मराठी

निरंजन

आवृत्ती १.२ — १६ जून, २०२०

https://gitlab.com/niranjanvikastambe/marathi

सारांश

XƎLaTeX चा वापर करून मराठीत अक्षरजुळणी[१] शक्य आहे, परंतु सुलभ नाही. LaTeX सह मराठीत अक्षरजुळणी करताना आपली बीजधारिका अनेक आज्ञांनी भरून जाते. हा आज्ञासंच अशा सर्व उपयुक्त आज्ञा आधीच लिहून ठेवतो. त्यामुळे नव्या वापरकर्त्यांना त्या नव्याने शिकाव्या लागत नाहीत. शिवाय blindtext ह्या आज्ञासंचाप्रमाणे मराठीकरिता नमुना मजकूर उत्पन्न करण्यासाठीची सोय ह्या आज्ञासंचात करण्यात आली आहे. LuaLaTeX हा अत्याधुनिक चालक वापरल्यास अंक रोमी लिपीत येतात, ती अडचणदेखील ह्या आज्ञासंचात सोडवण्यात आली आहे.

अनुक्रमणिका

१ प्रस्तावना २

२ आज्ञासंचाची घडण ४

२.१ अंतर ५

२.२ standalone लाटेक्-वर्ग ६

२.३ लुआ-लाटेक् ६

२.४ झी-लाटेक् व टंक ७

२.५ नमुना मजकूर ९

२.५.१ परिच्छेद ९

२.५.२ दस्तऐवज ९

[१]ह्या दस्तऐवजात लाटेक्-मध्ये प्रचलित असणाऱ्या अनेक इंग्रजी संज्ञांसाठी पर्यायी मराठी संज्ञा वापरण्यात आल्या आहेत, हे सर्व प्रयोग तुलनेने नवे असल्याने ह्याच दस्तऐवजात शेवटी दिलेल्या संज्ञासूचीत सर्व संज्ञांचे इंग्रजी अर्थ दिले आहेत.

१ प्रस्तावना

लाटेक्-मध्ये मराठीचा वापर करताना सर्वात महत्त्वाची गरज म्हणजे देवनागरी लिपी नीट दिसणे. लाटेक् विकसित झाले तेव्हा युनिकोड ही प्रणाली अस्तित्वात नसल्याने टेक्, लाटेक् ह्यांसारख्या चालकांसह युनिकोड अक्षरे वापरता येत नाहीत, परंतु लवकरच युनिकोड अक्षरांचा वापर लाटेक्-मध्ये करता यावा ह्याकरिता XeLaTeX (झी-लाटेक्) तसेच LuaLaTeX (लुआ-लाटेक्) ह्या नव्या चालकांचा विकास झाला. fontspec सदृश आज्ञासंचासह एखादा युनिकोड-आधारित टंक वापरणे व युनिकोड-मजकूर थेट झी-लाटेक् अथवा लुआ-लाटेक्-सोबत चालवणे हे ह्या नव्या चालकांचे प्रमुख उद्दिष्ट होते, परंतु हे काम अतिशय गुंतागुंतीचे होते. २०२० चे टेक्-वितरण येईपर्यंत व त्यात हर्फ़बझ नावाची नवी आज्ञावली येईपर्यंत लुआ-लाटेक् देवनागरी हाताळू शकलेच नाही. झी-लाटेक्-सह मात्र देवनागरी व्यवस्थित दिसणे शक्य होत होते.

देवनागरी दिसण्यासाठी टाकावयाच्या आज्ञा

देवनागरी योग्य तऱ्हेने दिसण्यासाठी काही आज्ञांचा वापर करणे अनिवार्य होते. देवनागरी लिपी दस्तऐवजात दाखवण्यासाठी लागणाऱ्या किमान आज्ञा पुढीलप्रमाणे.

```
% !TEX TS-program = xelatex
\documentclass{article}
\usepackage{fontspec}
\setmainfont{Shobhika} % अथवा कोणताही युनिकोड-आधारित देवनागरी टंक

\begin{document}
 नमस्कार
\end{document}
```

