2022年度・鼓山塾

K-15　デジタル社会の技術（プログラミング）
講義概要，プログラミングとは

伊東栄典

九州大学情報基盤研究開発センター

ito.eisuke.523@m.kyushu-u.ac.jp

本コースの目標

- 学んで欲しいこと
  - 将来，本・講義動画・オンライン教材を見て，**自分で学習を進められる**ようになって欲しい。
  - プログラム学習の**最大障壁**は「**作成・実行環境の準備と使い方**」。
    - どう作れば良いのか？
    - どうやって動かせば良いのか？
- 目標
  - プログラミング**作成・実行環境**「**Google Colab**（グーグル　コラボ）」**の使い方**を学ぶ。
    - 使い方がわかれば，自学自習できる。
  - プログラミングの**基礎学習と演習**を通して、初級プログラミングスキルを修得する。

講義の内容

- 1回目：
  - Google Colabの使い方，
  - Python言語でのプログラム記述と実行
  - 変数，四則演算
- 2回目：
  - 変数，四則演算
  - 文字列，リスト
  - 数学的なグラフ
- 3回目
  - if文による条件分岐
  - for文による繰り返し処理を体験

参考資料

Python ゼロからはじめるプログラミング
著者　　：三谷純
出版社　：翔泳社
発売日　：2021/5/24
ISBN　　：9784798169460

講義用のスライドも提供
https://mitani.cs.tsukuba.ac.jp/book_support/python/

本資料も，上記のスライドを援用しています。

無料で読めるPythonプログラム教材

- 東京大学「Pythonプログラミング入門」
  - Web：https://utokyo-ipp.github.io/
  - PDF：https://utokyo-ipp.github.io/IPP_textbook.pdf
- 明治大学 生田メディア支援事務室「Python入門 テキスト」
  - https://www.meiji.ac.jp/isys/doc/seminar/Python_text.pdf
- 京都大学「プログラミング演習 Python 2021」
  - https://repository.kulib.kyoto-u.ac.jp/dspace/bitstream/2433/265459/1/Version2021_10_08_01.pdf

上記の資料で，自学自習できます。(^_^)

1. はじめに

1. はじめに
2. プログラムとは
3. プログラムの構造

1. はじめに：プログラミングを学ぶ意義

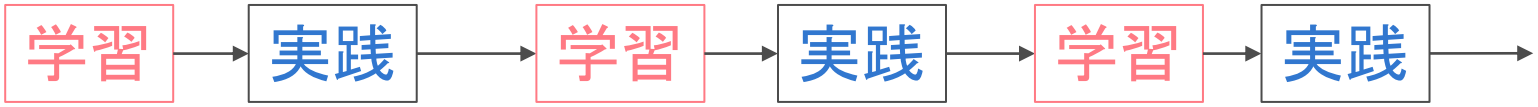
- ソフトウェアを使う立場から、作る立場へ
  - パソコン，スマートフォンなどで動作するアプリ
  - TV，エアコン，洗濯機などの電子機器の制御
  - 電子決済，電子申請，各種のシステム
- 個人での簡単な開発
  - 電卓・Excelの一歩先
  - データ処理・データ解析
  - 個人用途のアプリ開発
- スキルアップ
  - 情報処理関係の資格取得
  - プログラミングコンテスト

プログラミングを学ぶ意義（2）

- 実際に開発する立場になる予定が無くても
- アルゴリズム的思考（ものごとを処理する手順に関する合理的な考え方）が身につく。
- ソフトウェアの開発の様子、動作原理がわかることによる広い視野を獲得できる。
- 専門用語の理解、ITエンジニアとのコミュニケーション。
- 将来にプログラミングを独習したくなったときに役立つ。
  - 異なるプログラミング言語でも考え方の基本は同じ。

より良く学ぶために

- 以下を繰り返そう
  - 実際に手を動かしてプログラムコードを記述する。
  - 一部を変更して実験する
- 他にも
  - Web公開されているプログラムコードを動かしてみる
  - 本に記載さてれいる　　〃
  - 一部を変更して実験する
- 更に,
  - プログラミングコンテストに参加してみる
    （モチベーションアップ、実力向上、他者のコードからの学び）



