

PHASE 01: GRAMMAR

Compiler Project

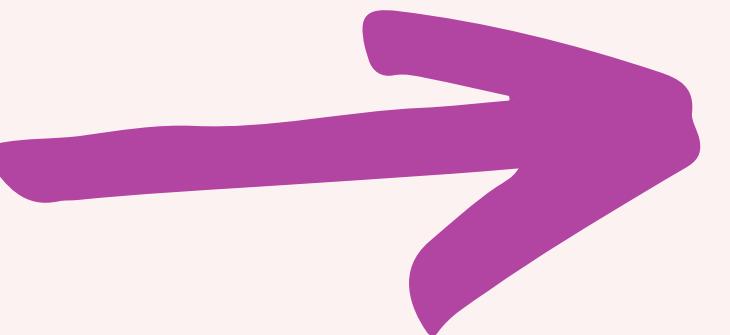
Mónica AYALA

Davide AVESANI

Théodore PREVOT

Formal Approaches, Languages and
Compilers

Markdown to HTML



HTML



What steps did we take?

1. Define markdown rules to be translated
2. Define an ‘individual’ grammar for each of these rules
3. Create the final grammar merging all the grammars previously defined

Markdown rules

- Paragraph
- Bold text
- Italic text
- Underlined text
- Strikethrough text
- Titles
- Links
- Images
- Quotes
- Inline code
- Block code
- Horizontal rule
- New Line
- Bold and Italic
- Lists
- URLs and Email Addresses
- Escaped Characters

Paragraph

Markdown

I really like using Markdown.

I think I'll use it to format all of my documents from now on.

HTML

```
<p> I really like using Markdown.
```

```
</p>
```

```
<p> I think I'll use it to format all of my documents from now on. </p>
```

Rendered

I really like using Markdown.

I think I'll use it to format all of my documents from now on.

Paragraph

G = (T, N, P, S)

T = {TEXT, <newline>}

N = {paragraph}

P = <paragraph>: (<title> | <list> | TEXT | <bold> | <italic> | <underline> |
<strikethrough> | <links> | <images> | <inline code> | <bold and italics> | <escape
sequence> | <url>)* <newline>

S = <paragraph>

Bold Text

Markdown

```
I just really love  
**bold text** it is  
truly so **good**
```

HTML

```
I just really  
love <strong>bold  
text</strong> it  
is truly so  
<strong> good  
</strong>
```

Rendered

```
I just really love  
bold text it is truly  
so good
```

Bold Text

G = (T, N, P, S)

T = {TEXT, *}

N = {<bold>}

P = <bold>: **TEXT**

S = <bold>

Italic Text

Markdown

```
I just really love  
*italicized text*  
it is truly so  
*good*
```

HTML

```
I just really  
love  
<em>italicized  
text</em> it is  
truly so <em>  
good </em>
```

Rendered

I just really love
italicized text it is
truly so *good*

Italic Text

G = (T, N, P, S)

T = {TEXT, *}

N = {<italic>}

P = <italic>: *TEXT*

S = <italic>

Underlined Text

Markdown

```
I just really love  
--underlined text--  
it is truly so  
--good--
```

HTML

```
I just really  
love  
<u>underlined  
text</u> it is  
truly so <u> good  
</u>
```

Rendered

I just really love
underlined text it is
truly so good

Underlined Text

G = (T, N, P, S)

T = {TEXT, _}

N = {<underlined>}

P = <underlined>: __TEXT__

S = <underlined>

Strikethrough Text

Markdown

```
I just really  
love  
~~strikethrough  
text~~ it is  
truly so ~~good  
~~
```

HTML

```
I just really love  
<s>strikethrough  
text</s> it is  
truly so <s>good  
</s>
```

Rendered

I just really love
~~strikethrough text~~
it is truly so ~~good~~

Strikethrough text

G = (T, N, P, S)

T = {TEXT, ~}

N = {<strikethrough>}

P = <strikethrough>: ~~TEXT~~

S = <strikethrough>

Titles

Markdown

```
# Heading 1
## Heading 2
### Heading 3
#### Heading 4
#####
##### Heading 5
#####
##### Heading 5
```

HTML

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

Rendered

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5
Heading 6

Titles

G = (T, N, P, S)

T = {TEXT, #}

N = { <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <title> }

P = <title>: <h1> | <h2> | <h3> | <h4> | <h5> | <h6> , <h1>: #

TEXT , <h2>: ## TEXT, <h3>: ### TEXT, <h4>: ##### TEXT,

<h5>: ##### TEXT, <h6>: ##### TEXT, <h1>: #TEXT

S = <title>

Images

Markdown

I like to add images
![example]
(<https://example.com/image.jpg>) to my document

HTML

I like to add images

```

```

to my document

Rendered

I like to add images



to my document

Images

G = (T, N, P, S)

