

OSS2 第1回

岩澤全規

自己紹介

- 名前：岩澤全規
- 出身地：東京都葛飾区
- 専門分野：高性能計算、天文学
 - FDPS(Framework For Developing Particle Simulators)の開発
 - 重力多体系（銀河、星団）
- 趣味：ギター、旅行

講義の目標

- Rubyの基本的な使い方を習得する。
- オブジェクト指向プログラミングを理解する。
- Linuxの基本操作を習得する。

講義でやる事

- Ruby
 - 実行方法
 - 入出力
 - 制御構造
 - 配列、連想配列
 - 正規表現
 - メソッド(関数)
 - クラス
- (Crystal): Rubyインスパイア系言語(速い！)

今日やること

- 環境構築
- 設計方法
 - インテグラル設計
 - モジュール設計
- モジュール化プログラミング
 - オブジェクト指向プログラミング
- Rubyの実行方法

環境構築

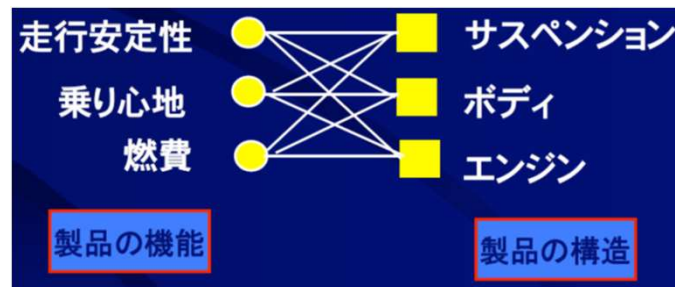
- この講義ではVMware上でUbuntu20を用います。
- やる事：
 - USBメモリ(SSD)を回すので、それを¥C¥vmにコピー
 - プログラムからVMware workstation playerを起動。
 - “仮想マシンを開く”から、¥C¥vm¥ubuntu20¥ubuntu 64ビット.ovfを選択。
 - インストールできたらログイン
 - ユーザー名: OSS2
 - パスワード: admin2020

設計方法

- インテグラル設計(すり合わせ設計)

- 個々の製品を相互に調整して全体を組み合わせる。

- 例) 車



藤本隆宏

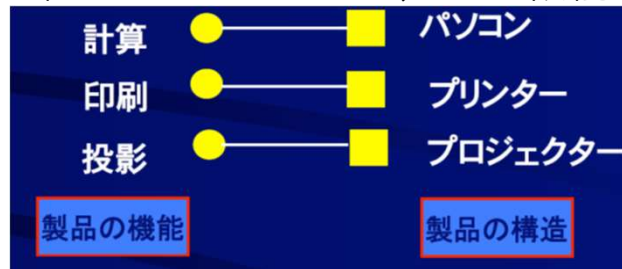
https://ocw.u-tokyo.ac.jp/lecture_files/gf_12/9/notes/ja/09fujimoto.pdf

- 機能が複数の製品に依存している。
- 製品をお互いすり合わせながら組み合わせて、性能を高めたり、小型化する。

設計方法

- モジュール設計

- 独立した製品(モジュール)を用意し組み合わせる
 - モジュール(module): 組み立てユニット、ある機能を持ったパーツ。
 - パソコンシステム



藤本隆宏

https://ocw.u-tokyo.ac.jp/lecture_files/gf_12/9/notes/ja/09fujimoto.pdf

- パソコン自体

CPU、メモリ、ハードディスク、ビデオカード、ネットワークカード

- 製品の再利用などが容易。
- 製品を入れ替えるだけで性能を上げられる。

- プログラミングもモジュール化する事で同じようなメリット

プログラムにおけるモジュール設計

- プログラムの全体の機能を機能単位で分解しそれぞれに独立した関数を用意して組み合わせることでプログラムを開発。
 - プログラム設計において、どのように機能を分割するかが非常に大事。
 - なるべく、個々のモジュールの依存関係がないように分割（疎結合）。
- プログラム言語におけるモジュール
 - C言語では関数、ファイル等

プログラムのモジュール化

- 生産性の向上
 - 各モジュールは独立しているので、独立に開発可能
- メンテナンス性の向上
 - プログラムが見やすくなる
 - モジュール毎のデバッグが可能
- モジュールの共有化
 - モジュールを別のプログラムでも再利用できる

DRYの原則：Don't Repeat Yourself

同じような機能のモジュールは作らずに、再利用する。

あるモジュールにバグがあったときに、そのモジュールだけを修正すればよい。

オブジェクト指向プログラミング (OOP: Object Oriented Program)

- OOP: モジュール化を推し進めたプログラミング。
 - プログラムを複数のオブジェクトに分割しオブジェクト間はメッセージをやり取りする。
 - オブジェクト：データと機能を持った実体。
 - OOPな言語：Ruby, C++, C#, Java, Python, Perl, etc...

