計算物理学B

第3回

野垣 康介、藤本 悠輝

講義予定

10/07 両名：四則演算
10/14 野垣：制御文(for, if)
10/21 野垣：関数
10/28 藤本：配列(numpy)
11/04 藤本：可視化(matplotlib)
11/11 野垣：数値微分
11/18 藤本：数値積分

～中間レポート～

12/09 野垣：モンテカルロ1
12/16 野垣：モンテカルロ2
12/23 藤本：微分方程式1
01/13 藤本：微分方程式2
01/20 藤本：微分方程式3
01/27 野垣：最適化
02/03 藤本：機械学習

～期末レポート～

はじめに

授業で用いるリンクをまとめておきます。

Google Colab

https://colab.research.google.com/?hl=ja

GitHub

https://github.com/nogaki/Computational_Physics_B

GitHub（今週の教材）

https://github.com/nogaki/Computational_Physics_B/tree/main/week3

前回の補足

GitHubからGoogle Colabへのノートブックの取り込みについて

Colabの画面で
[ファイル]→[ノートブックを開く]→タブ[GitHub]をクリック

GitHub上のノートブックのurlを打ち込むと、直接取り込めます。

https://github.com/nogaki/Computational_Physics_B/tree/main/week3/week3.ipynb

[注意!]この状態では、GitHub上のファイルを開いただけで、
個人のドライブに保存されていません!
[ファイル]→[ドライブにコピーを保存]を実行すること。

break文（前回の補足）

For文やwhile文の繰り返し処理の途中でループから抜け出す構文。

```
for i in range(10):
    print(i)
    if i > 5:
        break
```

実行結果

```
0
1
2
3
4
5
6
```

関数

論理的に等価な処理が繰り返し現れることがあります。

```
sum1 = 0
for i in range(10):
    sum1 = sum1 + (i+1)
```

1から10の総和

```
sum2 = 0
for j in range(30):
    sum2 = sum2 + (j+1)
```

1から30の総和

```
print(sum1 + sum2)
```

2つの合計

関数

このような場合には、関数を用いましょう。

```
def sum_one2n(n):
    sum = 0
    for i in range(n):
        sum = sum + (i+1)
    return sum
```

1からnの総和を返す関数

```
sum1 = sum_one2n(10)
sum2 = sum_one2n(30)
```

関数呼び出し

```
sum1 + sum2
```

2つの合計

関数

関数を用いることで、プログラムの可読性、可搬性を向上できます。

```
def 関数名(引数):
    [関数内部の処理]
    return 返り値
```

インデント（字下げ）する（Tabキー）

- 関数の内部で定義した変数は関数の外では、利用できません。
- 引数や返り値は複数設定できます。
- 返り値が複数の場合は、タプル（次回の内容）で返って来ます。

関数

```
def sum_all(x, y, z):
    tmp = x + y
    return (tmp + z)

l = 3
m = 2
n = 5

print(sum_all(l, m, n))
print(tmp)
```

出力は10

これはエラー

関数の活用

漸化式 $a_1 = 1,\ a_{n+1} = 2a_n + 3$ の $a_{n=10}$ は？

一般項 $a_n = 4 \cdot 2^{n-1} - 3$

```
def next_term(a):
    return 2*a + 3


a = 1
for _ in range(9):
    a = next_term(a)

print(a)
```

出力は2045

関数の活用

n, mの最小公倍数を求める関数（ユークリッドの互除法）

```
def gcd(n, m):
  while m != 0:
    r = n % m
    n = m
    m = r
  return n
```

ループ版

① nをmで割った余りrを求める
② mをrで割った余りr1を求める
③ r をr1で割った余りr2 を求める
……
余りが0になったときの割る数が、最大公約数

関数の活用

n, mの最小公倍数を求める関数（ユークリッドの互除法）

再帰処理版

```
def gcd_recursive(n, m):
    if m == 0:
        return n
    else:
        return gcd_recursive(m, n % m)
```

① nをmで割った余りrを求める
② mをrで割った余りr1を求める
③ r をr1で割った余りr2 を求める
……
余りが0になったときの割る数が、最大公約数

実習タイム

例題

$$\sum_{n=1}^{10} a_n \qquad a_n = n^2 + 4n + 5$$

フィボナッチ数列の10項目を求めよ（関数を用いて）

引数nが素数かどうか判定する関数を作成せよ。

import mathというおまじないを唱えると、
math.sqrt(n)でnの平方根を計算でき、
math.floor(x)で[x]を計算できる（[]はガウス記号）。