

## 2009年度プログラミング言語III(計算機言語II) 定期試験

以下の問はすべてRuby言語を用いて答えよ。

### I. Rubyプログラム

(1) 正方行列(行数と列数が同じ行列)

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

の対角要素の和  $a_{11} + a_{22} + \dots + a_{nn}$  をトレースと言う。下記のように、正方行列を配列で表現するとき、引数で指定された配列のトレースを計算するメソッド `trace` を実装せよ。行数と列数が同じであるかどうかは検査しなくてよい。(15点)

```
matrix = [[1, 1, 1], [1, 1, 1], [1, 1, 1]] # すべて要素が1である3行3列の行列を表す
trace matrix # 評価値は3
```

(2) 英文字のみで書かれたテキストファイル `data.txt` の中に含まれる文字数(空白や改行も含む)、単語数、行数を出力するプログラム `wc.rb` について、下記の未完成部分を完成させよ。`data.txt` 中の単語は必ず空白1つで区切られているとする。またファイルの各行の終わりには必ず改行があるとする。(20点)

```
# cat はファイルの中身を表示するUNIXコマンド
# 実行例は以下の通り

cat data.txt
This is the meaning of our
liberty and our creed
ruby wc.rb
49 10 2
# 出力の意味は 文字数 単語数 行数。間はそれぞれ空白1つ
```

`wc.rb`は次のようになる。

```
#!/usr/bin/ruby
chars = 0 # 文字数
words = 0 # 単語数
lines = 0 # 行数
# 以下未完成
```

## II. 正規表現

次のテキストファイルがあるとする。各行の左端の番号は行番号を表す。

```
1 <html xmlns="http://www.w3.org/xhtml1">
2 <head><title>My homepage</title></head>
3 <body>
4 <h1>My homepage</h1>
5 Office: <a href="http://www.oka-pu.ac.jp/">Okayama Pref. University</a>.
6 My lecture is as follows:
7 <ul>
8 <li>Programming Language II</li>
9 <li>Programming Language III</li>
10 <li>Compiler</li>
11 <li>Formal Language Theory</li>
12 </ul>
13 </body>
14 </html>
```

(1) 正規表現 `/¥S+/ にマッチする行の行番号を示せ。(10点)`

(2) 正規表現 `/<(¥w+)¥s+(¥w+)=/ にマッチする行の行番号を示し、それぞれについて変数 $1, $2 の値がどう`  
なるか示せ。(15点)

### III. Webアプリケーション

www.example.comというコンピュータ上のディレクトリ /home/user に、以下のRubyプログラム webserver.rb を置き、実行させた。このとき以下の問いに答えよ。

```
#!/usr/bin/ruby
require 'webrick'
include WEBrick

s = HTTPServer.new(
  :DocumentRoot => File.join(Dir::pwd, "www")
)
s.mount_proc("/req") { |req, res|
  res.body = "#{req.path}"
}
s.mount_proc("/plus") { |req, res|
  total = 0 # 合計値
  # 未完成
  res.body = total.to_s
}
trap("INT") {s.shutdown}
s.start
```

(1) `http://www.example.com/doc/index.html` にアクセスしたときに表示されるHTMLファイルは、サーバ上のどこに置けば良いか。絶対パスで答えよ。(10点) HTMLファイルの絶対パスを答える -Takeo Kunishima 10/02/10 14:48

(2) `http://www.example.com/req/what/is/response` にWWWブラウザでアクセスするとき、`www.example.com` から返ってくるHTTPステータスコードとレスポンスボディを示せ。(10点)

(3) 上記プログラムの未完成部分は、リクエストURL中の数字をすべて足し合わせた結果をレスポンスボディとして返すプログラムとしたい。例えば、`http://www.example.com/plus/29/2/49/32` にアクセスすると、 $29 + 2 + 49 + 32 = 112$  を返したい。未完成部分を完成させよ。(20点) URLは `http://www.example.com/plus/整数/整数/整数/ ...` という形であると仮定してよい。 -Takeo Kunishima 10/02/10 14:48

## 解答例

### I. Rubyプログラム

(1)

```
def trace matrix
  total = 0
  matrix.each_with_index { |item, index|
    total += item[index]
  }
  total
end
```

while式で書くなら、次のようになります。

```
def trace matrix
  total = i = 0
  while i < matrix.length
    total += matrix[i][i]
    i += 1
  end
  total
end
```

(2)

```
File.open ("data.txt") { |f|
  f.each_line { |line|
    lines += 1
    words += line.split(/ /).length
    chars += line.length
  }
  puts "#{chars} #{words} #{lines}"
}
```

### II. 正規表現

(1) 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14  
/^¥S+\$/ だと 3, 7, 10, 12, 13, 14 になります。

(2)

- ・ 1行目: \$1 = "html", \$2 = "xmlns"
- ・ 5行目: \$1 = "a", \$2 = "href"

### III. Webアプリケーション

(1) /home/user/www/doc/index.html

(2) HTTPステータスコード: 200, レスポンスボディ: /req/what/is/response

(3)

```
req.path.split(/¥//).each { |i|
  total += i.to_i if i != "plus"
}
```

ただし、Stringクラスのメソッド `to_i` は、整数への変換に失敗すると0を返します。

```
"/req/plus/29/2/49/32".split(/¥//)    #=> ["", "plus", "29", "2", "49", "32"]
"".to_i                                #=> 0
"plus".to_i                             #=> 0
```

したがって、実際には、以下でも正しく動作します。このように、“plus” や “” についても合計してしまっている  
答案も、減点はしていません。

```
req.path.split(/¥//).each { |i| total += i.to_i }
```