

第9講 プログラミング入門

教材のダウンロードは以下URLにアクセスし、印刷して受講すると効果的です。
赤字を書き込んでいくとより効果的です！！！！

<https://joho-tarou.com/>

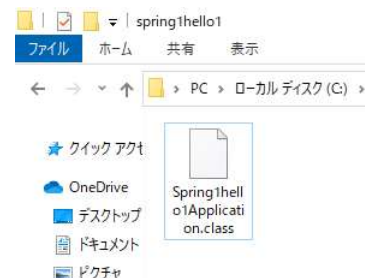
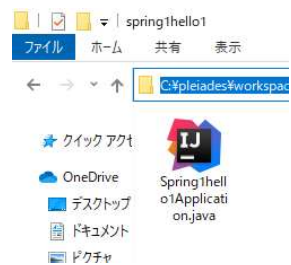


JOHO 太郎「情報 I」スーパー講義

9-1 プログラムの仕組み

コンピュータのハードウェア、ソフトウェアは**プログラム**によって動作しています。

■ プログラムを作る流れ



プログラム言語
を使ってプログラムを作る

ファイルとして
コンピュータの中に
ソースコードを保存

コンピュータが実行可
能なファイルに変換

コンピュータが実行可
能なファイルができる

プログラムが動く！



テストをした時に
見つかった不具
合を「バグ」とい
います。
このバグを取り除
く作業を「デバ
グ」といいます。



プログラム言語の中には、機械語に変換するコンパイルをファイル単位でまとめて変換する、**コンパイラ方式**と、1行ずつ機械語に都度変換して実行される**インタープリター方式**（機械語のファイルができない）に分かれます。

9-2 プログラミング言語

言葉には様々な国の言葉があるように、プログラミング言語も多くの種類があります。それぞれには特徴があり、最適なプログラミング言語を選択する必要があります。

プログラミング言語 人気ランキング

第1位

第2位

第3位

第4位

第5位



本講義では、**Python**を通じて、
プログラミング言語の基本を学んでいきます。
PythonはAIの開発で使用されています。

重要!

以降の講義は演習しながらになります。
必ず、聞いただけでなく実際にプログラムを入力しましょう
<https://colab.research.google.com/>

9-3-1 プログラミング入門 (変数、繰り返し)

プログラミング言語では、変数を定義し、繰り返しを使って、様々な処理を作ります。

```
hensu1=1  
hensu2=2  
print(hensu1 + hensu2)
```



出力結果： 3

※printは、pythonの画面表示用コマンド

POINT



変数を定義し、値を設定することができます。

hensu1 = 1 右から左に入れるイメージ！



POINT



変数同士で計算することができます。

変数に文字を入れると計算できず、プログラムはエラー。

```
for henCount in range(3):  
    print ('Count:', henCount)
```



出力結果： Count: 0
 Count: 1
 Count: 2

POINT



繰り返しを定義し、処理を指定回数繰り返すことができます
「ループ処理」とも呼ばれます。

※rangeは、pythonで指定した数分要素を持つというものです

※print で複数の値を指定すると一つずつ出力します

9-3-2 プログラミング入門 (条件分岐)

プログラミング言語では、条件分岐を使って、様々な処理を作ります。

```
hensuIn=int(input('1か2を入力してください:'))
if hensuIn==1:
    print('1が入力されました')
elif hensuIn==2:
    print('2が入力されました')
else:
    print('1、2以外の値が入力されました')
```



出力結果：

```
1か2を入力してください:1
1が入力されました
```



if 条件1 else if 条件2 else 条件3 で条件に応じた処理を書く事が出来ます。



前ページの「繰り返し(ループ処理)」と「条件分岐」を使いこなすことがプログラミングの基本となります。記述されているプログラミング言語によって書き方は微妙に異なりますが、**考え方は同じ**であり、**言語に関わらず、処理が理解できるようになる事がテストでは必要!**

※inputは、pythonでキーボードに入力した値を取得するコマンドを表します。

9-3-3 プログラミング入門 (配列)

プログラミング言語では、配列(リスト)を使うと1つの変数に複数の値を格納できます。

```
# hensuListという配列を定義
hensuList = [1, 2, 3, 4, 5]
# 配列の合計値を格納する変数を定義
goukei = 0

for i in range(len(hensuList)):
    goukei = goukei + hensuList[i]

print('合計:', goukei)
print('平均:', goukei/len(hensuList))
```



プログラミング言語には、コメントという機能があり、プログラムの説明を記載することができます。
プログラム処理はなされない、単なるメモです。
pythonの場合は#で始めるとコメントになります。