自学自習のための「**写経**」

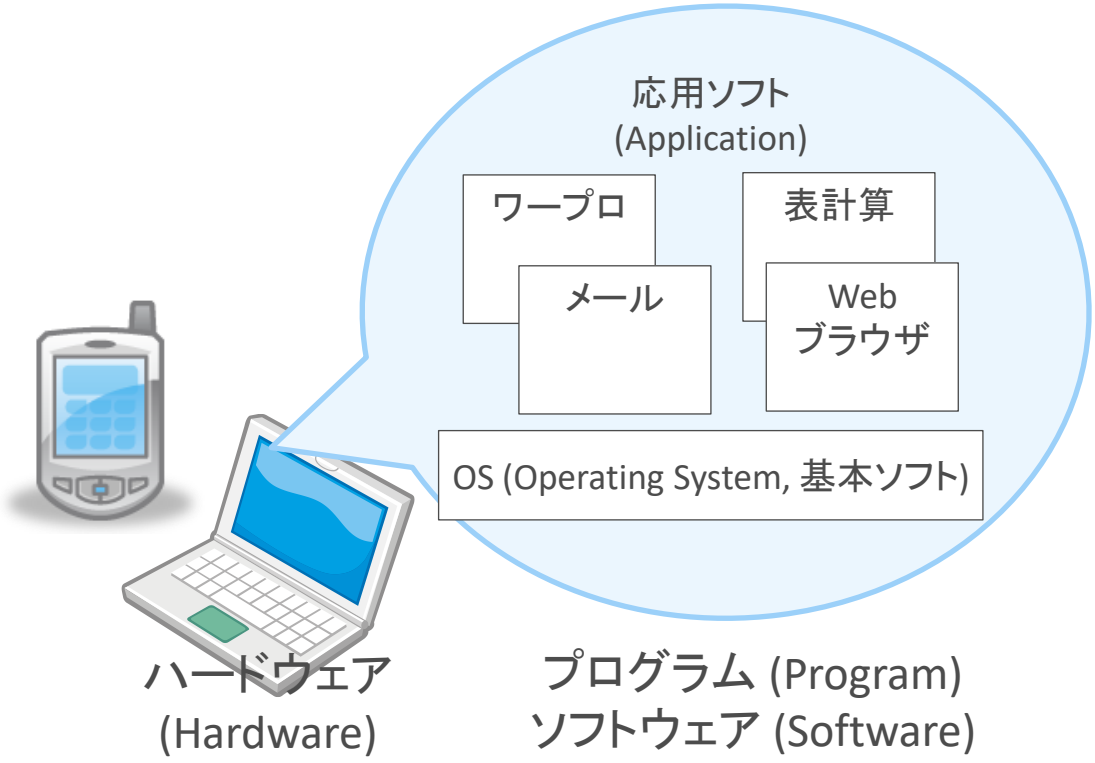
- そもそも「写経」とは、前述の通り「書き写すこと」です。
- **プログラミングにおける写経**
  - ネットや本に載っているサンプルコードを,
  - テキストエディタやプログラミングツールの上に,
  - **ひたすら手で打ち込む作業**のことを指す。
- **学ぶは真似ぶ**
  - 真似をしているうちに上達する。
  - 手を使って写すだけで学習効果が有る。

