

付録A 練習問題確認プログラム

教科書にある練習問題の解答が正しいかどうかを確認するプログラム `check.rb` の使い方を説明します。

注意: この確認プログラムは、定義された関数を「いくつかの値について計算させてみて、答えが合っているか確かめる」だけのものです。この検査を通過したからといって、関数の定義が正しいことは保証されません。

A.1 確認プログラムのダウンロード

まず、教科書のサポートページ

<http://lecture.ecc.u-tokyo.ac.jp/johzu/joho-kagaku/text/>

から配布プログラム `check.rb` をダウンロードしておきます。ダウンロードしたファイルは、練習問題を解答するプログラムと同じディレクトリ/フォルダに保存して下さい。

A.2 練習問題の解答を作成

教科書の練習問題には関数定義を書くファイル名が指定されています。例えば練習 1.9(a) (p.21) では関数 `celsius_to_fahrenheit(c)` をファイル `yardpound.rb` に作るよう指定されています。これに従って解答を作して下さい。

解答を作ったら `irb` の中で `load` して、正しい計算をすることを自分で確認しておきましょう。

A.3 確認データファイルをダウンロード

確認データファイルを `check.rb` や解答のファイルと同じディレクトリ/フォルダにダウンロードします。

授業で使用する場合のダウンロード元は、担当の教員の指示に従って下さい。

自習の場合は、教科書のサポートページに各章ごとの確認データファイルがありますので、それをダウンロードして使うことができます。

A.4 確認プログラムの実行

以下、確認データファイルが `ex01.rb` という名前だったとします。

ターミナルウィンドウの中で、以下の命令を入力して定義した関数を検査します。 `irb` の外で入力する必要があることに注意して下さい。

```
1 cm12345$ ruby check.rb ex01.rb
```

確認プログラムは解答ファイル中の関数定義を 1 つずつ検査して以下のようなメッセージを表示します。

```
1 テストデータファイルとして ex01.rb を使います...
2
3 ファイル distance.rb を読み込みます...
4 Ex1.2 a    [対象のメソッド: distance      ] => PASS
5
6 ファイル yardpound.rb を読み込みます...
7 Ex1.2 b    [対象のメソッド: feet_to_cm    ] => PASS
8 Ex1.2 c    [対象のメソッド: pound_to_kg   ] => NG (計
   算式: pound_to_kg(0.1,1)期待した値: 0.0737, 実際の値:
   0.04536)
9 Ex1.9 a    [対象のメソッド: celsius_to_fahrenheit] =>
   ERROR: 対象のメソッドが見つかりません
10 (略)
11
12 ファイル bmi_yp.rb を読み込みます...
13 ERROR: ファイル bmi_yp.rb が見つかりません。
14     ファイル名が異なっているかもしれません。
15     違うフォルダに存在するかもしれません。
16 Ex1.10     [対象のメソッド: bmi_yp      ] => ERROR:
   対象のメソッドが見つかりません
17 (略)
18
19 成功数: 12/13
20 (未定義で検査できなかった個数: 1)
```

それぞれのメッセージは次のようなことを表わしています。

- (4行目) 練習問題 1.2(a) の `distance` は、検査に合格した
- (8行目) 練習問題 1.2(c) の `pound_to_kg` は、期待した答えを返さなかった
- (10行目) 練習問題 1.9(a) の `celsius_to_fahrenheit` は `yardpound.rb` の中に定義されていなかった

- (15 行目) 練習問題 1.10 の `bmi_yp` が定義されているはずの `bmi_yp.rb` というファイルがなかった

これらのメッセージを参考に、間違いを直してゆきましょう。なお、間違いを修正する場合は、`irb` の中で関数を計算させる方がよいでしょう。その方がより詳しい情報が表示されます。

A.5 提出ファイルの作成

授業担当者から「確認プログラムを実行して解答をまとめたファイルを作り、提出せよ」と指示された場合は、次のようにして提出ファイルを作成します。

```
1 cm12345$ ruby check.rb --report ex01.rb
```

マイナス記号が 2 つ並んでいることに注意して下さい。これを実行すると、上と同様のメッセージが表示された後、一番最後の行に

```
1 提出用ファイル ex01-091021-204855.tgz ができました
```

と表示され、提出用ファイルが作られます。このファイルには、検査に使われた解答ファイルがまとめられているので、これを授業担当者に指示された方法で提出します。

なお、解答ファイルの中で、別のファイルを `load` していた場合、そのファイルも一緒にまとめられるようになっています。

A.6 まとめ

1. 配布プログラムページから `check.rb` をダウンロードする
2. 練習問題の解答を作成する
3. (指示された場所から) 確認データファイルをダウンロードする (例えば `ex01.rb` という名前だとします)
4. `ruby check.rb ex01.rb` を実行して検査する
5. `ruby check.rb --report ex01.rb` を実行して提出ファイルを作る

A.7 補足: Microsoft Windows 上で使う場合

Microsoft Windows で確認プログラム `check.rb` を使う場合には、`tar` というファイルアーカイブプログラムがインストールされている必要があります。tar プログラムは以下のような場所から入手できます。

また、tar プログラム (`tar.exe`) が置かれているディレクトリが、環境変数 `Path` の値に含まれている必要もあります。そのような設定されていることは、「コマンドプロンプト」の中で `tar` と入力して確認して下さい。正しく設定されていれば tar プログラムが起動するはずですが。

- Tar for Windows (<http://gnuwin32.sourceforge.net/packages/gtar.htm>)
- Cygwin (<http://www.gnu.org/software/tar/>)