

# 数理生物学演習

第1回 Hello, World!

野下 浩司 (Noshita, Koji)

✉ noshita@morphometrics.jp

🏠 <https://koji.noshita.net>

理学研究院 数理生物学研究室

## 本演習の進め方

# 演習の基本方針

- ★★★★ 数理生物学的トピック：数理生物学に興味を持ってもらう  
例：ロトカ-ボルテラモデルによる被食者-捕食者の個体群動態の記述
- ★ 研究への挑戦：研究を進める上で必要な、または役立つ経験をする  
例：文献の探し方、使うと便利なツールの紹介
- ★★ 数値計算的アルゴリズム：計算機で問題を解く方法を学ぶ  
例：ニュートン法により方程式を数値的に解く
- ★★★★ プログラミングのスキル：実際に具体的な問題を解いてみる  
例：if文, forループ, 配列



**自分の知りたいことへアプローチする方法を学ぶ**

## 演習の進め方：各回の基本的な取り組み

- 30分前後 1.内容の説明・フィードバック
- 150分前後 { 2.実際にプログラミング
- 好きなだけ { 3.課題をこなす
- 4.提出

**変更があればWebページにて連絡**

# 成績

## GPA目安

- A Excellent 90以上
- B Good 89-80
- C Fair 79-70
- D Pass 69-60
- F Fail 59以下

## 理論的最大点

- ノーマル：100
- ハード：120

- 課題提出：基本的に毎回提出。出席を兼ねる。  
Moodleにて提出
  - ノーマル：55点前後，ハード：65点前後
- 中間報告（第10回に実施予定，チームでの取り組み）  
第9回の課題発表と最終課題への方針の説明
  - 全員：10点前後
- 最終課題  
数理生物学的なテーマを自分で設定して，計算機を使ったアプローチで取り組み，レポートにまとめる
  - ノーマル：35点前後，ハード：45点前後

# 本日の課題

**注意：氏名，学籍番号，所属を必ず書く！**

1. 今回の授業でよくわからなかった点を述べる。特になければ，より知りたいと思った点を教えて。
2. 今後扱って欲しい内容を挙げる（複数可）。
3. 自作の「Hello, World!」プログラムのソースコード（.pyファイル）いずれか1つとノートブック形式（.ipynbファイル）を提出（これはPDFに含めない）。
4. その他質問，感想，要望をどうぞ。

**課題をPDFファイルにまとめて，Moodleにて提出すること**

# 連絡

- Webページ  
<https://koji.noshita.net/page/compbio/compbio2019/>
- e-mail  
野下 浩司 [noshita@morphometrics.jp](mailto:noshita@morphometrics.jp)
- Slack（後ほど説明します）  
[ku-compbio.slack.com](https://ku-compbio.slack.com)
- 訪問（事前連絡してください）  
伊都キャンパス ウエスト1号館 W1-C-906

研究室の様子を覗きに来るのも可  
その場合はW1-C-909付近で誰か見つけて話しかけて！

# 第1回：Hello, World!

## 本日の目標

- プログラミング言語とは？
- 本演習で使うプログラミング言語の紹介
- プログラミング環境の構築
- 簡単な計算や結果の表示

# プログラミング言語って？



このままでは分かり合えない  
→ 互いの言語の“翻訳”が必要

# プログラミング言語って？



>> 4%3

“翻訳”：人 → コンピュータ

```
#include <stdio.h>
int main(){
  int m,n;
  m=4;
  n=3;
  printf(“%d人”, m%n);
  return 0;
}
```

