

電子計算機入門

池田, 大輔
九州大学情報基盤センター

<http://hdl.handle.net/2324/6097>

出版情報 : 2003
バージョン :
権利関係 :



電子計算機入門 第6回

池田 大輔

daisuke@cc.kyushu-u.ac.jp

情報基盤センター

目次

- 前回課題の回答例
- while 文
- ファイルの利用
- DDBJ データベース

前回課題の解答例

- 最大共通部分文字列を解くアルゴリズムを書け

前回課題の解答例

- 最大共通部分文字列を解くアルゴリズムを書け

2つの入力引数を変数 $str1$ と $str2$ に代入

$str1$ のすべての部分文字列を配列 $ss1$ に代入

結果を格納する空配列 res

$0 \leq i < j \leq len(str2)$ なる i, j に対し

$len(str2[i : j + 1]) < len(res[0])$ なら何もしない

$len(str2[i : j + 1]) == len(res[0])$ なら

res に $str2[i : j + 1]$ を追加

$len(str2[i : j + 1]) > len(res[0])$ なら

$res = [str2[i : j + 1]]$

既存のスキ립トの実行

- すでに作ったプログラム (スキ립トファイル) は何度でも利用可能
 - 1 [File] → [Open] でプログラムを開く
 - 2 [File] → [Run] で実行

while 文

- 反復 (ループ) を制御する文
 - for 文は配列の要素に対する反復
 - while 文はより一般的で、対象が配列とは限らない

while 文

■ 反復 (ループ) を制御する文

- for 文は配列の要素に対する反復
- while 文はより一般的で、対象が配列とは限らない

■ 構文

```
while test: # コロンに注意
```

```
    # 処理範囲はインデントで
```

```
        if test: break # ループを終了
```

```
        if test:
```

```
            continue # 「他の処理」をせず次のループへ
```

```
        # 他の処理
```

```
else: # コロンに注意
```

```
    # break をしなかった場合の処理
```


while 文:補足

- `break`, `continue` は何回出現してもよいし、使わなくてもよい
- `else` 以下は省略可能
- `else` が実行されるのは、`while` ループの実行が `break` によって中断されたときのみ

while 文の例: 無限ループ

- 何もせずループするだけ

```
while 1:  
    pass # 何もしない python の文
```

while 文の例: 無限ループ

- 何もせずループするだけ

```
while 1:  
    pass # 何もしない python の文
```

- 止めるときは Ctrl+Alt+Del からタスクマネージャを呼びだし pythonwin を終了させる

while 文の例: 配列の検索

- 配列の要素の中から特定の値を持つものを探す

```
i = 0 # 初期化
while i < len(array): # チェック
    if array[i] が探してたもの: break
    i += 1 # 更新
    # i = i+1 と同じ
else:
    print 'Not Found' # 見つからなかった場合
```

while 文の注意

- 一般に while 文は、変化させるもの (i など) を
 1. ループの前で初期化する
 2. ループ中で更新する
 3. ループごとに変化させるものが条件を満たすかチェックすることで繰り返しを実現する
- 初期化や、更新を忘れないように注意する

ファイル読み込み

- 引数 (arguments) からの入力では長い入力文字列や、すでにあるデータを効率よく利用できない
 - ⇒ プログラムからファイルを読み込み、入力とする

ファイル読み込み

- 引数 (arguments) からの入力では長い入力文字列や、すでにあるデータを効率よく利用できない

⇒ プログラムからファイルを読み込み、入力とする

- 関数 `open` (ファイル名、モード)

- ファイルにアクセスするためのモノ (オブジェクト) を返す
- モードは「読み込み ("r")」、「書き込み ("w")」、「追加 ("a")」などを指定する

- 書式

```
f = open('sample' , "r")
```

- “sample” というファイルを開く
- `open()` が返したオブジェクトを変数 `f` に格納

ファイル読み込みの例:1行

```
import sys
file = sys.argv[1]
f = open(file, "r") # 変数に格納したファイル名
line = f.readline()
```


