

Rubyから広がるプログラミング入門教育

～小学校高学年向けプログラミング入門書『ユウと魔法のプログラミング・ノート』執筆から学んだこと～

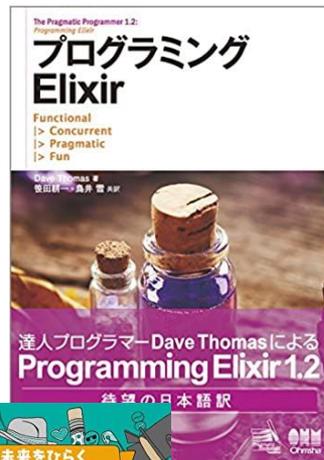


自己紹介

株式会社万葉のWebプログラマ

翻訳いろいろ

- プログラミング絵本『ルビィのぼうけん』
リンダ・リウカス
- 中高生向けプログラミング入門書『Girls Who Code 女の子の未来をひらくプログラミング』レシュマ・サウジャニ
- 『プログラミングElixir』デイブ・トーマス(笹田 耕一と共訳)



今日話すこと

- 『ユウと魔法のプログラミング・ノート』紹介(さわり)
- プログラミングの概念を**伝える言葉**にするために必要なこと
- **Ruby**の影響とこれからの教育

『ユウと魔法のプログラミング・ノート』紹介

『ユウと魔法のプログラミング・ノート』

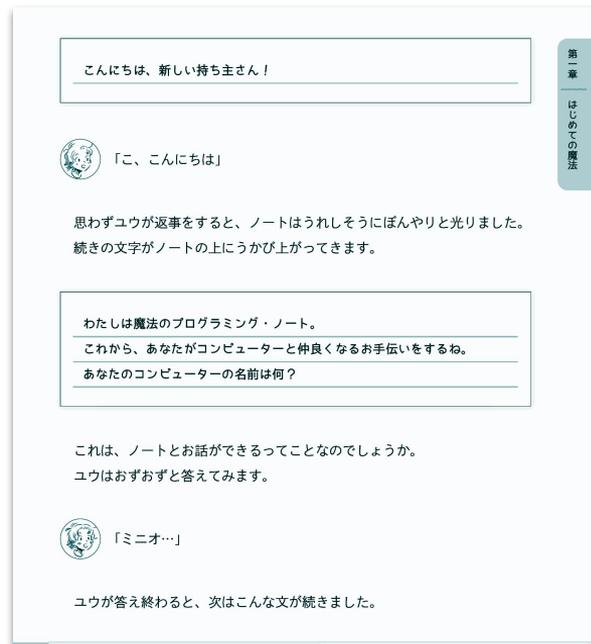
対象年齢：小学校高学年～

ストーリー：小学5年生のユウは、自分だけのコンピューター・ミノオを買ってもらいます。わすれっぽい自分をミノオに助けてもらおうと思いますが、ミノオは思った通りには動きません。そんな時、ユウのもとに魔法のプログラミング・ノートがやってきて、ミノオのプログラミングを手伝ってくれることになり……？