2. プログラムとは

1. はじめに
2. プログラムとは
3. プログラムの構造

2. プログラムとは

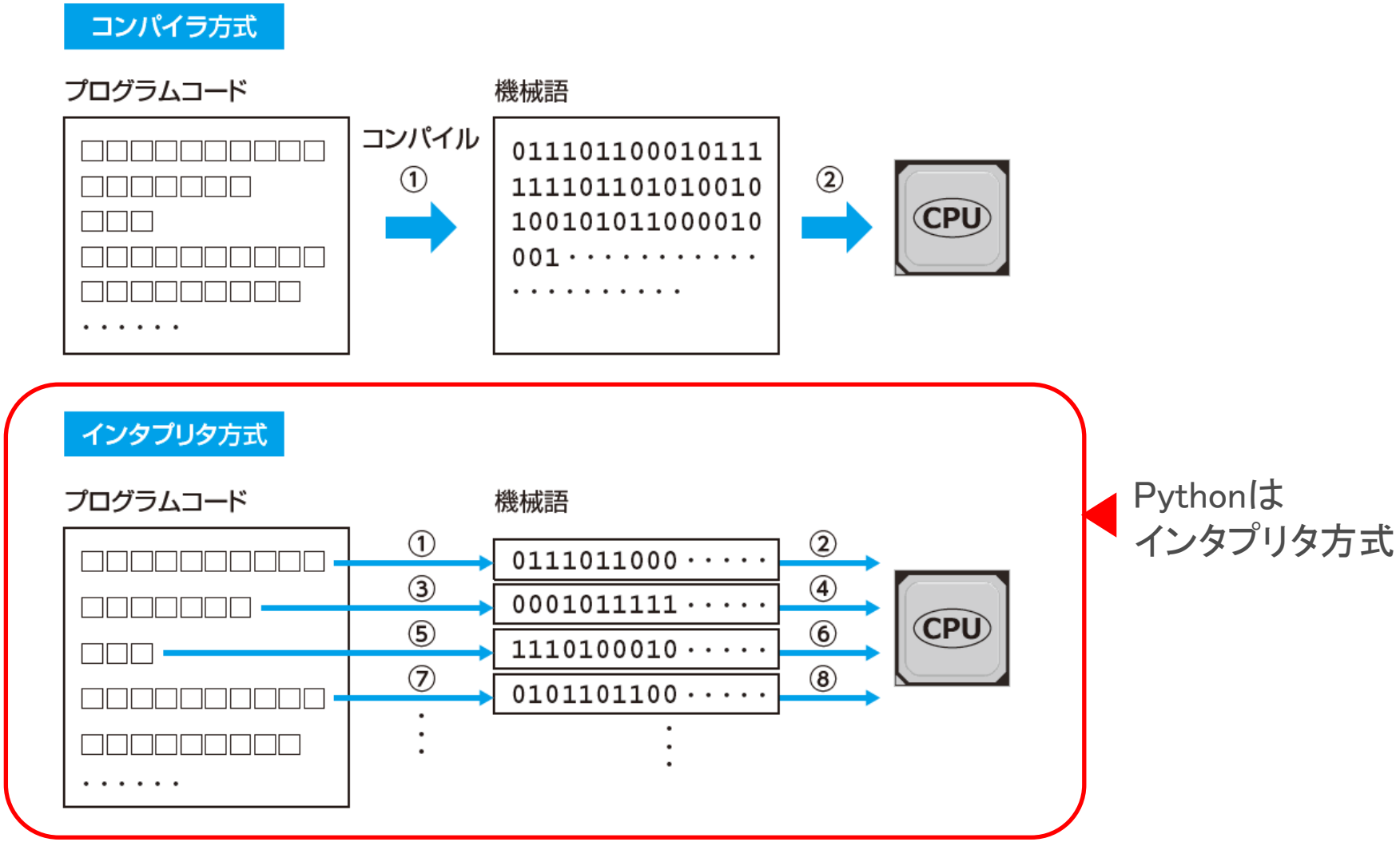
- プログラム　　コンピュータに命令を与えるもの
- プログラミング言語　プログラムを作成するための専用言語
- Python言語　　プログラミング言語の１つ



さまざまなプログラミング言語

言語	説明	備考
C	歴史のある言語	組み込みプログラム
C++	C言語の後継	オブジェクト指向
C#	C++ の後継	マイクロソフト
Perl	スクリプト言語	手軽な開発
PHP	サーバサイド	Webページ生成
Java	オブジェクト指向	大規模システム
JavaScript	ブラウザで動作	動的なWebページ
Python	**修得が容易**	**AI, 機械学習で普及**

プログラムコードが実行されるまで



Pythonのプログラム例

- 1から100までの整数を順番に足しあげて，その結果を画面に表示するプログラムのプログラムコード

```
total = 0
for i in range(1, 101):
    total += i
print(total)
```

- 半角英数と記号で記述する
- 人が読んで理解できるテキスト形式

2通りの実行方法

1. プログラムコードを記述したファイルを作成し、そのファイルを与えて実行させる
   - 一括モード (Batch mode)

