プログラミング演習I（平成30年度）
– 第1回　レポート課題 –

担当: 時井，松村

提出日11月12日 (月) 15時

課題1

歩きながらモンスターを捕まえるゲームを作る．Enter キーを押すことで歩き回り，モンスターに遭遇すると以下のメニューを表示するプログラム（sXXXXXXX-1_2018.rb）を提供するので，メニュー選択にしたがった機能を追加してプログラムを完成させる．

1. 投げる
2. 逃げる
3. 表示する
4. 最強はどれ？
5. 今日はもう寝る（終了）

各メニュー番号をキーボードから入力した時の機能とそれを実現する要件は以下とする．

1. 投げるボールがあるか確認し，ボールがなければ「もうボールがありません」と表示して，プログラムを終了する．ボールがあれば，ボールを投げてモンスターの捕獲を試みる．

   「ボールを投げてモンスターの捕獲を試みる」部分をメソッド throw として定義する．(後述)

   throw の**返り値**に応じて，捕獲できた場合には，捕まえた旨のメッセージを表示して，配列 monster_bag に名前を追加する．捕獲できなかった場合には，逃げられた旨のメッセージを表示する．ボールを投げた場合はボールの数を減らして，残りボール数を表示する．

   メソッド throw の**引数**はモンスターの強さを表す数値 cp とする．**引数**で指定されたモンスターの強さと，0 から 2000 までの擬似乱数とを比較して，モンスターの方の数値が小さければ捕獲できたとして true を**返り値**とし，モンスターの方が大きければ逃げられたとして false を**返り値**とする．

   なお，擬似乱数は rand メソッドを使う．例えば，さいころのような 1 から 6 までの疑似乱数（ランダムな数）を得るには，rand(1..6) とすればよい．

2. 何もしない．

3. 既に捕まえたモンスターを強さとともに表示する．この部分はメソッド display_list とする．**引数**は捕まえたモンスター名の入った配列 monster_bag とモンスターの強さの入ったハッシュcp_list の 2 つとし，**返り値**はなしとする．メソッド内の処理は，

   - 捕まえたモンスター数の表示
   - 各モンスターについて，番号，名前，強さ，の表示

   とする．

4. 1番強いモンスターとその強さを表示する．この部分はメソッド strongest とし，**引数**は monster_bag と cp_list の 2 つとし，**返り値**はなしとする．

5. 「おやすみなさい」と表示したあとプログラムを終了する．

[**実行の指定**]　実行結果は以下の順に操作する．

1. 作成したプログラムで，メソッド walk を呼び出している部分を**コメントアウトする**
2. 「1. 投げる」と「2. 逃げる」をそれぞれ 1 回以上使って，異なるモンスターを 3 匹捕まえる．(なお，3 匹捕まえる前にボールが尽きた場合は，その結果は破棄し，やり直すこと）
3. 「3. 表示する」で一覧表示する
4. 「4. 最強はどれ？」で一覧表示する
5. 「5. 今日はもう寝る（終了)」で終了する

（注）どのモンスターを捕まえたかは成績には影響しない．

[**実行結果表示例**]　以下は実行結果の一例である．レアなやつが現れて思わず余計な実行をしてしまっているが，実行結果の改ざんは良くないのでそのまま掲載している．あまりに長大な実行結果にならないように注意するように．

```
Z:\progI> ruby sXXXXXXX-1.rb
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
ダンバル (CP735) があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1.投げる
2.逃げる
3.表示する
4.最強はどれ？
5.今日はもう寝る（終了）
1
逃げられた！
ボールは残り 9 個です

■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
フワンテ (CP55) があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1.投げる
2.逃げる
3.表示する
4.最強はどれ？
5.今日はもう寝る（終了）
1
やった！
フワンテ (CP55) をつかまえた！
ボールは残り 8 個です

■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
```

カゲボウズ（CP520）があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1．投げる
2．逃げる
3．表示する
4．最強はどれ？
5．今日はもう寝る（終了）
2

■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
ハロウィンピカチュウ（CP10）があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1．投げる
2．逃げる
3．表示する
4．最強はどれ？
5．今日はもう寝る（終了）
1
やった！
ハロウィンピカチュウ（CP10）をつかまえた！
ボールは残り 7 個です

■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
アチャモ（CP640）があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1．投げる
2．逃げる
3．表示する
4．最強はどれ？
5．今日はもう寝る（終了）
1
やった！
アチャモ（CP640）をつかまえた！
ボールは残り 6 個です

■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
がまじゃんぱー（CP1980）があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1．投げる
2．逃げる
3．表示する
4．最強はどれ？
5．今日はもう寝る（終了）
1
逃げられた！
ボールは残り 5 個です

```
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
ダンバル (CP735) があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1. 投げる
2. 逃げる
3. 表示する
4. 最強はどれ？
5. 今日はもう寝る（終了）
3
つかまえたモンスターの数は 3
1: フワンテ (CP55)
2: ハロウィンピカチュウ (CP10)
3: アチャモ (CP640)

■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
カゲボウズ (CP520) があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1. 投げる
2. 逃げる
3. 表示する
4. 最強はどれ？
5. 今日はもう寝る（終了）
4
最強はアチャモ（CP640)！

■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
カゲボウズ (CP520) があらわれた！
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
1. 投げる
2. 逃げる
3. 表示する
4. 最強はどれ？
5. 今日はもう寝る（終了）
5
おやすみなさい

Z:\progI>
```

