

プログラミング演習 I (平成 22 年度)

– 第 1 回 レポート課題 –

担当: 時井, 松村

提出日 10月20日(水) 15時

1 課題 (設問 1)

つくばエクスプレスの出発駅と到着駅の駅名を入力すると出発駅から到着駅までに通過する駅名と各駅間の所要時間, さらに, 総所要時間を表示するプログラムを作成する。(下り(秋葉原 から つくば)のみ対応でよい)

[プログラム作成上の条件]

- 配列に関する条件
つくばエクスプレス(以後 TX と略)の駅名と駅間の所要時間(分)をそれぞれ eki, jikan という配列に保存する。
配列 eki の要素として, 秋葉原から順につくばまでの駅名を保存する。配列 jikan の要素として, 0 番目の要素に秋葉原と新御徒町の所要時間 2 を保存し, 最後の要素は 0 を保存することとする。
- 駅名の表示
繰り返し文と条件判断文を使い記述すること
- 総所要時間の計算
入力された出発駅と到着駅から配列 jikan の値を使って, プログラム中で計算し表示すること

[駅名と所要時間のデータ]

[駅名]

秋葉原, 新御徒町, 浅草, 南千住, 北千住, 青井, 六町, 八潮, 三郷中央, 南流山,
流山セントラルパーク, 流山おおたかの森, 柏の葉キャンパス, 柏たなか, 守谷, みらい平, みどりの, 万博記念公園, 研究学園, つくば

[所要時間] (秋葉原<->新御徒町, 新御徒町<->浅草 の順)

2, 2, 3, 3, 3, 2, 4, 3, 3, 3, 2, 3, 3, 4, 5, 3, 3, 3, 3

[実行結果 (表示例)]

```
D:\>ruby -Ks eki.rb
```

(注意) TX 下りのみ対応しています

TX 何駅から:つくば

TX 何駅まで:守谷
所要時間と路線図を表示することができません

D:\>ruby -Ks eki.rb

(注意) TX 下りのみ対応しています

TX 何駅から:守谷

TX 何駅まで:つくば

+---

[守谷]

|

| 5分

|

[みらい平]

|

| 3分

|

[みどりの]

|

| 3分

|

[万博記念公園]

|

| 3分

|

[研究学園]

|

| 3分

|

[つくば]

+---

[守谷] から [つくば] まで 17 分

オプション (加点対象項目)

*下り, 上り両方に対応できている場合は, 加点対象とする .

ヒント

下記のステップでプログラムを作成してみるとよい

- 秋葉原駅からつくば駅までの路線図を表示
- 総所要時間を計算し表示
- 駅名を入力できるようにし, その駅名が配列の何番目の要素であるか表示
- 入力された駅名間の路線図を表示

2 課題（設問2）

テキスト（プログラミング演習IのページでPDFを提供しているもの）に出てきたキーワードの索引を作成するプログラムを作成する。

プログラムの条件

- 索引は、キーワードをキーとし、出現位置を値とするハッシュ（ハッシュ名 `word_index`）で作成する。
- 出現位置は、演習回数-ページ番号という形式の文字列として扱う。例えば、「第3回演習のテキストの5ページ」の場合、3-5 という文字列として扱う。
- キーワード、演習回数、ページ番号の3つをキーボードから入力できるようにする。
- 入力は `while` を使った繰り返し文で記述し、キーワードは何個でも入力可能とする。
- キーワードの入力の終了条件は.(ピリオド)とする。
- 既にハッシュに存在するキーワードを指定したときは、上書きするかそのままにするかを選択できるようにする。
- 入力が終わったら、全ての索引を表示する。出現位置は、演習回数-ページ番号の形式のままとする。
- それぞれの操作に対して、適切なメッセージを表示する。

実行結果

- 以下のキーワードについて、出現箇所を調べる。

キーワード	演習回	ページ
改行		
print		
算術演算		
変数		
配列		
ハッシュ		
整数		
メソッド		

- 上記のキーワードを全て使って実行結果を示す。
- ただし、「プログラムの条件」を満たしているかどうかを確認できる実行結果となるようにする。

オプション（加対象項目）

- 最後の全ての索引の表示をキーワードのABCあうえお順にする。
- キーワードの出現位置を複数扱えるようにする（ヒント：ハッシュの値を配列にする）
- 索引の表示後にキーワードで検索できるようにする。
- 最後の全ての索引の表示を演習回毎に整理する。

などなど ...

3 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。
- 提出場所：学務課レポート提出用ポスト
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。

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ブラ（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 x 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。

4 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 rep1.rb
```

```
.....
.....
```

```
Z:\> ruby rep1.rb
```

```
.....
.....
```

プログラムの説明

```
.....
.....
.....
```