ह्या उदाहरणाने देवनागरी दिसत असले तरी फलित मात्र हवे तसे दिसत नाही. जोडाक्षरे तुटक दिसतात (उदा. नमस्‌कार). त्यासाठी \setmainfont ह्या आज्ञेस Script=Devanagari असे प्राचल द्यावे लागते. पुढील अडचण म्हणजे लाटेक् आपोआप पुरवणारे आकडे (उदा. पृष्ठक्रमांक, तळटिपांचे क्रमांक) देवनागरीत न येणे. त्याकरिता ह्याच आज्ञेस Mapping=devanagarinumerals असे आणखी एक प्राचल द्यावे लागते. इतके करूनही भाषेचा प्रश्न उरतोच! उदा. लाटेक्-ला इंग्रजीव्यतिरिक्त इतर कोणतीही भाषा कळत नसल्यामुळे मूलभूत इंग्रजी शब्दांची भाषांतरे पुरवणारे बेबल अथवा पॉलिग्लॉसिया ह्यांसारखे आज्ञासंच वापरून भाषा निवडावी लागते. रोहित होळकरांच्या latex-mr ह्या पुस्तिकेत ह्या सर्व अडचणींची तपशीलवार चर्चा झाली आहे.

सद्यपरिस्थितीत लाटेक्-चे किमान ज्ञान असलेल्या नव्या वापरकर्त्याला मराठी लिहिण्यासाठी एवढा सगळा प्रपंच करायला लावणे म्हणजे ज्या फांदीवर आपण बसलो आहोत तीच

तोडण्यासारखे आहे. त्यामुळे कोणत्याही वापरकर्त्याला केवळ लाटेक्-च्या किमान ज्ञानासह \usepackage{marathi} एवढी एक आज्ञा लिहून उत्तम देवनागरी अक्षरजुळणी करता यावी हा ह्या आज्ञासंचाचा उद्देश आहे. ह्या आज्ञासंचात पुढील आज्ञांचा समावेश आहे.

\परिच्छेद ही आज्ञा केवळ एक नमुना परिच्छेद निर्माण करते. ही आज्ञा एकामागोमाग एक अनेकदा टाकल्याने तितके परिच्छेद निर्माण केले जाऊ शकतात.

\नमुना नमुना ही आज्ञा सर्वप्रथम दिलेला लाटेक्-वर्ग पाहते व त्यानुसार त्या वर्गाच्या किमान क्षमता दर्शवणारी एक फलित-धारिका निर्माण करते. उदाहरणादाखल article, book, report, beamer व letter ह्या लाटेक्-वर्गांसह \नमुना ही आज्ञा चालवून पाहा. एकाच बीजधारिकेत केवळ लाटेक्-वर्ग बदलत असाल, तर लाटेक्-ने तयार केलेल्या साहाय्यक धारिका काही वेळा अनपेक्षित अडचण दाखवतात. जुनी माहिती शिल्लक असल्याने त्या दिल्या जातात. घाबरून न जाता, बीजधारिका दोनदा चालवावी. सर्व अडचणी सुटतात.

\टंक \टंक {⟨टंकाचे नाव⟩}

ह्या आज्ञासंचात शोभिका हा मूलटंक म्हणून निवडून ठेवला आहे. तो बदलायचा असेल तर \टंक ह्या आज्ञेची सोय करण्यात आली आहे. ह्या आज्ञेसह आपोआप देवनागरी टंकांसाठी आवश्यक असणारी Renderer=Harfbuzz, Script=Devanagari (लुआ-लाटेक्), Script=Devanagari, Mapping=devanagarinumerals (झी-लाटेक्) ही प्राचले लिहून ठेवली आहेत. शिवाय \setmainfont{टंकाचे नाव} ही आज्ञा नेहमीप्रमाणे चालतेच. टंकाचे नाव हा \टंक ह्या आज्ञेचा कार्यघटक आहे.

