

電子計算機入門

池田, 大輔
九州大学附属図書館

<https://hdl.handle.net/2324/2841>

出版情報 : 2004
バージョン :
権利関係 :

電子計算機入門 第3回 (05/19)

池田 大輔

`z4id01in@cse.ec.kyushu-u.ac.jp`

附属図書館

目次

前回課題の回答例

for ループ

range 関数

前回の課題

入力：1つの文字列 s と1つの正整数 i

出力：入力された文字列の長さが与えられた整数より長ければ文字列の前から i 文字を、短かければ文字列そのものを出力する

前回の課題

入力：1つの文字列 s と1つの正整数 i

出力：入力された文字列の長さが与えられた整数より長ければ文字列の前から i 文字を、短かければ文字列そのものを出力する

```
import sys
str = sys.argv[1]
i = int(sys.argv[2]) # int() で整数に変換
if i > len(str):
    print str
else:
    print str[:i]
```

前回の課題：別解

```
import sys
str, i = sys.argv[1], int(sys.argv[2])
# カンマで区切って一度に代入可能
print str[:i]
```

前回の課題：別解

```
import sys
str, i = sys.argv[1], int(sys.argv[2])
# カンマで区切って一度に代入可能
print str[:i]
```

実は `i` が `len(str)` より大きくても問題ない

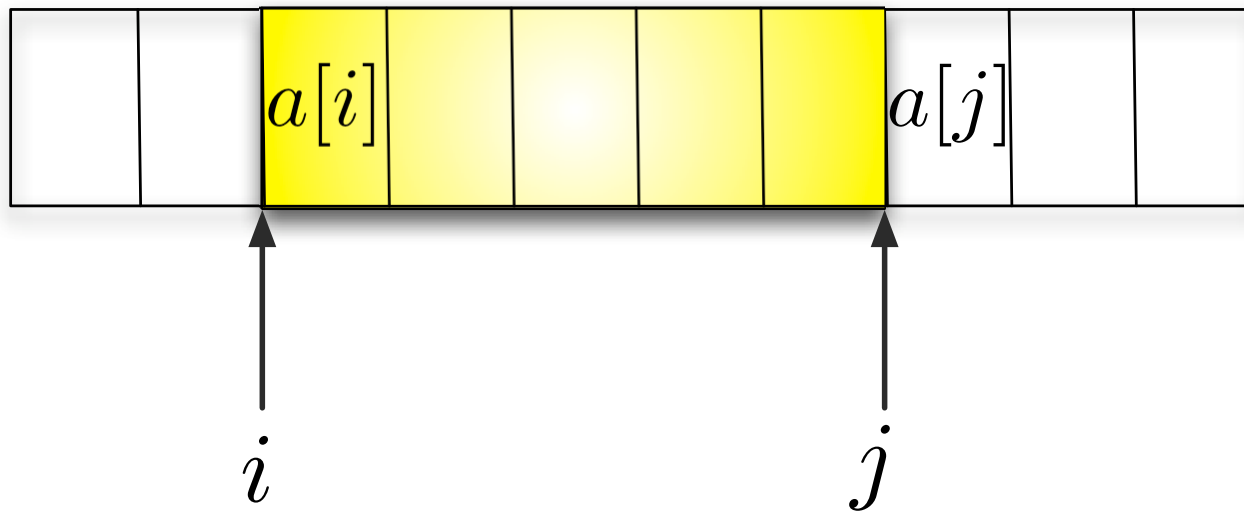
復習：配列

複数の値をまとめて扱う仕組み

$seq[i] \Leftrightarrow seq$ の i ($0 \leq i \leq len(seq)$) 番目の要素

$len(seq) \Leftrightarrow$ 配列の要素数

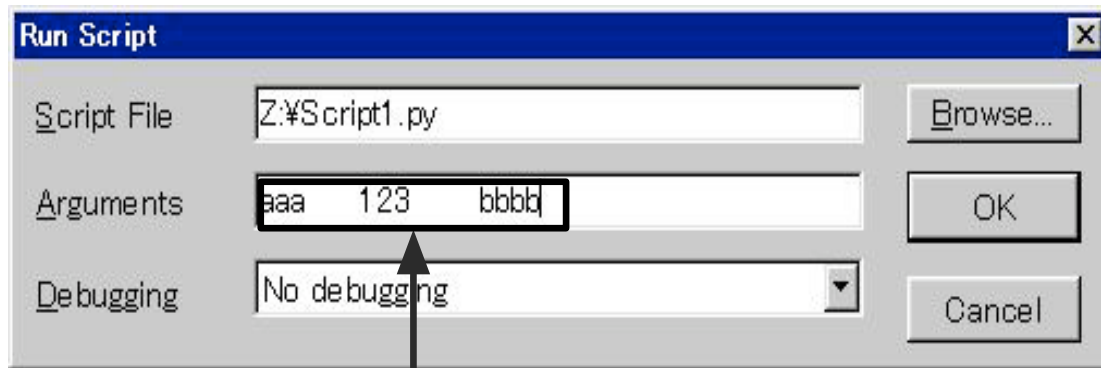
$seq[i:j] \Leftrightarrow i$ 番目から j 番目の部分配列 (部分文字列)