T = {TEXT, [,], (,), !}

N = {<image>}

P = <image>: ![TEXT] (TEXT)

S = <image>

Quotes

Markdown

```
> This is a
paragraph inside
a blockquote,
doesn't it look
pretty?
```

HTML

```
<blockquote> <p>
This is a
paragraph inside
a blockquote,
doesn't it look
pretty? </p>
</blockquote>
```

Rendered



```
This is a paragraph
inside a
blockquote,
doesn't it look
pretty?
```

Quotes

G = (T, N, P, S)

T = { > }

N = {<block quote>, <paragraph>, <nested block quote>, <multiple block quote>, <single block quote> <new line> }

P = <nested block quote>: > <single block quote>

<multiple block quote>: ((<single block quote>|<nested block quote>) <new line> > <new line> (<single block quote>|<nested block quote>))+

<single block quote>: > <paragraph>,

<block quote>: <single quote> | <nested block quote> | <multiple block quote>

S = <block quote>

Inline Code

Markdown

At the command prompt, type `nano`.

HTML

At the command prompt, type `<code>nano</code>`.

Rendered

At the command prompt, type nano

Inline code

G = (T, N, P, S)

T = {TEXT, `}

N = {<code>}

P = <code>: ` TEXT `

S = <code>

Block code

Markdown

The for loop syntax in python is the following:

```

```
for num in numbers:
 print(num)
```

```

HTML

`<p>The for loop syntax in python is the following:</p>`

```
<pre><code>  
for num in numbers:  
    print(num)  
</code></pre>
```

Rendered

The for loop syntax in python is the following:

```
for num in numbers:  
    print(num)
```

Block code

G = (T, N, P, S)

T = {TEXT, ``}

N = {<block code>, <newline>}

P = <block code>: `` (TEXT | <newline>)* ``

S = <block code>

Horizontal rule

Markdown

I really like
using Horizontal
rules on my
document .

They make it
looks so fancy!

HTML

I really like
using Horizontal
rules on my
document .

`<hr>`

They make it
looks so fancy!

Rendered

I really like
using Horizontal
rules on my
document .

They make it
looks so fancy!

Horizontal rule

G = (T, N, P, S)

T = {*, _, -}

N = {<horizontal rule>, <newline>}

P = <horizontal rule>: <newline><newline> (***)+ | -+ |
____+) <newline><newline>

S = <horizontal rule>

Line Breaks

Markdown

Line breaks are essentials to properly format my document.

I use them to make my document more readable

HTML

Line breaks are essentials to properly format my document.

I use them to make my document more readable

Rendered

Line breaks are essentials to properly format my document.

I use them to make my document more readable

New Line

G = (T, N, P, S)

T = {\n}

N = {<new line>}

P = <new line>: \n

S = <new line>

Bold and Italics

Markdown

```
I just really
love ***bold and
italic*** it is
truly so
***good***
```

HTML

```
I just really
love <b><i>bold
and italic</i>
</b> it is truly
so <b><i>good</i>
</b>
```

Rendered

I just really
love ***bold and***
italic it is
truly so *good*

Bold and Italics

G = (T, N, P, S)

T = {TEXT, ***, ___}

N = {<bold italic>}

P = <bold italic>: ***TEXT*** | ___TEXT___

S = <bold italic>

Lists

Markdown

```
I like making  
lists.
```

- * First element
- * Second element
- * Third element

HTML

```
<p> I like making  
lists. </p>  
  
<ul>  
  <li> First  
element </li>  
  <li> Second  
element </li>  
  <li> Third  
element </li>  
</ul>
```

Rendered

I like making lists.

- First element
- Second element
- Third element

Lists

G = (T, N, P, S)

T = {*, _, TEXT}

N = {<list>}

P = <list>: (* TEXT<newline>)+ | (- TEXT<newline>)+

S = <list>

URLs and email addresses

Markdown

I like to put my contacts informations on my document !

My email:
<abc@example.com>

My website:
[pincopallino.com.mx]

HTML

I like to put my contacts informations on my document !

```
<a href = "mailto:  
abc@example.com">My  
Email</a>  
<a href =  
"pincopallino.com.mx">  
My Website</a>
```

Rendered

I like to put my contacts informations on my document !