特定のプログラミング言語に限定せず、「プログラミングをする際の考え方」を解説する

コードは日本語の疑似コードを用いる

2022年12月→**2023年春** 刊行予定



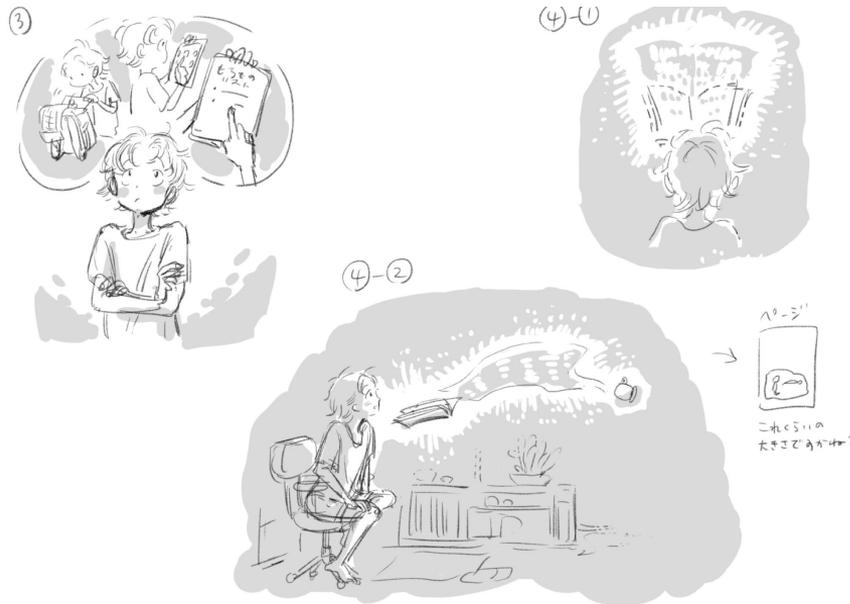
※イラスト・版組はラフの状態です。出版時には変更されま
ず。

ポイント1: ストーリーのある入門書

主人公のユウが、自分の身の回りの問題を一つずつ、プログラミングとコンピューターの方で解決していく構成。

プログラミング = 自分の問題の解決

のイメージを持ってもらう



※イラスト・版組はラフの状態です。出版時には変更されません。

ポイント2: イメージを広げ、物語世界に引き込むイラスト

挿画：鶴谷香央里（漫画『メタ
モルフォーゼの縁側』など）



※イラスト・版組はラフの状態です。出版時には変更されま
す。

『ユウと魔法のプログラミング・ノート』で扱う項目

- 変数 第二章 ユウのおてつだいポイント
- 条件分岐 第三章 水曜バスケットボール・クラブ・プログラム
- 繰り返し 第四章 みんなにメッセージを～くり返し～
- 配列 第五章 「みんな」をまとめて
- 関数 第六章 さいごの試練

※イラスト・版組はラフの状態です。出版時には変更されません。

『ユウと魔法のプログラミング・ノート』で扱う項目

- 変数
- 条件分岐
- 繰り返し
- 配列
- 関数

第二章 ユウの魔法

第三章

第四章

第五章

第六章

プログラム



これだけ??

これだけとはなんだ！
これでも解説できるギリ
ギリ精一杯だ！！！！

お題:「変数」を小学五年生に言葉で説明
してください

例:さまざまな子ども向けプログラミング書の「変数」解説

「WAKU_COLORの後ろに=（イコール）と数字があるね。これらの数字を変えると、」（以下動作が変化する説明）（『小学生から楽しむRubyプログラミング』 p31）

「変数は、数字や文字を置いておく入れ物だよ。いろんなものを入れておく鞆のようなイメージかな。入れ物なので、プログラムが動いているときに中身を入れ替えることができるんだ」（『Raspberry-Piではじめるときどきプログラミング』 p68）

「情報を保管するためのスペース。名前が付けられていて、中の値を変えることができる」（『10歳からはじめるプログラミング図鑑 たのしくまなぶスクラッチ&Python』用語集 p219）

例:さまざまな子ども向けプログラミング書の「変数」解説

「WAKU_COLORの後ろに=（イコール）と数字があるね。これらの数字を変えると、」（『小学生から楽しむRubyプログラミング』 p31）

「変数は、**プログラムを「動かす」ことを前提として** おく鞆の
ようなイメージ**いる書籍の解説はこれで十分。** 身を入れ
替えること**使ってみれば体感できる。** プログラミン
グ』 p68)

「情報を保管するためのスペース。名前が付けられていて、中の値を変えることができる」（『10歳からはじめるプログラミング図鑑 たのしくまなぶスクラッチ&Python』用語集 p219）

“言葉”として
変数を体感してもらうには
は
一章分の分量が必要.....

プログラミングの概念を伝える言葉にする
ために必要なこと

伝わる言葉にする手順

1. 使う場面を設定する、想像する
2. 概念を言葉で分解する
3. 分解に使った言葉を、理解できる言葉に直していく

1. 使う場面を設定する、想像する

「変数を使ってみよう」→ なんのために??