配列は、**変数[数字]** と指定することで、それぞれの値を取り出すことができます。(数字は番目と考えると判りやすいですね。**0番目から始まる。**)



出力結果 : 合計: 15
 平均: 3.0

※len(配列の変数) は、pythonでは配列に格納された個数を取得するコマンドです。

9-3-4 プログラミング入門 (関数)

プログラミング言語では、まとまった処理を一つの**関数**として定義する事ができます。

```
# 円の面積を求める関数を定義
def en_menseki(hankei):
    return hankei*hankei*3.14

# 定義した関数を使用した例
hankei=int(input('半径を入力してください:'))
menseki=en_menseki(hankei)
print('円の面積は', menseki)
```



出力結果：
半径を入力してください:10
円の面積は 314.0



関数のメリット

- ✓ 複雑な長いプログラムを分割することで整理できる
- ✓ 作成した関数を再利用することができる



イメージとしては、数学の関数と同じ！

$$y = f(x)$$



関数を外部から簡単に呼び出すことができるようにしたものを と呼びます
公開されているAPIを使うことで自分が作成したいプログラムで利用することができます。

9-3-5 プログラミング入門 (乱数)

プログラミング言語では、ランダムな数値を発生させる処理を持っています。シミュレーション等で使用できます。

```
# 乱数を使用するためにrandom(ライブラリ)をインポート
import random
# サイコロの目を示す変数にランダムに値を設定
saikoro=random.randint(1, 6)

print('サイコロの目は',saikoro)
```

⇒ 出力結果：サイコロの目は 4

※ random.randint(1, 6)は、pythonでは、1以上から6以下の数値をランダムで取得するという関数になります。



ある一定の範囲の中で、全ての数が同じ確率で発生する数を**乱数**といいます。プログラム内では乱数を作り出すことができます。



プログラミング言語では、よく使用する様々な**関数**を集めた**ライブラリ**を持っています、使いたいライブラリを参照することで、自分のプログラムの中で利用することができます。

9-4 大学入学共通テストに向けて

共通テストでは、特定のプログラム言語の形式では出題されません。

```
(01) Tomei = ["A党", "B党", "C党", "D党"] 配列
(02) Tokuhyo = [1200, 660, 1440, 180]
(03) Tosen = [0, 0, 0, 0]
(04) tosenkei = 0 変数
(05) giseki = 6
(06) m を0 から ア まで1ずつ増やしながら繰り返す:
(07) | Hikaku[m] = Tokuhyo[m]
(08) | セ < giseki の間繰り返す: 繰り返し
(09) | | max = 0
(10) | | i を0 から ア まで1ずつ増やしながら繰り返す:
(11) | | | もし max < Hikaku[i]ならば: 条件分岐
(12) | | | ソ
(13) | | | maxi = i
(14) | | Tosen[maxi] = Tosen[maxi] + 1
(15) | | tosenkei = tosenkei + 1
(16) | | Hikaku[maxi] = 切り捨て( タ / 子 )
(17) k を0 から ア まで1ずつ増やしながら繰り返す:
(18) | 表示する(Tomei[k], ":", Tosen[k], "名")
```

図9 各政党の当選者数を求めるプログラム

重要!

「変数」、「配列」、「繰り返し」、「条件分岐」の概念を理解しておくことが大事！



共通テストの形式は、本講義で説明している「python」に似ている！

【出典】令和3年3月公表
文部科学省「情報」サンプル問題 第2問 問3

第9講 確認問題 問1

共通テストの合計点を元に出願する大学を表示するプログラムを作りました。以下のプログラムを実行した場合、表示されるものを一つ選びなさい。

```
Kamoku = ['国', '数', '英', '社1', '社2', '理']  
Tokuten = [160, 180, 180, 80, 80, 80]
```

```
GoukeiTokuten = 0
```

i を0から5 まで繰り返す:

```
GoukeiTokuten = GoukeiTokuten + Tokuten[i]
```

もし GoukeiTokuten が 780 以上ならば:
表示する ('東大出願OK')

もし GoukeiTokuten が 740 以上ならば:
表示する ('一橋大出願OK')

もし GoukeiTokuten が 700 以上ならば:
表示する ('千葉大出願OK')

その他であったら:
表示する ('インターネットで確認しましょう')

- ① 東大出願OK
- ② 一橋大出願OK
- ③ 千葉大出願OK
- ④ インターネットで確認しましょう

第9講が完了しました！

Good job!

チャンネル登録をお願いします！