Notexbook Documentation

C. H. Lay

July 28, 2018

Contents

Chapter 1

Usage Principles

Any *ordinary page* is a standalone article. The article formats are optimized for computer science, mathematics and biology. The purpose is to write individual documents to refer back in the future without a need to reread from original sources. The base level for subject titles is *section*.

Because the articles are written in LaTeX, you can use any package that do not have conflicts with the commands defined here. However, if they do it is easier to conflicting ones in `Theme.sty` file.

Chapter 2

The Stylings

2.1 Terms

Many scientific texts seem to italicize a new term whenever it first occurs. We also follow that convention by italicizing and coloring the term.

Example 1. The *term style* is defined in `\term` command.

Chapter 3

Pseudocode Language

Pseudocode is written in a pseudolanguage which is a mix of Kotlin, Python and C. It also grows to support new programming paradigms whenever necessary. The core feature is to provide compact notations and readable syntax.

Also note that I have not designed or invented any of the features I am describing. This is just a way to organize and simplify the programmatic thought processes.

3.1 Basic Syntax

1. Every expression is ended by ';' character.
2.

3.2 Comments

Example 2. The language supports two kinds of comments.

```
# This is a single-line comment.
/*
 *  The multi-line comment.
 */
```

3.3 Data Types

Example 3. The data types are declared similarly to UML inspired by Kotlin.

```
a: Integer = 1         # or Int.
b: Char = 'j'          # Is equivalent to Integer.
d: String = "example" # or Str, is equivalent to Char[].
```

3.4 Control Flow

3.5 Functions

Example 4. The functions are inspired by Kotlin and Scheme.

```
define square(x: Integer)
{
    return (* x x);
}: Integer

# The function above can be simplified to
{def square(x: Int) (* x x)}: Int;
```

3.5.1 Iterative Process

3.5.2 Recursive Process

3.6 Standard Library

Lets fantasize that everything is accessible. The basic or common functions are defined in the `Theme.sty`.