\दुसराटंक \दुसराटंक{⟨टंकासाठीची आज्ञा⟩}{⟨टंकाचे नाव⟩}

दस्तऐवजात मूलटंकाखेरीज आणखी एखादा टंक वापरण्याची गरजदेखील पडू शकते. त्याकरिता \दुसराटंक आज्ञा पुढीलप्रमाणे वापरावी.

```
\documentclass{article}
\usepackage{marathi}
\दुसराटंक{\नवाटंक}{Mukta}

\begin{document}
नमस्कार, हा मजकूर शोभिका टंकात छापला गेला आहे, परंतु {\नवाटंक हा मजकूर मुक्त टंकात छापला गेला आहे.}
\end{document}
```

अंतर

हे प्राचल वापरल्यास आज्ञासंचातर्फे दोन ओळींमधील अंतरात कोणताही हस्तक्षेप केला जात नाही. अधिक माहितीसाठी २.१ वाचा. ह्या प्राचलाला किंमतदेखील देता येते. उदा. \usepackage[अंतर=2]{marathi} अशा प्रकारे आज्ञासंच वापरल्यास ओळींमधले अंतर दुप्पट होते. कोणत्याही प्राचलाशिवाय वापरल्यास आज्ञासंचातर्फे मूलभूत अंतराच्या दीडपट अंतर पुरवले जाते. पुढील कोष्टकाने अंतर ह्या प्राचलाचा वापर अधिक स्पष्ट होईल.

आज्ञासंचा वापर	ओळींमधले अंतर
\usepackage{marathi}	मूळ अंतराच्या दीडपट
\usepackage[अंतर]{marathi}	मूळ अंतरात कोणताही फरक नाही.
\usepackage[अंतर=2]{marathi}	मूळ अंतराच्या दुप्पट

कोष्टक १: अंतर

\अंतरबदल

\अंतरबदल{⟨किंमत⟩}

दस्तऐवजात कुठेही अंतर बदलावयाचे असल्यास ह्या आज्ञेने ते बदलता येते. ह्या आज्ञेला एक कार्यघटक आहे. त्यात ओळींमधले मूळ अंतर जितक्या पटींनी बदलायचे आहे ती किंमत टाकावी. पुढील उदाहरण पाहा.

```
\documentclass{article}
\usepackage{marathi}

\begin{document}
\परिच्छेद
\अंतरबदल{5}
\परिच्छेद
\end{document}
```

२ आज्ञासंचाची घडण

आता आपण आज्ञासंचाची घडण व त्यातील आज्ञांचा उपयोग लक्षात घेऊयात.

```
⟨@@=marathi⟩
⟨*package⟩
```

```
\ProvidesPackage{marathi}
\NeedsTeXFormat{LaTeX2e}
```

ह्या आज्ञांसह आज्ञासंचाची पायाभूत माहिती पुरवली.

२.१ अंतर

इंग्रजीमधली g, j, y अशी अक्षरे सोडली तर ह्याहून जास्त खोल असणारा मजकूर त्या लिपीत आढळत नाही. देवनागरीचे तसे नाही. क ह्या अक्षराहून क्क थोडे अधिक खोल. ट्ट त्याहून थोडे अधिक व ट्टू त्याहून. अशा असमान उंचीच्या अक्षरांमुळे लाटेक् आपोआप दोन ओळींमधले अंतर बदलते व त्यामुळे ओळींची उंची असमान दिसू लागते. साध्या मजकुरातील ओळींप्रमाणेच कोष्टकेदेखील कुरूप दिसू लागतात. ह्यावर तोडगा काय? ह्याची दोन उत्तरे आहेत. पहिले उत्तर मराठीच्या आजवरच्या छपाईच्या इतिहासाचा अभ्यास करून, मराठी लिहिताना दोन ओळींमध्ये सोडले गेलेले सरासरी अंतर किती ह्याचे संशोधन करणे व त्यानुसार सर्व ठिकाणी ते अंतर लागू करणे. हा मार्ग सहज नाही. हा संशोधनाचा विषय आहे. त्याकरिता टंकाच्या आकाराचे व त्यानुसार बदलणाऱ्या ओळींच्या अंतराचे प्रमाणदेखील काळजीपूर्वक अभ्यासावे लागेल. हे सर्व करणे जरी इष्ट व आवश्यक असले, तरी हे संशोधन पूर्ण होईस्तोवर सामान्य वापरकर्त्याला ज्या आज्ञा शिकण्याचे कष्ट पडतात ते कसे टाळावेत? त्यासाठी तात्पुरता तोडगा काढावा लागतो. तो असा की ओळींमधले अंतर दस्तऐवजाच्या सुरुवातीलाच वाढवून ठेवायचे. त्यासाठी पुढील आज्ञा वापरल्या जातात. ओळींमधल्या अंतरासाठी setspace हा आज्ञासंच वापरावा असा सल्ला जोनाथन स्प्राट ह्यांनी टॉप-आन्सर्स ह्या संकेतस्थळावर दिला. त्यानुसार हा बदल करत आहे.