ファイル読み込みの例:1行

```
import sys
```

```
file = sys.argv[1]
```

```
f = open(file, "r") # 変数に格納したファイル名
```

```
line = f.readline()
```

- オブジェクト (この場合は `f`) の後にピリオドをつけて、オブジェクトの機能を呼び出す

ファイル読み込みの例:1行

```
import sys
```

```
file = sys.argv[1]
```

```
f = open(file, "r") # 変数に格納したファイル名
```

```
line = f.readline()
```

- オブジェクト (この場合は `f`) の後にピリオドをつけて、オブジェクトの機能を呼び出す
- `readline()` はファイルから1行読み、文字列として返す
ファイルの終了時は空文字列 ("") を返す

ファイル読み込みの例:1行

```
import sys
file = sys.argv[1]
f = open(file, "r") # 変数に格納したファイル名
line = f.readline()
```

- オブジェクト (この場合は `f`) の後にピリオドをつけて、オブジェクトの機能呼び出す
- `readline()` はファイルから1行読み、文字列として返す
ファイルの終了時は空文字列 ("") を返す
- 他に、すべての行を一度に読みこみ、一行を一要素とする配列を返す `readlines()`

ファイルの与え方

- Pythonwin の引数としてファイル名を与える場合
ファイルが Z フォルダにある場合

ファイルの与え方

- Pythonwin の引数としてファイル名を与える場合
ファイルが Z フォルダにある場合
- 絶対パス
 - “Arguments” の欄に “Z:¥FILENAME” と入力する

ファイルの与え方

- Pythonwin の引数としてファイル名を与える場合
ファイルが Z フォルダにある場合
- 絶対パス
 - “Arguments” の欄に “Z:¥FILENAME” と入力する
- 相対パス
 - プログラムと読み込みたいファイルがある場所が同じ場合は “FILENAME” のみでよい

■ ファイル読み込みの例:全部の行

```
f = open(file , "r") # file にファイル名
lines = f.readlines() # 全部の行を配列に格納
for i in lines: # 1行ずつ処理
    print i, # カンマに注意
```

実習：ファイル読み込み

- 引数で与えたファイルの各行を表示するプログラムを `while` 文を使って書け

実習：ファイル読み込み

- 引数で与えたファイルの各行を表示するプログラムを `while` 文を使って書け

```
import sys
file = sys.argv[1]
f = open(file, "r")
line = f.readline() # 初期化
while line :
    print line,
    line = f.readline() # 更新
# ファイルの最後では空文字が返るので
# while が終了する
```

実習：ファイル読み込み

- さきほどの実習において `readline` を使わず `readlines` を使って書き直せ

実習：ファイル読み込み

- さきほどの実習において `readline` を使わず `readlines` を使って書き直せ

```
import sys
file = sys.argv[1]
f = open(file, "r")
for line in f.readlines(): # “s”を除いたら？
    print line
```

DDBJ

- 日本 DNA データバンクによる塩基配列データベース

<http://www.ddbj.nig.ac.jp/Welcome-j.html>

- DDBJ が提供する XML 形式のデータベースを利用する
XML Central of DDBJ

<http://www.xml.nig.ac.jp/index.html>

- 実際の例

実習：頻度カウント

- 配布するシーケンスから長さ5の部分シーケンスを直接目で数え、もっとも頻度の高いものを調べなさい
 - シーケンスは `<SEQUENCE>` と `</SEQUENCE>` に囲まれた部分
- これをプログラムにより自動的に行なわせるときに、どのような変数(データの形式)が必要か考えること

今日の課題

- ファイルの中身を行番号付きで表示するプログラムを書け

今日の課題

- ファイルの中身を行番号付きで表示するプログラムを書け
- ヒント
 - 行番号を数えておく変数を用意する
 - 1行出力するごとに、この変数の値を出力する
 - さらに値を1増やす