2. プログラムコードを1行ずつ与え、そのつど実行させる
   - 対話モード (Interactive mode)

3. コードブロック毎に実行する
   - 対話モードの拡張版
   - 本講義で使う「ノートブック」はこの実行形式

3. プログラムの構造

1. はじめに
2. プログラムとは
3. プログラムの構造

3. プログラムの構造

- アルゴリズムとプログラミング

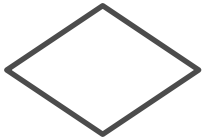
- Programming
  - プログラミング言語を用いて，プログラムを作成すること。
  - 作成したプログラムは，計算機で実行させることが出来る。

- Algorithm
  - 問題を解決するための方法や手順
  - Wikipedia「アルゴリズム」
    「計算可能」なことを計算する、形式的な（formalな）手続きのこと、あるいはそれを形式的に表現したもの。
  - アルゴリズムは，流れ図などで図式化すると分かりやすい。

流れ図 (Flow Chart)

- 以下の記号を繋げて，処理手順を図解したもの。

記号	名称	内容
	端子	開始と終了
	データ	データ入出力
	処理	演算などの処理
	表示	画面などに表示する

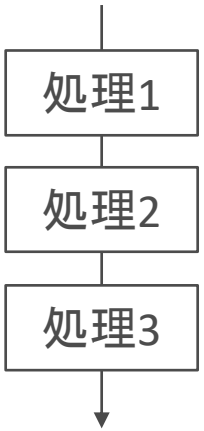
記号	名称	内容
	判断	条件による分岐
	ループ端	ループ（繰り返し）の始まり
		ループ（繰り返し）の終わり
	線	データや制御の流れ

制御構造と流れ図

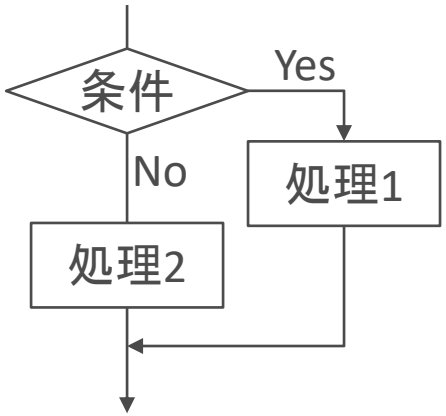
アルゴリズムおよびプログラムの制御構造は3種類

制御構造s	内容
順次 (Sequence)	1つ1つの処理を順番に行う
分岐 (Branch, Jump)	ある条件に応じて異なる処理を実行する
反復 (Iterative, Loop)	ある条件が満たされている間はその処理を繰り返し実行する