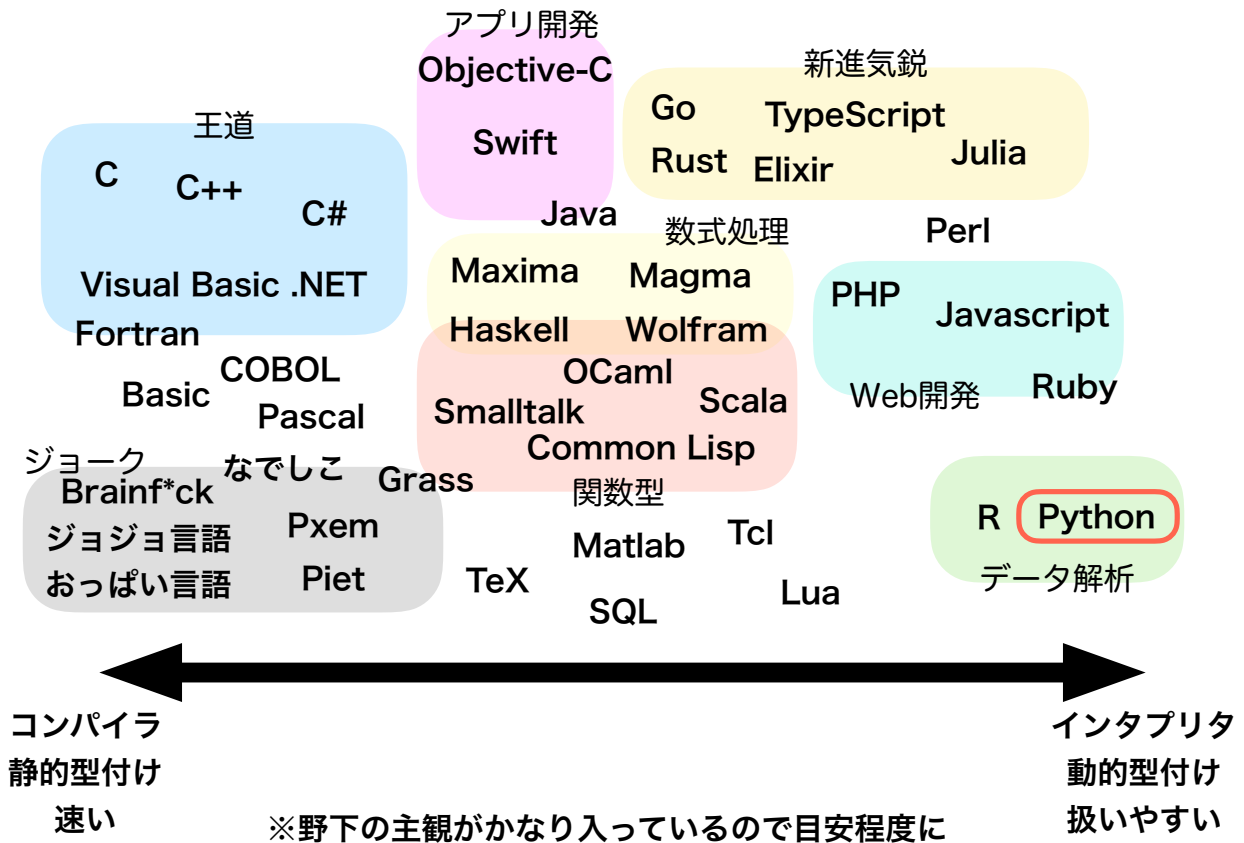
RandomChoice[{"I", "you", "he", "she"}]

- 1
- 1人
- you



“翻訳”：コンピュータ → 人

# プログラミング言語いろいろ



## Python

<https://www.python.org/>



この演習で利用するプログラミング言語

インタプリタ型, 動的型付け

用途: 数値シミュレーション, 可視化, データ解析

Pythonは数理生物学的なシミュレーションだけでなく、データ解析や画像解析などにも利用されている  
→ 生物の分野で扱うデータやその解析にも利用できる。様々な環境（ラボとか職場とか）へ応用しよう。

# Hello, World!

特に理由は無いけれどプログラミング言語を  
学ぶと最初に出てくる練習用プログラム

目的

ターミナル等に「Hello, World!」と出力する

- きちんと環境の設定が出来ているか
    - Pythonはインストールできている？
    - パスは通っている？
    - 開発環境の準備は？
- …などをチェックする.

プログラミング環境を準備しよう！

# 皆さんへのお願い

- わからないところがあればすかさずググろう！  
調べる習慣をつける。
- 困ったら手を挙げてTAを呼ぼう。  
周りに困っている人がいれば助けてあげよう。
- 質問や回答をSlackへ投稿するのもオススメ。  
情報が共有できる。一人の質問が皆の質問に！
- 演習中の休憩は自由。疲れ果てる前に休もう。

Slack

<https://slack.com>



チャット系のコミュニケーションツール

## 演習での活用方法

- 質問する
  - 例) インストールが～で止まる, ～の図がきれいに表示されない
- 他の人の質問に答える
  - 例) ～にチェックを入れるといいよ, 途中の～の計算ができてないんじゃない?
- 情報を共有する
  - 例) ～の参考になるページを見つけた, ～は～からダウンロードできる
- その他にも雑談など様々な活用してください



# Slack 数理生物学演習ワークスペース

[ku-compbio.slack.com](https://ku-compbio.slack.com)

以下の招待リンクから参加しよう！

デスクトップ, iOSアプリ, Androidアプリ, Webブラウザから利用可能

## 九州大学Moodle

<https://moodle.s.kyushu-u.ac.jp/>



オープンソースeラーニングプラットフォーム「Moodle」を利用した  
九州大学のオンライン学習システム (の一部)