ストーリーを設定して実際に使う状況を具体的に想像する



例：ユウはママのお手伝いをすると「おてつだいポイント」をもらえる。

おてつだいポイントを計算するプログラムを作ろう。

プログラムの中で、今のおてつだいポイントに1足した結果をミニオ（コンピュータ）に言ってもらうには？

2. 概念を言葉で分解する

変数とはなにか、変数を使うとはどういうことか

```
points = 3
```

→ points という名前の変数がある

3という数値のデータがある

変数 = データ で何が起こるか？ 変数がデータを指し示すようになる。指し示す先のデータは変数 = 別データ で何度でも変更できる。

変数とは、データを指し示す名前である。

プログラムの中で、変数はコンピューターに命令を上手に伝えるための道具である。

3. 分解に使った言葉を、理解できる言葉に直していく

— — —
<https://slow-communication.jp/info/1428/>

一般社団法人スローコミュニケーション

『「わかりやすさ」をつくる13のポイント』



「わかりやすさ」をつくる13のポイントのうち 4ポイント

―――

- フォントを選ぶ
- スペースを入れる
- 情報の絞り込み
- 具体的な説明
- 漢字の仕様とルビ
- 呼び方を統一する
- 主語をはっきりさせる
- 文は適度に短くする
- 漢語を避ける
- 漢字の連続を避ける
- なじみのある語はそのまま
- 「のりしろ」をつくる
- 知的障害の特性に合わせる
 - (当事者の意見を聞く)
 - (対面で説明する)

「わかりやすさ」をつくる13のポイントのうち4ポイント

――

- フォントを選ぶ
 - スペースを入れる
 - 情報の絞り込み
 - 具体的な説明
 - 漢字の仕様とルビ
 - 呼び方を統一する
- } 構成
- 主語をはっきりさせる
 - 文は適度に短くする
 - 漢語を避ける
 - 漢字の連続を避ける
 - なじみのある語はそのまま
 - 「のりしろ」をつくる
 - 知的障害の特性に合わせる
 - (当事者の意見を聞く)
 - (対面で説明する)
- } 文章

構成について1. 情報の絞り込み

おこずかいポイント（変数）の章で最初に用意したトピック

- ・ 永続化データの保存とよび出し
- ・ 変数名の重要性
- ・ 変数への代入
- ・ 変数の呼び出し
- ・ 変数を使った計算と再代入



ストーリー：

- ・ 変数への代入
- ・ 変数の呼び出し
- ・ 変数を使った計算と再代入

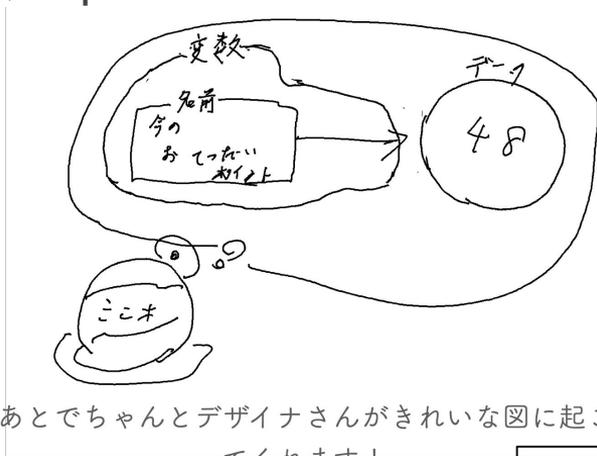
ワーク：永続化データの保存とよび出し

コラム：変数名の重要性

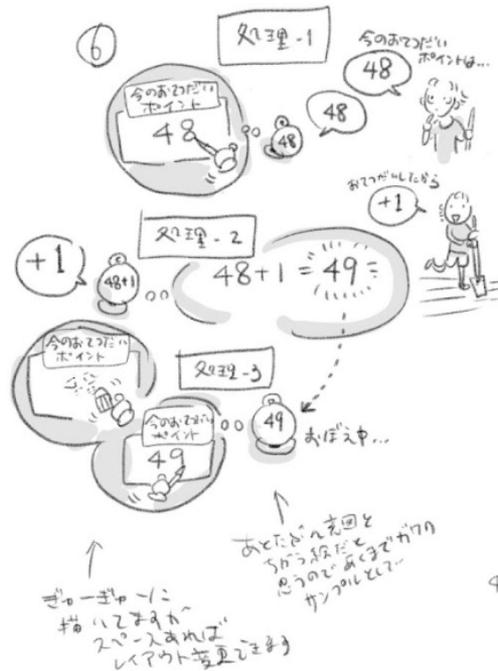
構成について2. 具体的な説明

比喩を最低限しか使わない（プログラムの道具）
変数を「箱」としない

図による理解のサポート



※あとでちゃんとデザイナーさんがきれいな図に起こしてくれませう！



※イラスト・版組はラフの状態です。出版時には変更されませう。

文章について1. 主語をはっきりさせる

これで完璧！

もしミニオだったら、この命令で十分なはず
です。だって、数をまちがえることはなさそうで
すし、一度覚えたものをまちがって思い出した
り、うっかり忘れてたりすることもないでしょう。