आमच्या काही सहकाऱ्यांच्या मते हा निर्णय अतिशय उग्र आहे. त्यामुळे दस्तऐवजातील इतर काही ठिकाणच्या अंतरांवर सूक्ष्म परिणाम घडतात, जे टाळणे अधिक इष्ट. त्यांच्या ह्या मताचा विचार करून ह्या आज्ञासंचाकरिता **अंतर** नावाचे प्राचल विकसित केले आहे. त्याचा वापर कसा केला जावा ही माहिती आपण १ मध्ये वाचलीच. हे प्राचल वापरल्यास आज्ञासंचातर्फे ओळींमधल्या अंतरात कोणताही हस्तक्षेप केला जात नाही. त्याकरिता पुढील आज्ञा समाविष्ट केल्या आहेत. ह्या आज्ञांसाठी जूल वोऊ ह्यांच्या टॉप-आन्सर्सवरील ह्या उत्तराची मदत झाली.

```
\RequirePackage{setspace}
\RequirePackage{pgfkeys}

\def\बदल#1{\pgfkeys{marathi/.cd,#1}}
\pgfkeys{
  marathi/.is family,marathi/.cd,
  अंतर/.code={\setstretch{#1}\selectfont},
  अंतर=1.5,
```

```
  अंतर/.default=1
}

\DeclareOption*{\expandafter\बदल\expandafter{\CurrentOption}}
\ProcessOptions
\providecommand{\अंतरबदल}[1]{\बदल{अंतर=#1}}
\def\arraystretch{1.2}
```

२.२ standalone लाटेक्-वर्ग

standalone हा विशेष लाटेक्-वर्ग केवळ दस्तऐवजात दिलेल्या गोष्टींच्या आकाराचे फलित तयार करण्यासाठी वापरला जातो. उदा. article लाटेक्-वर्गात एखादे कोष्टक टाकले तर पानाच्या नेहमीच्या आकाराइतकी जागा कोष्टकाच्या आजूबाजूला सुटतेच. standalone लाटेक्-वर्गात मात्र तसे न होता केवळ कोष्टकाच्या आकाराइतके फलित निर्माण होते, परंतु ह्या वर्गास polyglossia आज्ञासंच व त्यातून पुरवली जाणारी भाषांतरे अनावश्यक आहेत व म्हणून अडचणी निर्माण होतात. त्यामुळे ह्या लाटेक्-वर्गाकरिता केवळ fontspec हा आज्ञासंच वापरून, इतर सर्व वर्गांसाठी polyglossia हा आज्ञासंच वापरला आहे. त्यामुळे सर्व लाटेक्-वर्गांमध्ये भाषांतरेही मिळतात व standalone वर्गातदेखील आज्ञासंचामुळे कोणतीही अडचण येत नाही. पुढील आज्ञांनी हे साधले आहे.

```
\@ifclassloaded{standalone}{\RequirePackage{fontspec}}{}
```

