound.midi MidiSystem])

(let [synth (doto (MidiSystem/getSynthesizer) .open
                  channel (aget (.getChannels synth) 0))]
  (doseq [note [36 38 40 41 43 45 47 48]]
    (.noteOn channel note 127)
    (Thread/sleep 300)
    (.noteOff channel note)))
```


Let the demo begin!