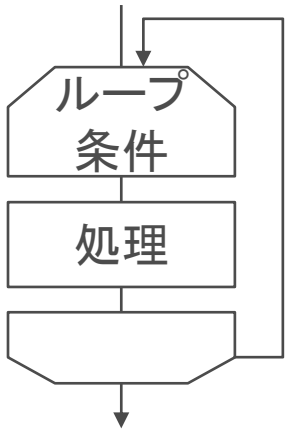
順次



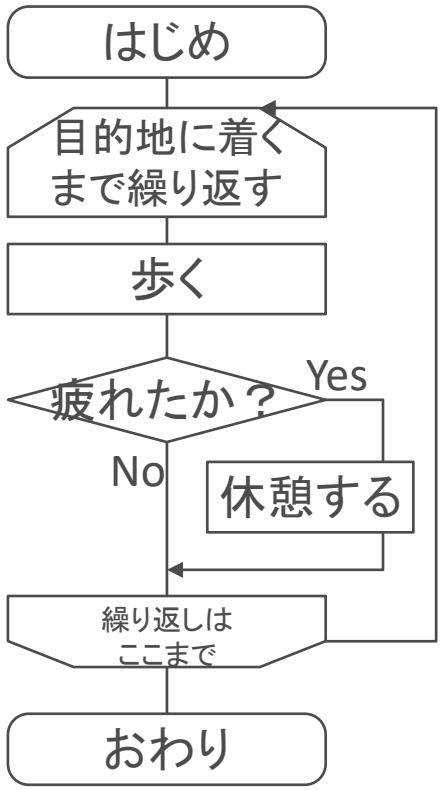
分岐



反復

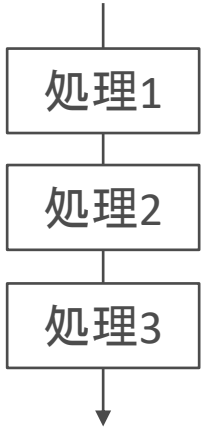


例：歩行



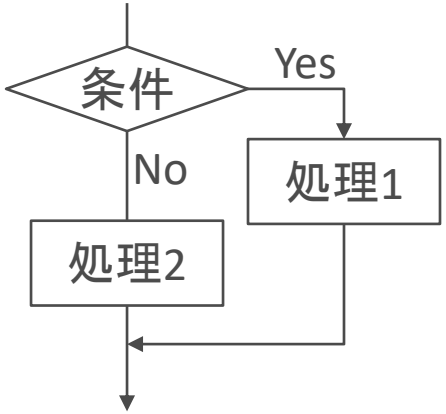
制御構造とプログラム例 (Python言語)

順次



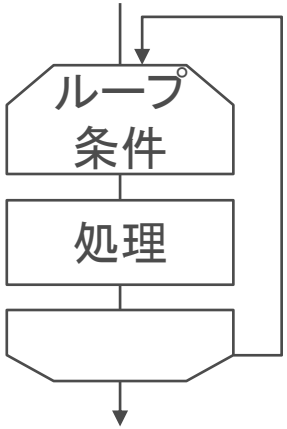

```
print(“Hello World”)
x = 1+2
print(x)
```

分岐




```
age = 16
if(age>=18):
  print(“You can vote”)
else:
  print(“You can’t”)
```

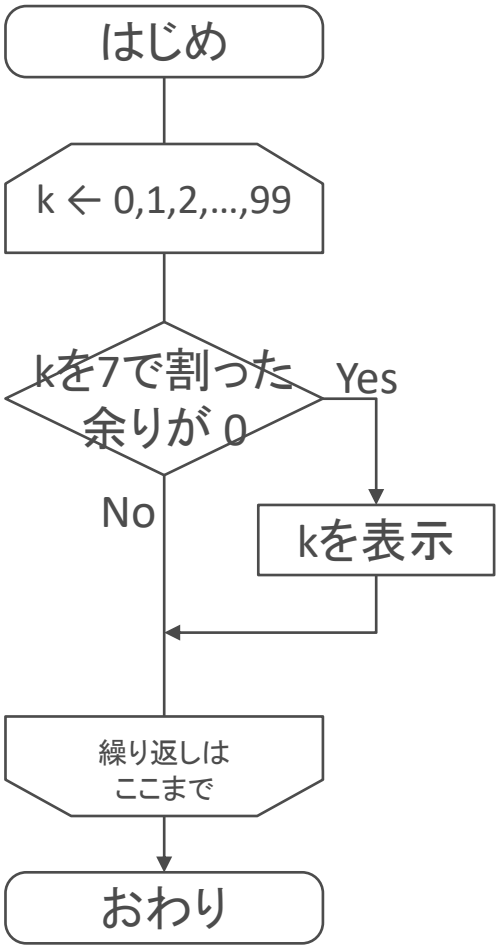
反復




```
sum = 0
for x in range(0..10):
  sum = sum + x

print(“合計 ”, sum)
```

例：0から99までの整数のうち，
7で割ったら余りが0となる
数を表示



Pythonプログラム

```
for k in range(100):
    if(k%7==0):
        print(k)
```

関数 (Function)

- 複雑なプログラムでは，コード行数が長くなり，制御の流れが分かりにくくなる。
- 特定の処理を「関数」という箱にかためる記述が楽に。

円の半径長を入力すると，その円の面積を出力するプログラム (python)

関数

```
def circle(r):
    return r*r*3.14
```

Main

```
r = int(input("円の半径は？ r="))
s = circle(r)
print("円の面積=", s)
```