२.३ लुआ-लाटेक्

लुआ-लाटेक् हा अत्याधुनिक चालक आता देवनागरीसाठी उपलब्ध झाला आहे. हर्फ़बझ़ ह्या लुआविशिष्ट आज्ञावलीच्या मदतीने देवनागरी व्यवस्थित दाखवली जात आहे. अशा वेळी तिचा पुरेपुर उपयोग का करू नये? लुआ-लाटेक् polyglossia पेक्षा babel सह चांगले फलित देते. त्यामुळे बेबलविशिष्ट आज्ञा पुढीलप्रमाणे पुरवल्या आहेत.

```
\RequirePackage{iftex}

\ifluatex

\usepackage{babel}
\babelprovide[import, main, maparabic, mapdigits,
counters/swar = अ आ इ ई उ ऊ ए ऐ ओ औ अं अः ॲ ऋ ऌ ऑ ,
alph=swar,
```

```
counters/anka = एक दोन तीन चार पाच सहा सात आठ नऊ दहा अकरा बारा
तेरा चौदा पंधरा सोळा सतरा अठरा एकोणीस वीस एकवीस बावीस तेवीस
चोवीस पंचवीस सव्वीस सत्तावीस अठ्ठावीस एकोणतीस तीस एकतीस बत्तीस
तेहतीस चौतीस पस्तीस छत्तीस सदतीस अडतीस एकोणचाळीस चाळीस
एकेचाळीस बेचाळीस त्रेचाळीस चव्वेचाळीस पंचेचाळीस शेहेचाळीस सत्तेचाळीस
अठ्ठेचाळीस एकोणपन्नास पन्नास एकावन्न बावन्न त्रेपन्न चौपन्न पंचावन्न छप्पन्न
सत्तावन्न अठ्ठावन्न एकोणसाठ साठ एकसष्ट बासष्ट त्रेसष्ट चौसष्ट पासष्ट सहासष्ट
सदुष्ट अडुसष्ट एकोणसत्तर सत्तर एकाहत्तर बाहत्तर त्र्याहत्तर चौऱ्याहत्तर पंचाहत्तर
शाहत्तर सत्त्याहत्तर अठ्ठ्याहत्तर एकोणऐंशी ऐंशी एक्याऐंशी ब्याऐंशी त्र्याऐंशी
चौऱ्याऐंशी पंच्याऐंशी श्याऐंशी सत्त्याऐंशी अठ्ठ्याऐंशी एकोणनव्वद नव्वद एक्याण्णव
ब्याण्णव त्र्याण्णव चौऱ्याण्णव पंचाण्णव शहाण्णव सत्त्याण्णव अठ्याण्णव नव्याण्णव शंभर,
Alph=anka,
counters/vyanjan = क ख ग घ ङ
च छ ज झ ञ
ट ठ ड ढ ण
त थ द ध न
प फ ब भ म
य र ल व श ष स ह ळ
]{marathi}

\renewcommand\thepart{\localecounter{anka}{part}}
\renewcommand\theenumiii{\localecounter{vyanjan}{enumiii}}
```

२.४ झी-लाटेक् व टंक

शोभिका हा लाटेक्-वितरणासह येणारा व देवनागरीची अतिशय चांगली अक्षरजुळणी करणारा टंक आहे. तो मूलटंक म्हणून ह्या आज्ञासंचाद्वारे निवडला जातो. अर्थात तो बदलण्याच्या सुविधेसकट. शिवाय \दुसराटंक ह्या आज्ञेमार्फत तात्पुरता टंक निवडता येतो. शिवाय कुठल्याही देवनागरी टंकाचे योग्य फलित दिसण्यासाठी Script=Devanagari हे प्राचल वापरावे लागते व लुआ-लाटेक्-मध्ये ह्यासह हर्फ़बझ ही आज्ञावली वापरावी लागते. \टंक ह्या आज्ञेद्वारे निवडल्या जाणाऱ्या टंकासदेखील ती प्राचले वापरणे हे पुढील आज्ञांनी साधले जाते. झी-लाटेक्-सह polyglossia वापरणं हाच चांगला पर्याय आहे. त्यामुळे तशी निवड करून ठेवली आहे.

```
\usepackage{fontspec}
\defaultfontfeatures{Renderer=HarfBuzz}
\babelfont{rm}{Shobhika}

```

```
\else

\RequirePackage{polyglossia}
\setdefaultlanguage{marathi}
\setmainfont[Script=Devanagari,Mapping=devanagarinumerals]{Shobhika}
\fi

\providecommand{\टंक}[1]
{\ifluatex
\babelfont{rm}{#1}

\else

\setmainfont[
Script=Devanagari,
Mapping=devanagarinumerals
]
{#1}
\fi
}

\providecommand{\दुसराटंक}[2]{
\ifluatex
\newfontfamily{#1}[Renderer=Harfbuzz,Script=Devanagari]{#2}
\else
\ifxetex
\newfontfamily{#1}[Script=Devanagari,Mapping=devanagarinumerals]{#2}
\fi\fi
}
```

ह्या आज्ञांमुळे धारिका लुआ-लाटेक् अथवा झी-लाटेक् ह्यांपैकी कोणत्याही चालकासह चालवता येते.