[**プログラムの説明とコメント**]　課題 1 に関しては，自分で書いた以下の部分のみの説明をするだけでよい．

- メソッド `throw`
- メソッド `display_list`
- メソッド `strongest`
- メニューを選択したあとの処理

ただし，プログラム中のコメントは適切に書くこと

課題 2

降車駅名をキーボードから入力すると新橋から降車駅までの駅名，所要時間，周辺施設を表示するプログラムを作成する．降車駅が何番目にあるかを保存した変数、駅名配列、所要時間配列、周辺施設配列を引数として渡すと降車駅名までの駅名一覧と所要時間を表示するメソッドを定義し使用すること．(授業ページにコメントのみのテンプレート `sXXXXXXX-2_2018.rb` を掲載したので構成を確認すること．)

[**プログラムの条件**]

- 駅名，所要時間と周辺施設は下記の条件を満たすように配列を使ってデータを保存すること．
  - 駅名を保存する配列に関して
    - ゆりかもめの停車駅を順に保存する配列名は，`eki` とする．
    - 新橋，汐留，竹芝，....，豊洲の順に保存する．
  - 所要時間を保存する配列に関して
    - ゆりかもめの各駅間の所要時間（分）を保存する配列を準備する．
    - 所要時間を順に保存する配列名は， `jikan` とする．
    - 新橋と汐留, ... 市場前と新豊洲，新豊洲と豊洲の順に所要時間を保存する．
  - 周辺施設を保存する配列に関して
    - 各駅周辺の周辺施設を保存する配列名は，`shisetu` とする．
    - 新橋から順に各駅の周辺施設を保存する．
- 降車駅名をキーボードから入力すると，まずは降車駅は配列の何番目にあるか判定し，何番目にあったかを変数に保存する。「降車駅は，駅名配列 eki の*番目にあります」と表示し，駅一覧等を表示するメソッド hyouji を呼ぶ．駅名がない場合は,「入力された駅名は，駅名配列 eki にありません」と表示する．
- メソッド hyouji に関して
  引数として，降車駅が何番目にあるかを保存した変数、駅名配列、所要時間配列、周辺施設配列を受け取る
  - 表示結果は，例に示したように，まずは，新橋から降車駅までの駅名リストを表示する．つぎに，所要時間を表示する．最後に，駅名と周辺施設のリストを表示する．
- 実行例として，降車駅名に，豊洲，つくば，新橋，台場をこの順番で指定すること．
- 駅名，所要時間と周辺施設は下記のデータを使うこと (授業ページに掲載)
  - 所要時間 (分) 新橋 ↔ 汐留, 汐留 ↔ 竹芝, ... 新豊洲 ↔ 豊洲の順
- 降車駅名をキーボードから入力し変数に保存できるようにするためにプログラムの 1 行目の `#encoding` 文は下記のように記述してください．

```
#encoding: Windows-31J
```

[**実行結果表示例**]

```
Z:\progI>ruby sXXXXXXX-2.rb
*** 降車駅名を入力してください:豊洲
降車駅は，駅名配列 eki の 15 番目にあります
---------------------------------------
新橋=>汐留=>竹芝=>日の出=>芝浦ふ頭=>お台場海浜公園=>台場=>船の科学館=>テレコムセンター=>青海=>
国際展示場正門=>有明
>有明テニスの森=>市場前=>新豊洲=>豊洲
所要時間:24 分
----- 沿線情報を表示します（駅名 [名所など]）-----
0 新橋 [汐留シオサイト]====>
1 汐留 [浜離宮恩賜庭園]====>
2 竹芝 [竹芝客船ターミナル]====>
3 日の出 [水上バス乗場]====>
4 芝浦ふ頭 [レインボーブリッジ]====>
5 お台場海浜公園 [虹の下水道館]====>
6 台場 [メディアージュ]====>
7 船の科学館 [東京湾岸警察署]====>
8 テレコムセンター [日本科学未来館]====>
9 青海 [パレットタウン]====>
10 国際展示場正門 [東京ビックサイト]====>
11 有明 [パナソニックセンター東京]====>
12 有明テニスの森 [有明コロシアム]====>
13 市場前 [中央卸売市場]====>
14 新豊洲 [MAGIC BEACH]====>
15 豊洲 [がすてなーにガスの科学館]

Z:\progI>ruby sXXXXXXX-2.rb
*** 降車駅名を入力してください:つくば
入力された駅名は，駅配列 eki にありません

Z:\progI>ruby sXXXXXXX-2.rb
*** 降車駅名を入力してください:新橋
降車駅は，駅名配列 eki の 0 番目にあります
---------------------------------------
新橋
所要時間:0 分
----- 沿線情報を表示します（駅名 [名所など]）-----
0 新橋 [汐留シオサイト]

Z:\progI>ruby sXXXXXXX-2.rb
*** 降車駅名を入力してください:台場
降車駅は，駅名配列 eki の 6 番目にあります
---------------------------------------
新橋=>汐留=>竹芝=>日の出=>芝浦ふ頭=>お台場海浜公園=>台場
所要時間:11 分
----- 沿線情報を表示します（駅名 [名所など]）-----
```

```
0 新橋 [汐留シオサイト]====>
1 汐留 [浜離宮恩賜庭園]====>
2 竹芝 [竹芝客船ターミナル]====>
3 日の出 [水上バス乗場]====>
4 芝浦ふ頭 [レインボーブリッジ]====>
5 お台場海浜公園 [虹の下水道館]====>
6 台場 [メディアージュ]

Z:\progI>
```

