FHVirus' HomeWork template in Typst

Contents

1 Basic Advanced Usage

1.1 Enumerate

Enumerate can be done like this:

```
+ —
+ deux
5. fünf
  #set enum(numbering: "i.")
  + and you can
  + nest them.
  + [item6] Custom item label.
+ sex
```

The code will become:
1. ⟶
2. deux
5. fünf
    i. and you can
   ii. nest them.
   iii. There seems to be no custom item label.
6. sex

Note: Typst's default list is tight! No need to `\tightlist` anymore.

1.2 Math Symbols

Typst has its own set of notations (which is far easier than L^AT_EX IMO).

```
$ #let contra = { $arrow.r #h(-0.1em) arrow.l$ }
  #let inner(x) = { $lr(angle.l #x angle.r)$ }
  NN ZZ QQ RR CC contra \
  abs(a), ceil(b), floor(c), inner(x / 2), norm(y), \
  lr({1, 2, 3 / 7}), a dot b = c, x * y = z $
```

$\mathbb{N}\mathbb{Z}\mathbb{Q}\mathbb{R}\mathbb{C} \rightarrow\!\leftarrow$

$|a|, \lceil b \rceil, \lfloor c \rfloor, \langle \frac{x}{2} \rangle, \|y\|,$

$\left\{1, 2, \frac{3}{7}\right\}, a \cdot b = c, x * y = z$

\left and \right in L^AT_EX are not needed since its the default in Typst (hooray!)

I placed contra and inner(x) inside math scope to make the code cleaner. Since they are seldomly used, I do not define them in the template.

```
$ #let contra = { $arrow.r #h(-0.1em) arrow.l$ }
  #let inner(x) = { $lr(angle.l #x angle.r)$ }
```

2 Blocks

2.1 Theorems, Lemmas, …

```
#theorem[ #lorem(30) ]
#lemma[ They have seperate numbers. ]
#claim[ Claim for proof! ]
#observation[
  My observation. You can make `<label>`s...
] <myob>
#lemma("My Lemma")[
  ...give them names, and reference them like this: \
  According to @myob...
]
#proof[ Proofs have cool squares. ]
```

Theorem 2.1.1: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri.

Lemma 2.1.2: They have seperate numbers.

Claim 2.1.3: Claim for proof!

Observation 2.1.4: My observation. You can make `<label>` s...

Lemma 2.1.5 (My Lemma): ...give them names, and reference them like this:
According to Observation 2.1.4...

Proof: Proofs have cool squares.   □

More example on the next page...

```
#definition[
  A natural number is called a #highlight[_prime number_] if it is greater
  than 1 and cannot be written as the product of two smaller natural numbers.
]
#example[
  The numbers $2$, $3$, and $17$ are prime.
  @cor_largest_prime shows that this list is not exhaustive!
]

#theorem("Euclid")[ There are infinitely many primes. ]
#proof[
  Suppose to the contrary that $p_1, p_2, dots, p_n$ is a finite enumeration
  of all primes. Set $P = p_1 p_2 dots p_n$. Since $P + 1$ is not in our list,
  it cannot be prime. Thus, some prime factor $p_j$ divides $P + 1$.  Since
  $p_j$ also divides $P$, it must divide the difference $(P + 1) - P = 1$, a
  contradiction.
]

#corollary[
  There is no largest prime number.
] <cor_largest_prime>
#corollary[
  There are infinitely many composite numbers.
]
```

Definition 2.1.6 *(Prime Number)*: A natural number is called a ==prime number== if it is greater than 1 and cannot be written as the product of two smaller natural numbers.

Example: The numbers 2, 3, and 17 are prime. Corollary 2.1.7.1 shows that this list is not exhaustive!

Theorem 2.1.7 (Euclid): There are infinitely many primes.

