

計算機構成論（おまけ）

CとPythonの比較

2023年度春学期
情報理工学部 Rクラス担当
越智裕之

例題 1 : 1から10までの整数の総和

(1) C言語の場合

```
#include <stdio.h>
int main( void ) {
    int s,i;
    s = 0;
    i = 1;
    while (i <= 10) {
        s = s + i;
        i = i + 1;
    }
    printf("%d\n", s);
}
```

始めの呪文

使う変数の**宣言**

代入文

文末には
;が必要

while文による繰り返し
繰り返す文を**{と}**で囲む

変数 s の値を表示する呪文

終わりの呪文

例題 1 : 1から10までの整数の総和

(2) Python言語の場合

```
s = 0
i = 1
while i <= 10 :
    s = s + i
    i = i + 1
print(s)
```

最初の呪文は不要

使う変数の**宣言は不要**

代入文

文末の
;は無くても可

while文による繰り返し
繰り返す文を
インデントで表す

変数 s の値を表示する呪文

終わりの呪文は不要

例題 1 : 1から10までの整数の総和 ソースコードの比較

- Pythonの方が簡潔だが、基本的には似ている

```
#include <stdio.h>
int main( void ) {
    int s,i;
    s = 0;
    i = 1;
    while (i <= 10) {
        s = s + i;
        i = i + 1;
    }
    printf ("%d\n",s) ;
}
```

C言語 (11行)

```
s = 0
i = 1
while i <= 10 :
    s = s + i
    i = i + 1
print(s)
```

Python言語 (6行)

例題 1 : 1から10までの整数の総和 実行方法の比較

- Cはコンパイルが必要

```
% gedit sum.c  
% gcc -o sum sum.c  
% ./sum  
55  
%
```

ソースファイルの編集
(拡張子は .c)

コンパイラ gcc を使い
実行形式ファイルを生成

実行

printf 文の表示

- Pythonはすぐに実行できる

```
% gedit sum.py  
% python sum.py  
55  
%
```

ソースファイルの編集
(拡張子は .py)

