1. 计算 $x + y$
2. 第一布里渊区的范围是：$-\frac{\pi}{a} < k < \frac{\pi}{a}$
3. 白日依山尽，黄河入海流

问题：

- 如果传入的公式是行间公式，不清楚如何放置下划线

  $$\int f(x)\,\mathrm{d}x$$

  那么会怎么样呢？是否会有缩进 a very very very very very very very very very very very very very very long text
- 不清楚如何针对公式的宽度自适应下划线的位置
  对于 $\mathrm{p^{+}n}$ 结，其扩散电容的表达式为：$C_D = \left(\frac{Aq^2 p_{n_0} L_p}{k_o T}\right) \exp\left(\frac{qV}{k_o T}\right)$

你好。下面是一段测试文字 。

Python

```python
x = 3
print("Hello, World!")
```

Here is the content of the variable **with style**

Hello World   Typst

```typst
#let forecast(day) = block[
  #box(square(
    width: 2cm,
    inset: 8pt(c),
    fill: if day(a).weather == "sunny"(b) {
      yellow
    } else {
      aqua
    },
    align(
      bottom + right,
      strong(day.weather),
    ),
  ))
  #h(6pt)
  #set text(22pt, baseline: -8pt)
  #day.temperature °#day.unit // 你好呀
]
#let forecast(day) = block[
  #box(square(
    width: 2cm,
    inset: 8pt,
    fill: if day.weather == "sunny" {
      yellow
    } else {
      aqua
    },
    align(
      bottom + right,
      strong(day.weather),
    ),
  ))
  #h(6pt)
  #set text(22pt, baseline: -8pt)
  #day.temperature °#day.unit
]
```

`zebraw` 测试

Adding `rbx` to `rcx` gives the desired result.

What is `fn main()` in Rust would be `int main()` in C.

Rust

```rust
fn main() {
    println!("Hello World!");
}
```

This has `` `backticks` `` in it (but the spaces are trimmed). And `here` the leading space is also trimmed.

Hello world

Hello World

10

raw(text: "strong", block: false, lang: "typc")
`content`

`Hello world ! this is a very very very very very very very very very very very long text`