復習：引数

現状ではプログラムは固定した動きしかできない
決まった計算しかできない電卓のようなもの！

⇒ 入力を**引数**としてプログラムに与える
実行時の Run Script ウィンドウの [Arguments] に
与えたい入力を入れる



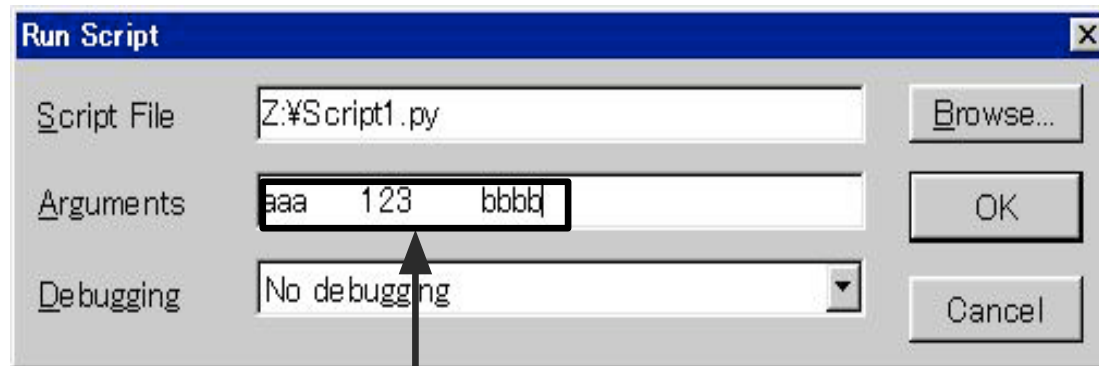
↑
3つの引数

復習：引数

現状ではプログラムは固定した動きしかできない
決まった計算しかできない電卓のようなもの！

⇒ 入力を**引数**としてプログラムに与える

実行時の Run Script ウィンドウの [Arguments] に
与えたい入力を入れる



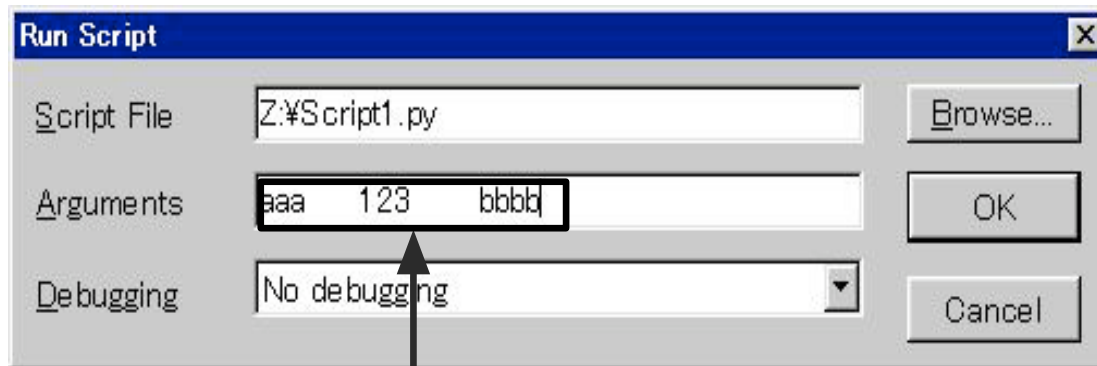
3つの引数

復習：引数

現状ではプログラムは固定した動きしかできない
決まった計算しかできない電卓のようなもの！

⇒ 入力を**引数**としてプログラムに与える

実行時の Run Script ウィンドウの [Arguments] に
与えたい入力を入れる



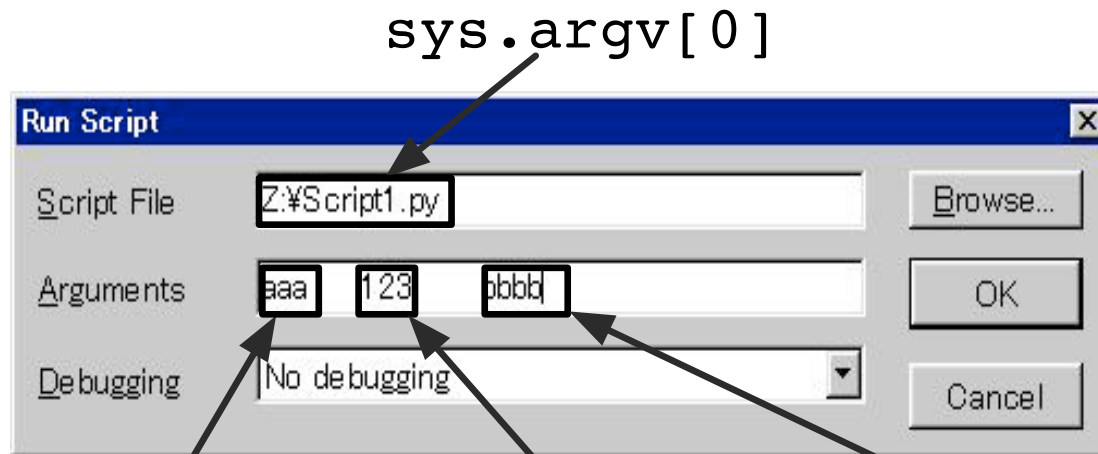
3つの引数

復習：引数(2)

プログラムから、この入力の i 番目を利用するには
`import sys #` を最初に

`v1 = sys.argv[1] #` $v1$ に `aaa` が代入

`i = sys.argv[2] #` $v2$ に **文字列の** `123` が代入
のように利用する



`sys.argv[1]`

`sys.argv[2]`

`sys.argv[3]`

引数利用時の注意

`sys.argv[i]` とプログラムで書いた場合、 i 個の入力を与えないとエラーがでる