OOPのイメージ



- データ
 - 名前
 - HP
 - MP
 - 攻撃力
 - 防御力
- 関数
 - 攻撃
 - 魔法
 - ダメージ



- データ
 - 名前
 - HP
 - 攻撃力
 - 防御力
- 関数
 - 攻撃
 - 防御
 - 逃げる
 - ダメージ

- 攻撃関数はある値を返す
 - `Int 攻撃(){return 攻撃力/10;}`
- ダメージ関数は引数の値を自分のHPから引く
 - `Void ダメージ(int val){HP -= val}`

勇者がスライムを攻撃するには

スライム.ダメージ(勇者.攻撃())

スライムが自分の関数“ダメージ”を呼ぶ。引数は勇者の攻撃。

Rubyの特徴

- 純粹オブジェクト指向言語
 - あらゆるものがオブジェクト
 - 整数も浮動小数点も文字列も全部オブジェクト
 - 例えば、整数型のオブジェクトはある整数と整数に関する関数を持つ。
 - `100.to_s` (数字の100を文字列"100"に変換する)。
- インタープリタ型言語
 - コンパイルの必要なし
- 型推論、動的型付け
- 同じ機能を実現するのに色々な書き方ができる。
 - Pythonと逆
 - 例)繰り返し処理
 - Ruby: `for`, `while`, `until`, `loop`, `times`, `each`, `step`,
 - Python: `for`, `while`くらい

Rubyの参考書



試し読み

たのしい Ruby 第6版

わかりやすくてNo.1! Ruby入門書の決定版。

書店販売部数第1位のRuby本

本書は初心者の方の気持ちを裏切らない出来になっています

Rubyの文法をわかりやすく解説

100 Creative

たのしいRuby 第6版 (Informatics&IDEA) (日本語) 単行本 - 2019/3/19

高橋 征義 (著), 後藤 裕蔵 (著)

★★★★☆ 13個の評価

> その他 の形式およびエディションを表示する

Kindle版 (電子書籍)	単行本
¥2,717	¥2,860
獲得ポイント: 55pt	獲得ポイント: 58pt

今すぐお読みいただけます: **無料アプリ**

¥2,065 より 20 中古品
¥2,860 より 19 新品
¥5,916 より 1 コレクター商品

お届け日: **11月8日 - 9日** 詳細を見る

初版から17年。改訂とともに変化しながら、ずっと読まれ続けている定番書です。第6版では、最新のバージョンに対応。

プログラミング初心者でも読み解けるように、チュートリアル、基礎、クラス、実践とテーマを切り分けて、平易に解説。Rubyの基礎から応用までがわかる一冊。続きを読む

著者をフォロー

高橋 征義 + フォロー

キャンペーンおよび追加情報

- 入会特典をこの商品に利用した場合、2,860円が**860円**に！ **Amazonクラシックカード** 新規ご入会で、**2,000円分** のAmazonポイントプレゼント。 **今すぐチェック**

よく一緒に購入されている商品

シェアする

新品

¥2,860

ポイント: 58pt (2%)

詳細はこちら

通常配送無料 詳細

残り10点 (入荷予定あり)

Kindle版は今すぐお読みいただけます。在庫状況についてこの商品は、Amazon.co.jp が販売、発送します。

数量: 1

☐ この注文でお急ぎ便、お届け日時指定便を無料体験 **Amazonプライム無料体験について**

カートに入れる

今すぐ買う

お届け先を選択

中古品

¥2,198

ほしい物リストに追加する

この商品をお持ちですか？

マーケットプレイスに出品する

- 分かりやすい。
- わりとなんでも書いてある。
- 値段もそんなに高くない。
- 図書館にも数冊ある。
 - rubyはVer1.9で結構変わったので、Ver1.9以降対応のものがよい。

Rubyの実行方法1/3

- irb(Interactive Ruby)を使う
 - ターミナルを開き、irbとタイプ。
 - Hello Worldを出力するコードは以下。

```
masaki@Ubuntu18: ~/lecture/matsue/2020/2020_oss2/lec01$ irb
irb(main):001:0> print "Hello World\n"
Hello World
=> nil
irb(main):002:0> print "Hello", " ", "World\n"
Hello World
=> nil
irb(main):003:0>
```

- 終了は、exit もしくは quit
- 簡単なRubyコードのテストやちょっとした計算に便利。
- 大規模なプログラム開発には向かない

Rubyの実行方法2/3

- viなどで適当なファイル(hello00.rb)を開く。
- 以下のように記述

```
=begin
hello00.rb
2020/11/07 M. Iwasawa
=end
print "Hello World\n"
print "Hello", " ", " World\n"
```

- =beginから=endで囲まれた部分はコメントアウトされる。
(C言語の'/*'、'*/'と同じ。)

- 以下のコマンドで実行
 - \$ ruby hello.rb

Rubyの実行方法3/3

- 適当なファイル(hello01.rb)を開き、以下のように記述

```
#!/usr/bin/ruby
=begin
hello01.rb
2020/11/07 M. Iwasawa
=end
print "Hello World\n"
print "Hello", " ", "World\n"
```

行頭に'#!/usr/bin/ruby'を付ける

- 実行権限を付与
 - \$ chmod u+x hello2.rb
- 以下のコマンドで実行
 - \$./hello2.rb

まとめ

- 設計方法
 - インテグラル設計(すり合わせ設計)：車
 - 各部品をすり合わせて高性能化する。
 - モジュール設計：PCシステム、PC自体
 - 生産性の向上
 - メンテナンス性の向上
 - モジュールの共有化
- オブジェクト指向プログラミング(OOP)
- Rubyの実行方法
 - irb
 - rubyコマンドでファイルから実行