プログラム作成にあたって

最初からメソッドを作成するのは難しいなと感じた方は，
次のような順にプログラムを組み立ててみるとよい．

step1 入力された駅名が配列に含まれるか判定する．
step2 降車駅は何番目かを判定する．その値を変数に保存する．
sete3 駅名一覧を表示する．
step4 降車駅までの駅名と周辺施設を表示する．
　　所要時間を計算し，表示する．
step5 メソッドにする部分を考えて，作成する．

さらに上を目指す方

乗車駅も入力できるようにする場合は，メソッドの引数に乗車駅名が配列の何番目にあるかを保存した変数を保存した変数を追加してよい．

提出における注意

- 締切：締切を厳守すること．締切に遅れたレポートは受理しない．
- 提出物：紙，プログラム
- 提出場所：(紙）学務課レポート提出用ポスト,（プログラム）progI-ml@klis.tsukuba.ac.jp
  自分の受講クラスのポストに入れること．他のクラスのポストに入れた場合は受理しない．
- プログラムについて
  - 作成したプログラム ***.rb を電子メールに添付して送付する．
  - ファイル名は，ユーザ名-課題番号.rb とする．すなわち，sXXXXXXX-1.rb，sXXXXXXX-2.rb の二つである．
  - 宛先は，progI-ml@klis.tsukuba.ac.jp
  - 件名は，rep1
- 紙について以下を遵守しないレポートは減点の対象となる．
  - 設問毎に A4 用紙にまとめ左上をステープラ（ホチキス）でとめる．(今回は設問が 2 個あるので，合計 2 部提出)
  - 設問毎に表紙，本文の順でとめる．
    - 表紙： 科目名，曜日クラス，第 1 回レポート　設問番号，提出日，学籍番号，氏名を記入する．見本を Web ページに掲載するので，その形式にしたがうこと．
    - 本文： 行番号付きプログラムリスト，実行結果とプログラムの説明をこの順で載せる．
      プログラムリストと実行結果は続けて記述しても良いが，プログラムの説明は別ページとすること．
      いずれも機械出力とする．その際，プログラムリストと実行結果は，等幅フォントとする．
      出力したプログラムや実行結果に手書きしないこと．
      なお，付録に形式の一例を載せたので参考にしてほしい．
  - 実行結果に関する注意事項
    - 実行は，全て全学計算機システムの Windows 上で行うこと．
    - 実行 (例 ruby ***.rb)，出力される実行結果の順となるように，一連の操作の出力を記述する．(注意：余分な操作を途中で入れないこと）
    - 実行結果は必要に応じて複数示すこと．
  - 可読性が良くなるよう努めること.「プログラム」は適切な字下げと空行の挿入をし,「説明」は適切な見出しを付けるなど，構造がわかるように注意すること．
  - 片面印刷にすること．

その他の注意

- 同一/類似レポートは「両成敗」．すなわち，見た方も見せた方も不正行為とみなし，単位を出さない．また，他の科目にも影響することもある．他人が不正にプログラムにアクセスできないよう，各自の責任において対処すること．
- レポートの差し替えや再提出は認めないので，内容をよく確認してから提出すること．
- レポートを書く上での一般常識として，参考文献がある場合には書誌情報を載せること．

- フォーマットが細かく設定されているのは決して意地悪からではない．100 名を越える受講者のレポートを効率的かつ公平に採点するために必要なことだからである．

  言い替えれば，フォーマットを無視したレポートは演習担当者の作業を著しく妨げることになる点を理解してほしい．

(付録) 行番号付きプログラム等の形式の例

以下は一例であるので，各自で考えてより読みやすい形式に整えること．

設問 1：球の体積計算プログラムを作成する．

```
----------------------------------------
プログラムリスト
----------------------------------------
 1  print("球の半径を入力してください\n")
 2  r = gets.chomp.to_f
 3  v =
 4
 ......................

----------------------------------------
実行結果
----------------------------------------
Z:\> ruby s9911999-1.rb
 ......................
 ......................

Z:\> ruby s9911999-1.rb
 ......................
 ......................
```

```
----------------------------------------
プログラムの説明
----------------------------------------
 ......................
......................
......................
```