### 演習での活用方法

- 課題の提出
  - 超重要！提出方法については最後に説明します。
- 質問する, 質問に答える
  - Q&Aの掲示板を利用。

### 登録方法

1. 九州大学Moodleへログイン (SSO-KIDとパスワードが必要)
2. コースを検索する → 「数理生物学演習」をキーワードに検索
3. 「2019年度前期・月3月4・数理生物学演習」を選択
4. 「私を登録する」をクリック

# PC環境の設定

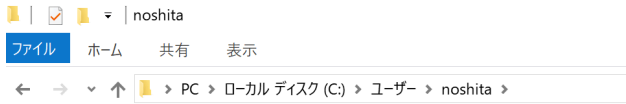
## 1. 作業ディレクトリの作成

ホームディレクトリの直下に演習用のディレクトリを作る

### Win

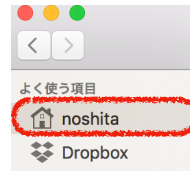
ホーム：エクスプローラーから

ローカルディスク (C:) → ユーザー → ユーザー名



### Mac

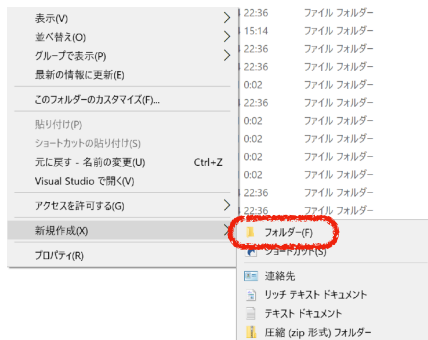
ホーム：Finderの家のマーク



歯車マーク→新しいフォルダ



右クリック→新規作成→フォルダー



ディレクトリ名は「CompBio2019」にしてください

# PC環境の設定

## 2. 今回（第1回）用の作業ディレクトリの作成

~/CompBio2019の直下にディレクトリを作る

ディレクトリ名は「01」にしてください

今回は、この「01」の中にコードを配置していきます。

次回以降は新たに02, 03, …と回数を名前にしたディレクトリを~/CompBio2019の直下に作って、毎回その中にコードを配置するようにしてください

# 用語解説

- ディレクトリ directory  
データの保管場所のこと。フォルダとも呼ばれる。
- パス path  
ディレクトリやファイルの“位置”を示すもの。  
例) ~/CompBio
- CUI (Character User Interface) ・ CLI (Command Line Interface)  
文字ベースのインターフェース。コマンドなどを入力して、結果を文字列で受け取る。ターミナルやコマンドプロンプトはCUI。  
皆さんが普段使うアプリのインターフェースは殆どがGUI (Graphical User Interface) 。
- インデント indentation 字下げ  
行頭に空白を入れて開始位置を下げる。Pythonではインデントを特定の構文の範囲指定に利用するため注意が必要 (詳しくは今後の演習で) 。

## 演習 Webページ

<https://koji.noshita.net/page/compbio/compbio2019>

# Pythonの環境構築

<https://koji.noshita.net/page/compbio/compbio2019/>

# テキストエディタ (VSCode) の環境構築

<https://koji.noshita.net/post/2019/0407-envvscode2019/>

実際にプログラムを組んでみよう！

## Pythonの実行方法いろいろ

- 対話型評価環境

式やコードを入力し、それをインタプリタが解釈・実行し、結果を表示する。ちょっとした計算や確認に用いられる。

- ソースコード

プログラミング言語で書かれたテキストファイル（=ソースコード）を読み込んで、上から順に解釈・実行していく。大規模な計算や繰り返し用いられるコード（パッケージ）を利用する場合に用いられることが多い。

- ノートブック

セルと呼ばれる入力・出力エリアからなる実行環境であり、コードやその出力結果、注釈などのテキストを記録できるファイル。データ解析を行う際の環境としてよく利用される。

**本演習では主にノートブックを利用して進めていきます**

# 対話型評価環境 (REPL) インタラクティブシェル

1. ターミナルを起動し, 仮想環境に入る
2. pythonと入力

```
~/D/1/2/C/n/01 >>> python
Python 3.7.3 (default, Mar 27 2019, 16:54:48)
[Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

3. Hello, World!