多く与える分にはエラーにならない

先程の3番目の引数“bbbb”は単に利用されないだけで、エラーとはならない

for 文による繰り返し

配列のすべての要素になんらかの作業を行なう
書式

```
for 変数 in 配列 :  
    各要素に対する処理
```

“配列”はその要素を処理したい配列

“変数”は各要素をいれる変数で任意の名前でよい

セミコロン (“:”) で終わる

インデントで for の範囲

for 文の利用例

```
seq = ["aaa", 123, "bbbb"]  
for e in seq :  
    print len(e) # 各要素の長さを出力
```

for 文の利用例

```
seq = ["aaa", 123, "bbbb"]  
for e in seq :  
    print len(e) # 各要素の長さを出力
```

ループ 1 回目: e に "aaa" が代入される

for 文の利用例

```
seq = ["aaa", 123, "bbbb"]  
for e in seq :  
    print len(e) # 各要素の長さを出力
```

ループ 1 回目: e に "aaa" が代入される

ループ 2 回目: e に 123 が代入される

for 文の利用例

```
seq = ["aaa", 123, "bbbb"]  
for e in seq :  
    print len(e) # 各要素の長さを出力
```

ループ 1 回目: e に "aaa" が代入される

ループ 2 回目: e に 123 が代入される

ループ 3 回目: e に "bbbb" が代入される

注: print len(seq) との違いを考えよ!