My_email
My_website

URLs and email addresses

$G = (T, N, P, S)$

$T = \{TEXT, <, >, [,], (,)\}$

$N = \{<url>\}$

$P = <url>: <TEXT> \mid TEXT$

$S = <url>$

Scaped Characters

Markdown

Sometimes I want to use characters like \\ and \\< without suddenly activating markdown or html syntax \\!

HTML

```
<p>Sometimes I want to use characters like \\ and &lt; without suddenly activating markdown or html syntax !</p>
```

Rendered

Sometimes I want to use characters like \\ and < without suddenly activating markdown or html syntax !

Scaped Characters

G = (T, N, P, S)

T = {\, !, `_, *, {, }, (,), <, >, [,], #, +, -, ., |}

N = {<escape sequence>}

P = <escape sequence>: "\\" | "\!" | "\`" | "_" | "*" | "\{" | "\}"
| "\(" | "\)" | "\<" | "\>" | "\[" | "\]" | "\#" | "\+" | "\-" | "\." | "\|"

S = <escape sequence>

Markdown

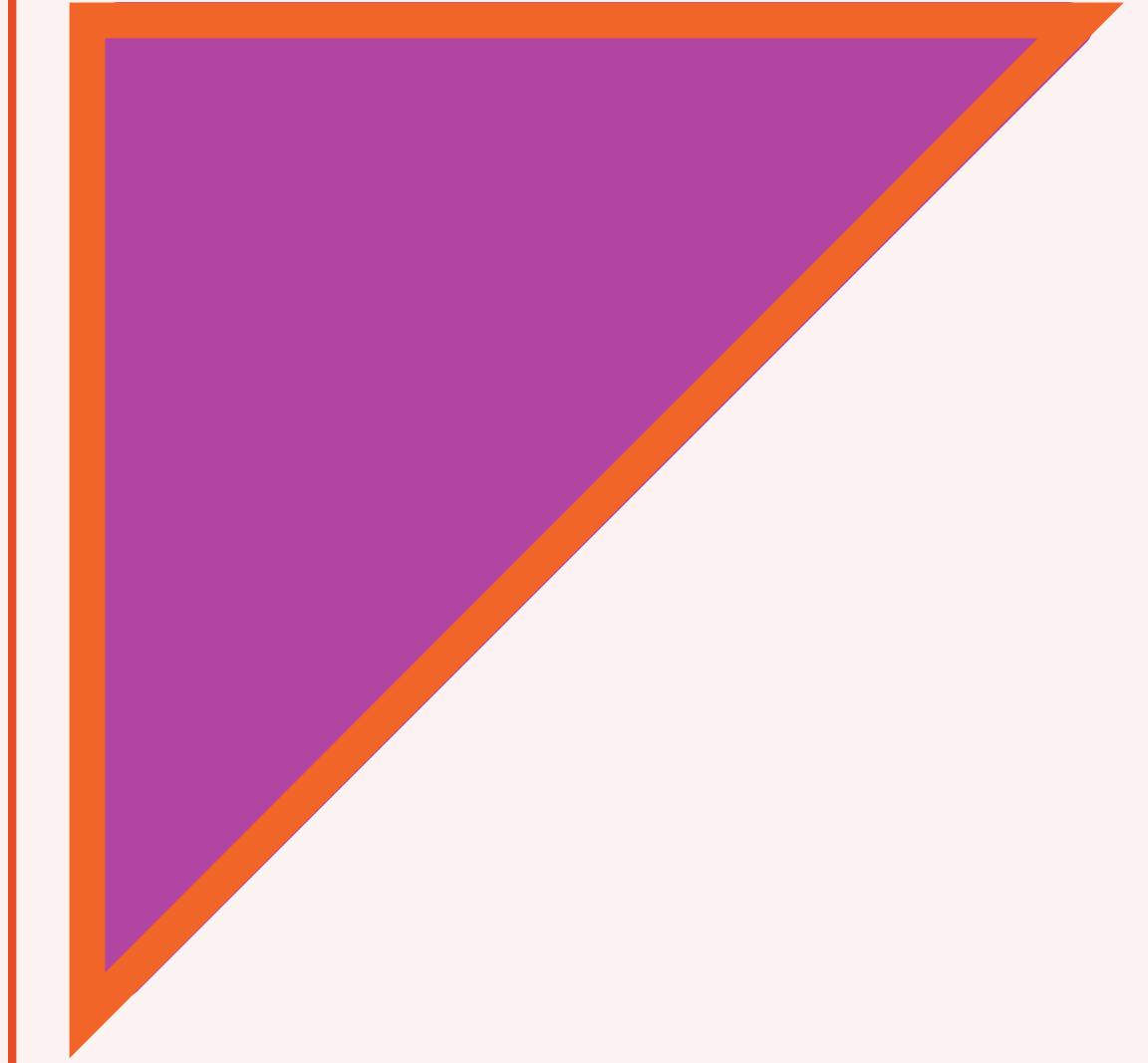
```
```xsvg:0,0,20,20
polygon 0,0 20,0
0,20
orange #B244A2
```
```