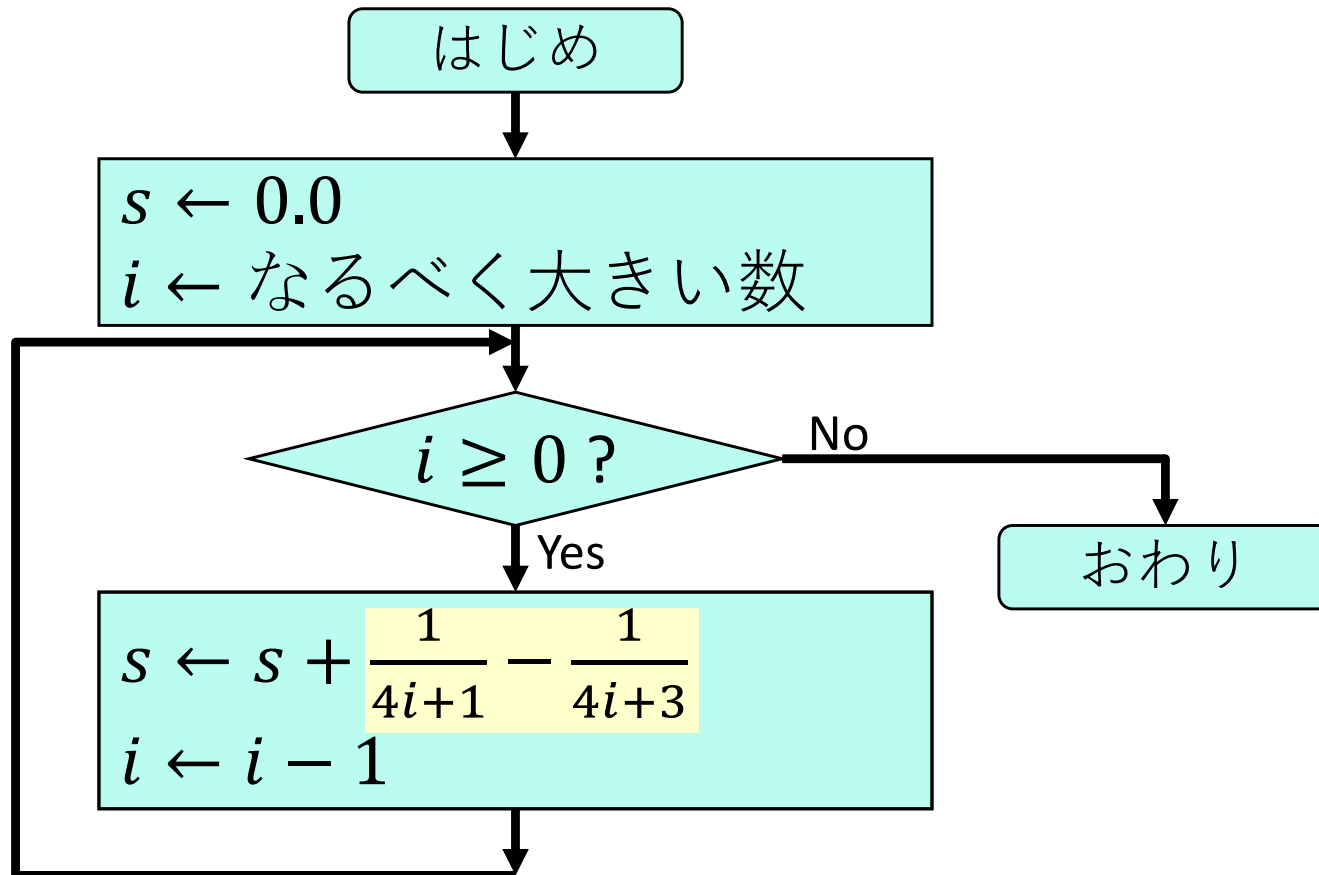
pythonコマンドにソー
スファイルを読ませる
ことで実行

print 文の表示

例題 2 : ライプニッツの公式 アルゴリズム

- 以下の無限級数で円周率の近似値を求める

$$\frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} + \dots + \frac{1}{4i+1} - \frac{1}{4i+3} \dots = \frac{\pi}{4}$$



例題 2 : ライプニッツの公式

(1) C言語のソースコード

```
#include <stdio.h>
int main(void) {
    double s;
    int i;
    s = 0.0;
    i = 1000;
    while (i>=0) {
        s = s + 1.0/(4*i+1) - 1.0/(4*i+3);
        i = i-1;
    }
    printf("%13.11f¥n", 4*s);
}
```

例題 2 : ライプニッツの公式 (2) Python言語のソースコード

```
s = 0.0
i = 1000
while i >= 0 :
    s = s + 1.0/(4*i+1) - 1.0/(4*i+3) ;
    i = i-1;
print(4*s)
```


例題 2 : ライプニッツの公式 実行結果

i の初期値	計算結果	Pythonの 実行時間	Cの 実行時間
10	3.09616152646	0.0秒	0.0秒
100	3.13664218887	0.0秒	0.0秒
1000	3.14109315312	0.0秒	0.0秒
10000	3.14154265859	0.0秒	0.0秒
100000	3.14158765364	0.1秒	0.0秒
1000000	3.14159215359	0.3秒	0.0秒
10000000	3.14159260359	2.4秒	0.0秒
100000000	3.14159264859	24.2秒	0.2秒
1000000000	3.14159265309	297.1秒	2.1秒

※ Intel Core i7 10thGen. CPU搭載PCで実験



約140倍の性能比

CとPythonの比較

まとめ

- 文法は様々な違いがあるが、類似点も多い
 - **Pythonの方が、プログラムは概して簡潔になる**
 - 1つの言語をしっかりマスターすれば、別の言語を修得するのは、そんなに大変ではない
- Pythonはインタプリタ型、Cはコンパイル型の言語
 - **Pythonは、ソースコードをインタプリタに読ませることで実行できる（コンパイル不要）**
 - Cはソースコードをコンパイルして実行形式（機械語）ファイルを生成し、それを実行させる
- 実行性能は大きく差が出る
 - **Cはコンパイラで機械語に翻訳してから実行するので、高い性能が得られる**
 - 先ほどの実験では約**140倍**ぐらいの差がみられた