

北海道大学シラバス

<b>科目名</b> 一般教育演習(フレッシュマンセミナー)					
<b>講義題目</b> プログラミングで問題を解く：集計から人工知能まで					
<b>責任教員 (所属)</b> 瀧川 一学(大学院情報科学研究科)					
<b>担当教員 (所属)</b> 瀧川 一学(大学院情報科学研究科)					
<b>科目種別</b>	全学教育科目(一般教育演習)			<b>他学部履修等の可否</b>	可
<b>開講年度</b>	2018	<b>期間</b>	1 学期	<b>時間割番号</b>	000020
<b>授業形態</b>	演習	<b>単位数</b>	2	<b>対象年次</b>	1～
<b>対象学科・クラス</b>	基礎1-53組			<b>補足事項</b>	
<b>ナンバリングコード</b>	GEN_LIB 1000				
<b>大分類コード</b>	<b>大分類名称</b>				
GEN_LIB	全学教育 (教養科目)				
<b>レベルコード</b>	<b>レベル</b>				
1	全学教育科目 (語学上級科目、高年次対象科目を除く)				
<b>中分類コード</b>	<b>中分類名称</b>				
0	一般教育演習 (フレッシュマンセミナー)				
<b>小分類コード</b>	<b>小分類名称</b>				
0	一般教育演習 (フレッシュマンセミナー)				
<b>言語</b> 日本語で行う授業					

**キーワード**

プログラミング、コンピュータ、アルゴリズム、データ構造、人工知能

**授業の目標**

プログラミングとは何だろうか？ある人はそれは「アート」な知的活動だと言い、ある人はそれはコンピュータに魔法をかけるための「呪文」だと言う。今や私たちの社会や生活は、電子機器を動かしたりデータをやり取りする無数のコンピュータプログラムが支えている。プログラミングとはそれらのコンピュータプログラムを作る活動である。この入門授業では、実際にプログラムを作って動かすという実習とフォローアップディス

カッションを通してプログラミングとは何かを理解することを目標とする。この「アート」な「呪文」に関心があるなら、文系理系やプログラミングの経験は全く問わない。

## ■ 到達目標

1. 「プログラミングとは何か？」について自分なりの答えが持てるようになる。
2. プログラミングに最低限必要な「コンピュータの仕組み」や「アルゴリズム」「データ構造」と言った初歩的知識を身につける。
3. 実際にプログラミング言語を用いて簡単なプログラムを作成するための基本的なスキルを身につけ実践することができる。
4. 今後、プログラミングを自ら習得していくために何をしていけば良いかの理解や知識を得る。
5. コンピュータプログラムが我々の生活社会の基盤にあることを意識し、品質の高い美しいプログラムを作り上げることの価値を議論できるようになる。

## ■ 授業計画

1. ガイダンス：授業の進め方を説明
2. プログラミングとは？(1)：自己紹介～コンピュータの仕組みとプログラム、その社会における役割
3. プログラミングとは？(2)：実習環境設定～初めてのプログラミング
4. 基本文法を学ぼう(1)
5. 基本文法を学ぼう(2)+フォローアップ
6. 標準ライブラリを使ってみよう(1)
7. 標準ライブラリを使ってみよう(2)+フォローアップ
8. プログラムの構造と設計(1)
9. プログラムの構造と設計(2)+フォローアップ
10. データを読み解こう(1)
11. データを読み解こう(2)+フォローアップ
12. 品質の高い美しいプログラムを作るには(「アート」な「呪文」への道)+ディスカッション
13. 画像を認識するAIプログラムを作ってみよう(1)
14. 画像を認識するAIプログラムを作ってみよう(2)
15. 授業のまとめと振り返り：この後どうする？

## ■ 準備学習(予習・復習)等の内容と分量

各自の必要に応じて、時間内に終わらなかった実習課題の自習や、プログラミング言語習得についての自主的情報集を薦める。30分程度で良いので、予習か、復習かいずれかとして、授業でやった内容を少し自分なりに整理するだけでも、より学習効果があると思われる。各自のPCでも実際に動かせる環境の整え方から実習するので、実際に自分のPCでも色々と遊んでみるのが望ましい。

## ■ 成績評価の基準と方法

- スキル習得の演習的な側面があるため、基本的には全ての授業への参加が望ましいが、やむを得ない場合や既にスキルがある場合もあるので、出席2/3(10回)以上を成績評価対象とする。満たない場合、不合格とする。  
- 成績評価は、授業や演習への積極性(30%)、レポートや成果物の内容(50%)、ディスカッションへの貢献(20%)を総合的に評価する。入門授業であるため、プログラムのクオリティそのものよりも、考察やコメントによる思考過程のプレゼンテーションをより重視する。プログラミングとは何かを理解し、プログラムが作れるようになっていく「学習過程の磨き方そのもの」を最も評価する。

## ■ テキスト・教科書

適宜、参考となる図書を紹介する。

## ■ 講義指定図書

■ ■ 参照ホームページ

■ ■ 研究室のホームページ

[http://art.ist.hokudai.ac.jp/~takigawa/index\\_ja.html](http://art.ist.hokudai.ac.jp/~takigawa/index_ja.html)

■ ■ 備考

プログラミングに興味があるが、(1)ほとんど、あるいは、全くしたことがない人、(2)してみたが理解がふわっとしてよくわからなかった人、(3)したことがあるが自信がないので考え直したい人、(4)既にもものすごくできるがなぜか急に入門に参加してみたくなった人(プログラミングを学ぼうとする同志をサポートしたい人)、(5)その他すべての人、を対象としています。そのため、プログラムが最初からきちんと作れることは期待されていませんし、強く評価にもしません。むしろ、プログラミングとは何かを実習しながら考え、うまく行かなかったところがなぜなのかを考え、仕組みを理解し、美しいプログラムを作る難しさを体感しながら、実際に作れるようになっていく学習過程をサポートしたいと思っています。「アート」な「呪文」の世界へようこそ。

■ ■ 更新日時

2018/01/27 19:18:18