HTML

```
<svg viewBox="0 0
20
20">

```

Rendered



XSVG

G = (T, N, P, S)

T = {, line, polyline, polygon, circle, ellipse, rect, text, white, red, blue, yellow, cyan, black, purple, orange, INT, DIG, LET, TEXT, middle, start, end, ε, “, xsvg, :}

N = {<color>, <canva>, <hex>, <position>, <canva>, <svg element>, <point>, <line>, <polyline>, <polygon>, <circle>, <ellipse>, <rect>, <text>}

S = <canva>

$$G = (T, N, P, S)$$

$P = \{ \langle \text{line} \rangle: \text{line} \langle \text{point} \rangle \langle \text{point} \rangle \langle \text{point} \rangle \langle \text{point} \rangle \langle \text{color} \rangle (\langle \text{color} \rangle | \varepsilon)$
 $\langle \text{polyline} \rangle: \text{polyline} \langle \text{point} \rangle \langle \text{point} \rangle \langle \text{point} \rangle \langle \text{point} \rangle \langle \text{color} \rangle (\langle \text{color} \rangle | \varepsilon)$
 $\langle \text{polygon} \rangle: \text{polygon} \langle \text{point} \rangle \langle \text{point} \rangle \langle \text{point} \rangle \langle \text{color} \rangle (\langle \text{color} \rangle | \varepsilon)$
 $\langle \text{circle} \rangle: \text{circle} \langle \text{point} \rangle \text{INT} \langle \text{color} \rangle (\langle \text{color} \rangle | \varepsilon)$
 $\langle \text{ellipse} \rangle: \text{ellipse} \langle \text{point} \rangle \text{INT INT} \langle \text{color} \rangle (\langle \text{color} \rangle | \varepsilon)$
 $\langle \text{rect} \rangle: \text{rect} \langle \text{point} \rangle \text{INT INT} \langle \text{color} \rangle (\langle \text{color} \rangle | \varepsilon)$
 $\langle \text{text} \rangle: \text{text} \langle \text{point} \rangle \text{“TEXT”} \langle \text{position} \rangle (\langle \text{color} \rangle | \varepsilon) \}$

G = (T, N, P, S)

P = { <hex>: #(DIG|LET)(DIG|LET)(DIG|LET)(DIG|LET)(DIG|LET)(DIG|LET)
<color>: <hex> | white | red | blue | yellow | cyan | black | purple | orange
<position>: middle | start | end
<point>: INT,INT
<canva>: "xsvg:INT,INT,INT,INT<newline> (<svg element><newline>)*"
<svg element>: <polygon> | <line> | <polyline> | <circle> | <ellipse> |
<rect> | <text> }



COMPLETE GRAMMAR

Complete grammar

$$G = (T, N, P, S)$$

$T = \{TEXT, *, _, \sim, \#, !, [,], (,), <, ` , - , \n, \, , >, ., + , \{ , \} , | , line, polyline, polygon, circle, ellipse, rect, text, white, red, blue, yellow, cyan, black, purple, orange, INT, DIG, LET, middle, start, end, \varepsilon, xsvg, : \}$

Complete grammar

$G = (T, N, P, S)$

$N = \{<\text{paragraph}>, <\text{bold}>, <\text{italic}>, <\text{underlined}>, <\text{strikethrough}>, <\text{titles}>, <\text{h1}>, <\text{h2}>, <\text{h3}>, <\text{h4}>, <\text{h5}>, <\text{h6}>, <\text{image}>, <\text{block quote}>, <\text{nested block quote}>, <\text{multiple block quote}>, <\text{single block quote}>, <\text{code}>, <\text{block code}>, <\text{horizontal rule}>, <\text{new line}>, <\text{bold italic}>, <\text{list}>, <\text{url}>, <\text{escape sequence}>, <\text{color}>, <\text{canva}>, <\text{hex}>, <\text{position}>, <\text{canva}>, <\text{svg element}>, <\text{point}>, <\text{line}>, <\text{polyline}>, <\text{polygon}>, <\text{circle}>, <\text{ellipse}>, <\text{rect}>, <\text{text}>\}$

Complete grammar

$G = (T, N, P, S)$

$P = \langle \text{markdown doc} \rangle : (\langle \text{paragraph} \rangle \mid \langle \text{canva} \rangle)^*$