ユウはこれで完璧だと思いました。

もしユウがミニオだったら、この命令で十分
だと思ったのです。だって、コンピューターは、
数をまちがえることはなさそうです。それにコ
ンピューターは、一度覚えたものをまちがって
思い出したり、うっかり忘れてたりすることもない
でしょう。

文章について2. 文は適度に短くする

だって、数をまちがえることはなさそうですし、一度覚えたものをまちがって思い出したり、うっかり忘れてたりすることもないでしょう。



だって、コンピューターは、数をまちがえることはなさそうです。それにコンピューターは、一度覚えたものをまちがって思い出したり、うっかり忘れてたりすることもないでしょう。

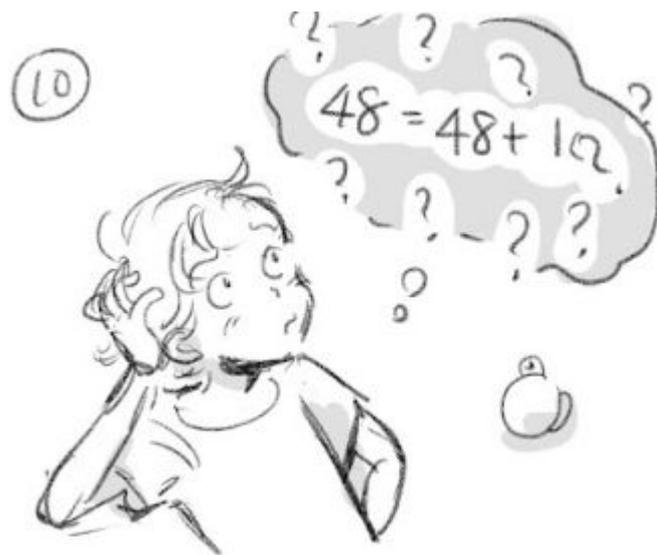
伝わるかな？

変数っていうのは、コンピューターの中にあるいろんなデータを、プログラムの中で覚えておくための、道具だよ。

プログラムの中で変数という道具を使うと、ミニオは今のおてつだいポイントっていうデータを忘れずに覚えておけるし、出力でも思い出せるようになる。

(ストーリーが入る)

変数って、「データを指し示している名前」のこと！



※イラスト・本文・版組はラフの状態です。出版時には変更されます。

Rubyの影響とこれから

Rubyの影響

「繰り返し」で最初に思い浮かべるコード

```
array = [1, 2, 3, 4]
array.each { |e| puts e }
```

- データを並べた配列がある
- 配列の要素一つずつに繰り返し処理する
- 配列の要素がなくなったら繰り返しを終了する

Rubyは使いたい**状況**をうまく抽出して抽象化している

分解してこうも書ける

```
array = [1, 2, 3, 4]

i = 0
while i <= 4 do
  puts array[i]
  i += 1
end
```

分解度に合わせてさまざまな書き方ができるRubyはプログラミングの習得に向いている

Rubyが(特に子供向けの)第一言語に選ばれるには何が必要か？

- Rubyの存在を知る機会
- 平易な日本語によるRubyの文法ガイド
- 気軽にRubyを動かして試せる日本語環境

あんまりないので**作りたい**

これから作りたいもの(と、スポンサー募集)

プログラミング・ノート内疑似コードをRubyのコードとして書いて動かせる

プログラミング・ノートの内容に連動した平易な日本語によるRuby文法のガイドがある

プログラミング・ノートの登場人物や舞台の挿画があしらわれて、物語との連続性を感じられる

※赤字の部分にお金がかかります！！！！！！

Rubyでこれからの教育、やっていきましょう！（よろしくお願いします）