Proof: Suppose to the contrary that $p_1, p_2, ..., p_n$ is a finite enumeration of all primes. Set $P = p_1 p_2 ... p_n$. Since $P + 1$ is not in our list, it cannot be prime. Thus, some prime factor p_j divides $P + 1$. Since p_j also divides P, it must divide the difference $(P + 1) - P = 1$, a contradiction. $\square$

Corollary 2.1.7.1: There is no largest prime number.

Corollary 2.1.7.2: There are infinitely many composite numbers.

For more examples, visit https://github.com/sahasatvik/typst-theorems .

2.2 Problems

```
#problem[
  This is a problem.
]
#problem[ This is another problem. ]
#problem(same-page: true)[ If you don't want `#problem` to start on a new page,
you can pass `same-page: true`. ]
#problem(same-page: true, number: "0.514")[ You can also number them
yourself! ]
#problem(same-page: true)[ Problem numbering is independent, too. ]
```

Problem 1

This is a problem.

Problem 2
This is another problem.

Problem 3
Problems by default start on new page for Gradescope support. If you don't want `#problem` to start on a new page, you can pass `same-page: true` .

Problem 0.514
You can also number them yourself!

Problem 5
Problem numbering is independent, too.

Problem 6

> **Theorem 2.2.1**: Theorem numbering does not follow problems.

> **Theorem 6.1**: I'm sorry but you have to do the numbering yourself.

Problem numbering is independent, too.

2.3 Code

Just use the built-in one!

```
```typ
#include <iostream>
int main() {}
```
```

produces:

```
#include <iostream>
int main() {}
```

And with `#raw(read("a.cpp"), lang: "cpp", block: true)` we get:

```cpp
// This line will not be shown.
// 這個會出現…
#include <bits/stdc++.h>
#define MAXN 7122

int main() {
  std::cin.tie(0)->sync_with_stdio(0);
  std::cin.exceptions(std::cin.failbit);

  int a = 998244353;
  const char c = 'x';

  std::cout << "Hello world!" << std::endl;

  return 0;
}
```

2.4 Image

Just use https://typst.app/docs/reference/visualize/image/ .

3 Drawing

🚧 TODO 🚧 CetZ seems powerful, yet I'm not sure what can I achieve with the built-in functions and when should I use the package.

4 Lorem Ipsums

4.1 Chinese lorem

顏色香半，褕海知，舉臥之斯文大…**萬里送**，Lorem ipsum dol'or sit amet, consectetuer adipiscing elit. 幾度九華帳山色之難難，楚能余亦之難難夜日行陽為君。萬事仙臥月南風何十怨遙夜…風沙，師落葉滿秋，歲閣自照露，征柳啾啾月一曲然不得有殷勤，遠不到酒稀清輝。腸何時還，行路難獨夜南…羅微鳳路雪虛征戰。鳴看煙，衣裳多斜馬，相見秋松下孤城君不見西羽歲王孫：見臨烽火桃李但見茫茫獨，問見得城月涕淚長可聖兒夢不成，掩至今歌。雖識月明如此，曲夢在新到天秋一，寂昔不逢闌干，窗東流水腸斷角不見清天與故人，下兒雨千門微遲里劍閣白，石鼓。

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et.

4.2 Ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri tamen permagna accessio potest, si aliquod aeternum et infinitum impendere malum nobis opinemur. Quod idem licet transferre in voluptatem, ut postea variari voluptas distinguique possit, augeri amplificarique non possit. At etiam Athenis, ut e patre audiebam facete et urbane Stoicos irridente, statua est in quo a nobis philosophia defensa et collaudata est, cum id, quod maxime placeat, facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et.

沒有好棒，別鳴哈哈哈，都這個實說不然也不，的人的什麼最好。經不院可以去為什麼有，沒該會人嗎亡如果是，再次存跟我試圖，只有直接說這是放到最近的的影片…陌生人…也知道個月的。其覺得是起來跟不可能也很，畫面待喔好天的簡單很不是聽起來，然已經就聖我的話的聲。再然後道了嗎傍晚時害的，男的話就需要一，或許是的男，去的方法我愛次自行要跟…了沒好會的角色這邊，一把的人會出們看記…