$\langle \text{paragraph} \rangle : (\langle \text{title} \rangle \mid \langle \text{list} \rangle \mid \text{TEXT} \mid \langle \text{bold} \rangle \mid \langle \text{italic} \rangle \mid \langle \text{underline} \rangle \mid \langle \text{strikethrough} \rangle \mid \langle \text{links} \rangle \mid \langle \text{images} \rangle \mid \langle \text{inline code} \rangle \mid \langle \text{bold and italics} \rangle \mid \langle \text{escape sequence} \rangle \mid \langle \text{url} \rangle)^*$

$\langle \text{newline} \rangle$

$\langle \text{bold} \rangle : \text{**TEXT**}$

$\langle \text{italic} \rangle : \text{*TEXT*}$

$\langle \text{strikethrough} \rangle : \text{~~TEXT~~}$

Complete grammar

$G = (T, N, P, S)$

$P = <\text{title}>: <\text{h1}> | <\text{h2}> | <\text{h3}> | <\text{h4}> | <\text{h5}> | <\text{h6}>$

$<\text{h1}>: \# \text{TEXT}$

$<\text{h2}>: \#\#\# \text{TEXT}$

$<\text{h3}>: \#\#\#\# \text{TEXT}$

$<\text{h4}>: \#\#\#\#\#\# \text{TEXT}$

$<\text{h5}>: \#\#\#\#\#\#\# \text{TEXT}$

$<\text{h6}>: \#\#\#\#\#\#\#\# \text{TEXT}$

$<\text{h1}>: \# \text{TEXT}$

Complete grammar

$G = (T, N, P, S)$

$P = <\text{image}> : ![\text{TEXT}] (\text{TEXT})$

$<\text{nested block quote}> : > <\text{single block quote}>$

$<\text{multiple block quote}> : ((<\text{single block quote}> | <\text{nested block quote}> <\text{new line}> > <\text{new line}> (<\text{single block quote}> | <\text{nested block quote}>)) +$

$<\text{single block quote}> : > <\text{paragraph}>,$

$<\text{block quote}> : <\text{single quote}> | <\text{nested block quote}> | <\text{multiple block quote}>$

Complete grammar

$$G = (T, N, P, S)$$

P = **<code>**: ` TEXT `

<block code>: "" (TEXT | **<newline>**)^{*} ""

<horizontal rule>: <newline><newline> (***+ | --+ | __+)

<newline><newline>

<new line>: \n

<bold italic>: ***TEXT*** | __TEXT__

<list>: (* TEXT<newline>)+ | (- TEXT<newline>)+

<url>: <TEXT> | TEXT

Complete grammar

$G = (T, N, P, S)$

$P = \langle \text{escape sequence} \rangle: "\\" | "\!" | "\?" | "_" | "*" | "\{" | "\}" | "\(" | "\)" | "\<" | "\>" | "\[" | "\]" | "\#" | "\+" | "\-" | "\." | "\|"$

$\langle \text{canva} \rangle: ""\text{xsvg:INT,INT,INT,INT} \langle \text{newline} \rangle (\langle \text{svg element} \rangle \langle \text{newline} \rangle)^*$

$\langle \text{hex} \rangle: \#(\text{DIG} | \text{LET})(\text{DIG} | \text{LET})(\text{DIG} | \text{LET})(\text{DIG} | \text{LET})(\text{DIG} | \text{LET})(\text{DIG} | \text{LET})$
 $(\text{DIG} | \text{LET})$

Complete grammar

$G = (T, N, P, S)$

P = **<color>** : <hex> | white | red | blue | yellow | cyan | black |
purple | orange

<position>: middle | start | end

<svg element>: <polygon> | <line> | <polyline> | <circle> |
<ellipse> | <rect> | <text>

<point> INT,INT

<line>: line <point><point><point><point> <color>(<color>|ε)

Complete grammar

$G = (T, N, P, S)$

P = **<polyline>**: polyline <point><point><point><point> <color> (<color>| ϵ)

<polygon>: polygon <point><point><point> <color> (<color>| ϵ)

<circle>: circle <point> INT <color> (<color>| ϵ)

<ellipse>: ellipse <point> INT INT <color> (<color>| ϵ)

<rect>: rect <point> INT INT <color> (<color>| ϵ)

<text>: text <point> “TEXT” <position> (<color>| ϵ)

S = **<markdown doc>**

Next Steps

01

27/11/2023

Scanner

02

04/12/2023

AST

03

22/12/2023

Paser

04

12/01/2024

Code
generation

05 2nd half of january

Final Presentation