

基本情報/Basic Information

開講科目名 ／Course	バイオデータプログラミング演習 I /Exercise of Biological Data Programming I
時間割コード ／Course Code	47243-23
共通科目コード ／Common Course Code	GFS-MJ6D04S1
学部/大学院 ／Undergraduate/Graduate	大学院 / Graduate
開講所属 ／Course Offered by	新領域創成科学研究科 / Graduate School of Frontier Sciences
ターム・学期 ／Term・Semester	2025 年度 / Academic Year S1 / S1
曜限 ／Day, Period	水 / Wed 3, 水 / Wed 4
開講区分 ／semester offered	S1 / S1
単位数 ／Credits	1.0
学年 ／Year	M1/M2/D1/D2/D3
他学部履修 ／Available to students in other faculties	可
備考 ／Notes	
主担当教員 ／Main Instructor	笠原 雅弘
教室 ／Classroom	理学部3号館 412, 新領域情報生命科学実験棟 2F 講義室 / Faculty of Science Bldg.3 412, Computational Biology Laboratory, GSFS 2F 講義室

担当教員情報/Instructor Information

教員名 /Instructor	教員所属名 /Affiliation	職名 /Job Title
笠原 雅弘	メディカル情報生命専攻	准教授
伊藤 耕一	メディカル情報生命専攻	教授
木立 尚孝	メディカル情報生命専攻	准教授

講義題目 /Subtitle	バイオデータプログラミング演習 I	
授業の目標、概要 /Course Objectives/ Overview	<p>必要に迫られてプログラミングを書籍で独学すると基礎知識の不足に悩むことが多い。本演習ではそのような学生が過去に躓いたポイントを調べ、必要な基礎知識を大量に補った独自のオンラインプログラミング学習教材を作成した。生物学・医科学のプログラミングに必要な概念をゼロから広く浅く学び、研究に必要な高度なプログラミングを将来的に独学できるような「基礎体力」を養う。初学者は大量の自習が必要であり、かなり時間を必要とする。</p>	
授業情報 /Class information	<p>生命科学系の情報解析を行う全ての学生に知って欲しい内容となっている。また、初学者にとっては内容が極めて多いが各項目は比較的易しい。</p> <p>本演習の履修はは教育プログラムである DSTEP の修了要件の一つとなっている。また、講義の性質上、途中から受講することも可能である。</p> <p>本講義は対面・オンラインのハイブリッド講義を予定している。</p> <p>柏: 情報生命科学実験棟2F講義室 (http://www.kashiwa.u-tokyo.ac.jp/download/GuideMap2018.pdf の12番の建物の階段を上がってすぐの部屋)</p>	
授業のキーワード /Keywords	日本語用 /Japanese	Python, プログラミング, Linux, 統計, 正規表現, SQL, データベース, R
	英語用 /English	Programming, Linux, Statistics, Regular expression, SQL, Database, R, Python
授業計画 /Schedule	<p>1. 初回チュートリアル プログラミング言語の分類と選択 エディタの基本的な使い方</p> <p>2. コンピューターの基本的な仕組み UNIX コマンドライン</p>	

ソフトウェアのインストール方法

3.

Python の基本制御構造・関数

Python のクラス

文字列処理

Python で CSV/TSV ファイルを処理

4.

正規表現

JSON と XML

5.

Python のモジュールと PyPI

関係データベースと SQL

6.

R の変数・基本制御構造と関数

R の リスト・行列

整然データ

7.

集計とグラフ描画

統計検定

8.

R のパッケージ

Bioconductor の紹介

Jupyter

9.

バージョン管理

10.

テストと継続的インテグレーション

11.

SSH と遠隔サーバーの使い方

分散ファイルシステム

12.

バッチジョブキューシステム

	<p>13. ワークフローと再現性</p> <p>14. 復習と試験(最終実技試験は5月以降いつでも受けることができる)</p>
<p style="text-align: center;">授業の方法 ／Teaching Methods</p>	<p>プログラミングやコマンドラインの演習は手を動かしてプログラムを書き、コマンドを自分で叩いて鍛錬することがとにかく大事である。 プログラムを書き、書いたプログラムが思い通りに動かずに悩んだ時間を積み上げることなくしてプログラミングスキルが向上することはない。</p> <p>プログラミングの演習をこなす速度は非常に個人差が大きく、同じ課題に対してある者は1日で全てを終えてしまうこともできるだろうし、ある者はまるまる10日以上を学習に費やすかもしれない。 このため、いわゆるマस्पロ型の講義・演習では進度が早すぎて大部分の者がついてこれない、もしくは進度が遅すぎて大部分の者が退屈するという大きな問題があった。</p> <p>そこで、本演習は講義トピックの説明をほぼ全て動画で行い、演習の時間はほぼ全て質問時間(と試験時間)とする。 このやり方では個別の学生が進度に応じて学習を進められるようにする。 具体的には、学生は動画の説明を聞いて Web ベースのテスト問題を解く形式で演習を進める。このため、イヤホンやヘッドホン等を持ってくることを強く推奨する。 また、演習の時間中は随時質問を受け付けている。</p> <p>また、出席は取らないので質問が無ければ自宅等で学習を進めても構わない。 しかし、プログラミング初心者の学生は全く質問をせずに滞りなく学習を進められることは考えにくいいため注意されたい。</p>
<p style="text-align: center;">成績評価方法 ／Grade Evaluation</p>	<p>試験の大部分は Web ベースの知識テストで毎日受けることができ、成績評価される。 多くの受講生は提出したプログラム等に基づき成績評価される。</p> <p>しかし、選択問題や穴埋め問題など創造性に乏しく誰が解いても同じような回答となる問題のみを 主に解いている受講者については通常の Web テストに加えて本人確認テストを行う。</p> <p>本人確認試験は Web テストで正答した問題の中から出題される。 プログラミング課題については内容を問う質問を行うことがある。 本人確認試験に合格すればその場で Web テストに基づいた評価で単位の取得を認</p>

	める。 正答率が低い場合には本人確認失敗となり、次回に再チャレンジとなる。
教科書 ／Textbook	本講義で指定する特定の教科書は無いが、自分の不得意分野については関連する教科書を読むことを推奨する。
参考書 ／Reference	<ul style="list-style-type: none"> ・Python スタートブック(辻真吾著;初心者向け) ・Python 言語によるプログラミングイントロダクション(第2版): 世界標準 MIT 教科書(ジョン・V. グッターグ(著), John V. Guttag(著), 久保幹雄(翻訳))(中級者向け。とても良い教科書だが分厚く、期間内に読み切るのは難しいだろう) ・統計学:Rを用いた入門書(Michael J.Crawley(著), 野間口 謙太郎(翻訳), 菊池 泰樹(翻訳))(初心者～中級者向け)
履修上の注意・準備学習等(予習、復習) ／Notes on Taking the Course ／Assignments / Instructions for Preparation and Review	本演習ではノートパソコン等(Windows/Linux/macOSのうち何れか)を持ってくること。 中級者以降の Windows ノートパソコン所持者は Windows Subsystem for Linux 2 (WSL2) を用いた Linux 環境を構築してから持参してほしい。
関連ホームページ ／Course-Related Websites	https://bioprogram1.2025.class.kasahara.ws/
その他 ／Others	
メールアドレス ／e-mail address	mkasa@k.u-tokyo.ac.jp
授業使用言語 ／Language Used in Class	日本語
授業実施形態 ／Course delivery modalities	対面・オンライン併用型 A(総時間数の半数以上を対面で実施)／Hybrid Type A(Face-to-face/Online): Conduct classes in-person 50% or more of the total hours of the course