२.५ नमुना मजकूर

२.५.१ परिच्छेद

परिच्छेद ही आज्ञा दस्तऐवजात कुठेही वापरली तरी एक लहानसा परिच्छेद आपोआप छापला जातो. त्याकरिता एक मजकूर धारिका आज्ञासंचासोबत येते. ती आज्ञासंचात पुढील आज्ञांनी समाविष्ट करून घेतली आहे.

```
\providecommand{\परिच्छेद}{\input{namuna-para}}
```

२.५.२ दस्तऐवज

मागे म्हटल्याप्रमाणे नमुना मजकूर तयार करण्यासाठी ह्या आज्ञासंचाचा वापर करता येतो, परंतु त्याकरिता लाटेक्-ला थोडी माहिती पुरवावी लागते. उदा. दस्तऐवजाचा/ची लेखक/लेखिका, दस्तऐवजाचं शीर्षक इत्यादी. ही माहिती पुरवण्याचे विशिष्ट स्थान आहे. लाटेक्-मध्ये मूळ दस्तऐवज सुरू होण्यापूर्वी ही माहिती पुरविण्याकरिता आज्ञापीठ असते, तिथे ही माहिती पुरवली जाते, परंतु ह्यामुळे फलित-धारिकेच्या पायाभूत माहितीत ती नावे दिसू लागतात. ह्यासाठी आज्ञासंचात ही माहिती पुरवली गेली नाही आहे, ह्याउलट सोबत जोडलेल्या वेगवेगळ्या दस्तऐवजांमध्ये ती माहिती पुरवली गेली आहे. त्या धारिका केवळ नमुना मजकूर पुरवण्यासाठी आहेत. निरनिराळ्या लाटेक्-वर्गांसाठी संबंधित धारिका निवडणे व दस्तऐवजात लिहिलेला लाटेक्-वर्ग कोणता आहे हे पाहून त्यानुसार नमुना मजकूर छापणे ह्यासाठीच्या आज्ञा पुढीलप्रमाणे.

```
\newcounter{क्र}
\@ifclassloaded{article}{\setcounter{क्र}{1}}{}
\@ifclassloaded{book}{\setcounter{क्र}{2}}{}
\@ifclassloaded{report}{\setcounter{क्र}{3}}{}
\@ifclassloaded{beamer}{\setcounter{क्र}{4}}{}
\@ifclassloaded{letter}{\setcounter{क्र}{5}}{}
\providecommand{\नमुना}{
  \ifnum\value{क्र}=1\input{namuna-article}\else
  \ifnum\value{क्र}=2\input{namuna-book}\else
  \ifnum\value{क्र}=3\input{namuna-report}\else
  \ifnum\value{क्र}=4\input{namuna-beamer}\else
  \input{namuna-letter}\relax\fi\fi\fi\fi
}
```

beamer लाटेक्-वर्ग वापरताना serif ही टंकछटा निवडावी लागते, त्याशिवाय देवनागरी लिपी दिसत नाही. त्या आज्ञा पुढीलप्रमाणे भरल्या आहेत. expex आज्ञासंचाचे स्थानिकीकरणदेखील

पुढील आज्ञांमध्ये समाविष्ट आहे.

```
\@ifclassloaded{beamer}{%
  \usefonttheme{serif}}

\@ifpackageloaded{expex}{
\definelabeltype{devanagari}
{labelgen=list,labellist={अ,आ,इ,ई,उ,ऊ,ए,ऐ,ओ,औ,अं,अः},
labelformat=A.,
fullrefformat=XA,
labelalign=left,
labelwidth=1.5em}

\lingset{labeltype=devanagari}
}{}
\endinput
⟨/package⟩
```

संज्ञासूची

अक्षरजुळणी Typesetting. १

बीजधारिका Source file. पर्यायी - स्रोत-धारिका. १

आज्ञासंच Package. १

आज्ञा Command. १

चालक Compiler. १

टंक Font. २

टेक्-वितरण TeX-distribution. २

आज्ञावली Program. २

प्राचल Parameter, option. २

लाटेक्-वर्ग Document class. ३

फलित-धारिका Output file. ३

साहाय्यक धारिका Auxiliary files. ३

अडचण Error. ३

मूलटंक Default font. ३

कार्यघटक Argument. ३

आज्ञापीठ Preamble. ४

पायाभूत माहिती Metadata. ४