**Regex Basics**

A regular expression is a pattern that matches strings.  
In C++, regex(R"(pattern)") is used to create regex patterns.  
Some useful symbols:

* . → any character
* \* → 0 or more repeats
* + → 1 or more repeats
* ? → optional (0 or 1)
* \ → escape special characters like \*, (, etc.
* (abc) → capture group
* ^ → start of line
* $ → end of line

**Regex Patterns in my Code**

**1. Bold Text (\*\*bold\*\*)**

regex(R"(\\*\\*(.\*?)\\*\\*)")

Matches anything between \*\* and \*\*.

| **Part** | **Meaning** |
| --- | --- |
| \\*\\* | Escaped \*\* to match actual asterisks |
| (.\*?) | Match *any characters* (non-greedy), save them |
| \\*\\* | End with another \*\* |

**Example**: \*\*bold\*\* → match "bold" → output: <strong>bold</strong>

**2. Italic Text (\*italic\*)**

regex(R"(\\*(.\*?)\\*)")

Similar to bold, but with only one \*.

**Example**: \*italic\* → <em>italic</em>

This will also match inside bold like \*\*bold \*italic inside\* bold\*\* unless escaped or made stricter.

**3. Image (![alt](url))**

regex(R"(!\[(.\*?)\]\((.\*?)\))")

| **Part** | **Meaning** |
| --- | --- |
| !\[ | Match literal ![ |
| (.\*?) | Capture alt text inside [] |
| \]\( | Match ]( exactly |
| (.\*?) | Capture URL inside () |
| \) | Match ) |

**Example**: ![logo](img.png) → <img src="img.png" alt="logo" />

**4. Link ([text](url))**

regex(R"(\[(.\*?)\]\((.\*?)\))")

Almost same as image but no !.

**Example**: [Google](https://google.com) → <a href="...">...</a>

**5. Horizontal Rule (---)**

regex(R"(\s\*---\s\*)")

| **Part** | **Meaning** |
| --- | --- |
| \s\* | 0 or more spaces before and after |
| --- | Match three dashes exactly |

This ensures that even if someone writes:

---

…it still works.

**6. List Items (- item)**

regex(R"(\s\*-\s.+)")

| **Part** | **Meaning** |
| --- | --- |
| \s\* | 0 or more spaces (allows indentation) |
| - | Match dash |
| \s | One space after the dash |
| .+ | One or more characters (the list text) |

Matches:

- Item

- Subitem

**Extra Tip: Inline Regex Matching in C++**

C++ uses std::regex\_replace() to apply regex:

string result = regex\_replace(original, regex("pattern"), "replacement");

For example:

string x = "\*\*bold\*\*";

x = regex\_replace(x, regex(R"(\\*\\*(.\*?)\\*\\*)"), "<strong>$1</strong>");

$1 refers to the **first captured group** (i.e., bold).

**Summary Table**

| **Purpose** | **Regex Pattern** | **Example Input** | **HTML Output** |
| --- | --- | --- | --- |
| Bold | \\*\\*(.\*?)\\*\\* | \*\*bold\*\* | <strong>bold</strong> |
| Italic | \\*(.\*?)\\* | \*italic\* | <em>italic</em> |
| Link | \[(.\*?)\]\((.\*?)\) | [Google](url) | <a href="url">Google</a> |
| Image | !\[(.\*?)\]\((.\*?)\) | ![alt](img.jpg) | <img src="img.jpg" alt="alt" /> |
| Horizontal Rule | \s\*---\s\* | --- | <hr /> |
| List Item | \s\*-\s.+ | - item | <ul><li>item</li></ul> |