実習

与えられた引数の長さをすべて出力するプログラムを作れ
ただし、引数は1つ1行の形式で表示すること
`print sys.argv` は全部を1行で出力する

実習

与えられた引数の長さをすべて出力するプログラムを作れ
ただし、引数は1つ1行の形式で表示すること

`print sys.argv` は全部を1行で出力する

```
import sys
```

```
for arg in sys.argv:
```

```
    if arg != sys.argv[0]:
```

```
        print arg
```

実習

文字列が2つ与えられたとき、両方に出現する文字をすべて出力するプログラムを書け

実習

文字列が2つ与えられたとき、両方に出現する文字をすべて出力するプログラムを書け

ヒント: 片方の文字列の各文字に関するループを作る

実習

文字列が2つ与えられたとき、両方に出現する文字をすべて出力するプログラムを書け

ヒント: 片方の文字列の各文字に関するループを作る

ヒント: その文字がもう片方にあるかどうか調べる

実習

文字列が2つ与えられたとき、両方に出現する文字をすべて出力するプログラムを書け

ヒント: 片方の文字列の各文字に関するループを作る

ヒント: その文字がもう片方にあるかどうか調べる

```
import sys

seq1 = sys.argv[1]
seq2 = sys.argv[2]

res = [] # 結果を格納する配列

for i in seq1: # seq1 の各文字に対し
    for j in seq2: # seq2 の各文字に対し
        if i == j and i not in res:
            res.append(i) # 見つけた文字を配列に追加
```


実習：別解

```
import sys
seq1 = sys.argv[1]
seq2 = sys.argv[2]
res = []
for i in seq1:
    if i in seq2 and i not in res:
        res.append(i)
```

range 関数

ある範囲の等差数列を生成する

書式

```
range(10)
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

0 から 1 おきに (10-1) までの数列

```
range(1, 11)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

最初の数を 1 からに変更

```
range(0, 30, 5)
```

```
[0, 5, 10, 15, 20, 25]
```

5 おきに変更

range関数とforループ

添字で配列にアクセスする時に利用

```
str = "abcd"
for i in range(len(str)) :
    # i に配列 (文字列) の添字が代入される
    print str[i:] # len(i) ではない!
    # i 文字から後ろを全て出力するため、
    # 添字でのアクセスが必要
```

ループ 1 回目: i に 0 が代入され、"abcd" が出力

ループ 2 回目: i に 1 が代入され、"bcd" が出力

ループ 3 回目: i に 2 が代入され、"cd" が出力

以下、同様

range 関数と for ループ (2)

添字の範囲内で適切にアクセスすること

```
for i in :  
    print array[i], array[i+1]
```

最後の要素でエラーになる

```
for i in range(len(array)):  
    if i+1 < len(array):  
        print i, array[i], array[i+1]
```

今日の課題：range()

文字列が与えられたときに、長さ3の部分文字列をすべて出力するプログラムを書け

例えば

"accgt"が与えられたら、出力はacc, ccg, cgtである

ヒント

文字列 `str` の部分文字列は `str[i:j]`

for ループと `range` 関数を使って、添字によるアクセス

第1回レポート

与えられた文字列のすべての部分文字列を出力する
プログラムを書け

例えば"acgt"が入力されたら、出力は
a, c, g, t, ac, cg, gt, acg, cgt, acgt
である

文字列の各位置が部分文字列の最初の文字になりうる
スタートより後の各位置が部分文字列の最後の文字
for を2回使う

提出に関する注意

プログラムと適当に選んだ文字列に対する出力例を1つ添えて提出しなさい

プログラムは添付すること

出力例はコピー&ペーストしてメールにはりつける

コピーしたい範囲をドラッグして反転させる

右クリックから [コピー] を選択

はりつけたい場所 (GraceMail など) で右クリックから [貼り付け] を選択

締切 6/1(火)

少なくともエラーなく動作することを教育用システムのパソコンで確認すること

家のパソコンで作成しても構わない

pythonwin のインストール方法を授業の Web ページに用意しています

付録：for文の制御

else, break と continue で for 文を制御する

break ⇔ for ループを打ち切る

else ⇔ break が実行されなかったときに実行される

continue ⇔ 残りを無視して次の要素以降をループ

例

```
for i in array:
    if i == "fin": break
    # "fin"という要素があるかないか
    if i == "continued": continue
    # これより下の行は実行せず次の array の要素へ
    他の処理
else:
    print "Not found"
```