```
01. Hello, World!
print("Hello, World!")
```

4. 終了

```
~/D/1/2/C/n/01 >>> python
Python 3.7.3 (default, Mar 27 2019, 16:54:48)
[Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello, World!")
Hello, World!
>>> exit()
~/D/1/2/C/n/01 >>> █
```

# 対話型評価環境 (REPL) IPython

IP[y]: IPython  
Interactive Computing

1. ターミナルを起動し, 仮想環境に入る
2. ipythonと入力

```
((miniconda3-4.3.30) ~/D/1/2/C/n/01 >>> ipython
Python 3.6.8 |Anaconda, Inc.| (default, Dec 29 2018, 19:04:46)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.4.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]:
```

3. Hello, World!

```
01. Hello, World!
print("Hello, World!")
```

4. 終了

```
((miniconda3-4.3.30) ~/D/1/2/C/n/01 >>> ipython
Python 3.6.8 |Anaconda, Inc.| (default, Dec 29 2018, 19:04:46)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.4.0 -- An enhanced Interactive Python. Type '?' for help.

In [1]: print("Hello, World!")
Hello, World!

In [2]: exit()
((miniconda3-4.3.30) ~/D/1/2/C/n/01 >>> █
```

# ソースコード

## 02-1. Hello, World! ソースコード1

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

print("Hello, World!")
```

シバン (shebang) : UNIX環境でどのインタプリタ  
を利用するか指定する

文字コードの指定

## 02-2. Hello, World! ソースコード2

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

if __name__ == '__main__':
    print("Hello, World!")
```

- print(##)  
##を出力

出力

Hello, World!

↑  
インデント

**!注意 : pythonではインデントがとても重要な意味を持ちます。  
詳しくは、今後の演習で解説していきます。**

実行方法

- python ファイル名

```
(miniconda3-4.3.30) ~/D/1/2/C/n/01 >>> python 02-1.py
Hello, World!
```

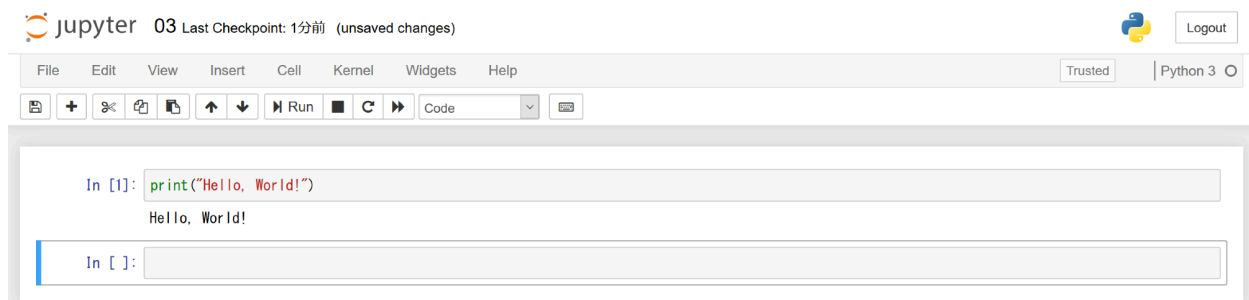
# ノートブック

「数理生物学演習 (2019年度) 向けPython環境構築」の  
「基本的な使い方」を参照

<https://koji.noshita.net/post/2019/0405-envpython2019/>

03. Hello, World! ノートブック  
print("Hello, World!")

ノートブックに「03」という名前をつけて  
保存しよう!

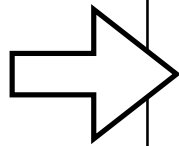


本演習では主にノートブックを利用して進めていきます

# 課題提出 Moodle

1. Moodleへのログイン
2. 「2019年度前期・月3月4・数理生物学演習」コースへ移動
3. 各回の課題モジュールから提出

ココ！



## 第1回 Hello, World!

- プログラミングの意義, プログラムが動くまで
- 環境導入

### 第1回

1. 今回の授業でよくわからなかった点を述べる. 特になければ, より知りたいと思った点を教えて.
2. 今後扱って欲しい内容を挙げる (複数可).
3. 自作の「Hello, World!」プログラムのソースコード (.pyファイル) いずれか1つとノートブック形式 (.ipynbファイル) を提出 (これはPDFに含めない).
4. その他質問, 感想, 要望をどうぞ.

## 本日の課題

**注意：氏名，学籍番号，所属を必ず書く！**

1. 今回の授業でよくわからなかった点を述べる. 特になければ, より知りたいと思った点を教えて.
2. 今後扱って欲しい内容を挙げる (複数可).
3. 自作の「Hello, World!」プログラムのソースコード (.pyファイル) いずれか1つとノートブック形式 (.ipynbファイル) を提出 (これはPDFに含めない).
4. その他質問, 感想, 要望をどうぞ.

**課題をPDFファイルにまとめて, Moodleにて提出すること**