

2019年12月27日(金)情報科実践事例報告会

# Pythonで実践プログラミング



やなとり しんぺい  
神奈川県立厚木西高等学校 梁取 新平

# 何を教えるプログラミング

五大装置 論理演算 論理回路 真理値

誤差 計算誤差 オーバーフロー

センサ アクチュエータ

アルゴリズム 制御構造 順次 分岐 反復 変数 流れ図

リスト 乱数 関数 引数 戻り値 WebAPI

線形探索 二分探索 選択ソート クイックソート

動的モデル 静的モデル 確定モデル 確率モデル

シミュレーション モンテカルロ法 一様乱数

自然現象 放物運動 生命体増殖 ランダムウォーク

# 何を教えるプログラミング

## 目的その①

コンピュータがやっている

ことを想像できる

# 何を教えるプログラミング

## 目的その②

仕事を**バラバラ**に分けて  
考えることができる

何を教えるプログラミング

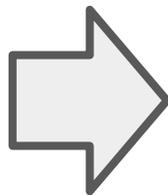
目的その③

わからないことも

**予測できる**ことに気づく

# 何を教えるプログラミング

コンピュータ  
仕事  
現象



# 何を教えるプログラミング

わからないものを解明する**タノシミ**

+ 試行錯誤してうまく動いた**ヨロコビ**

+ 自分が作ったアプリを  
他人に動かしてもらおう**ドキドキ**

# どうしてPython？

- ・英語で入力
- ・インタプリタ
- ・情報が多い



# どうしてPython？



Python  
ふつうの人

モジュール  
(ライブラリ)  
呼び出し



どんなプログラムにも  
変身が可能

どうしてPython？

Python + Numpy = 数値計算

Python + Scipy = 科学計算

Python + Matplotlib = グラフ

どうしてPython？

Python + Tkinter = GUIアプリ

Python + Pygame = ゲーム

Python + PyInstaller  
= 実行(EXE)ファイル作成

# Pythonで授業

- ・3年選択「情報の科学」

- ・開発ツール

- ①Google Colaboratory

- ②WinPython(Anaconda)

# Pythonで授業

## ①Google Colaboratory



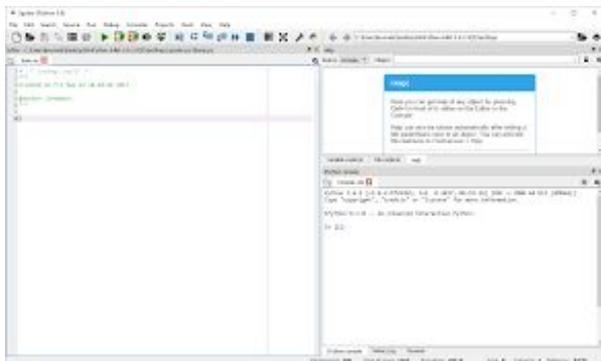
- ・ウェブで動作

- ・Chromebook

- ・導入時に○

# Pythonで授業

## ②WinPython (Anaconda系)



- ・統合開発環境
- ・とにかく重い
- ・EXE化できる

# Pythonで授業

## 1) 数当てゲーム

(分岐、反復、乱数、入出力)

## 2) ユークリッドの互除法

(関数の作成と利用)

# Pythonで授業

## 3) 高校数学

### に挑戦

(ライブラリの利用)

```
In [28]: fvx = a + sympy.sin(d) * cf / m * ((vy+Lf*w)  
fvx
```

```
Out[28]: a + \frac{cf}{m} \left( -d + \frac{1}{vx} (Lfw + vy) \right) \sin(d) + v
```

```
In [29]: sympy.diff(fvx,vx)
```

```
Out[29]: -\frac{cf}{mvx^2} (Lfw + vy) \sin(d)
```

```
In [30]: sympy.diff(fvx,vy)
```

```
Out[30]: \frac{cf}{mvx} \sin(d) + w
```

# Pythonで授業

## 4) タイピングゲーム

- ・ **GUI**アプリの作成
- ・ **実行ファイル**化と相互評価



# Pythonで授業

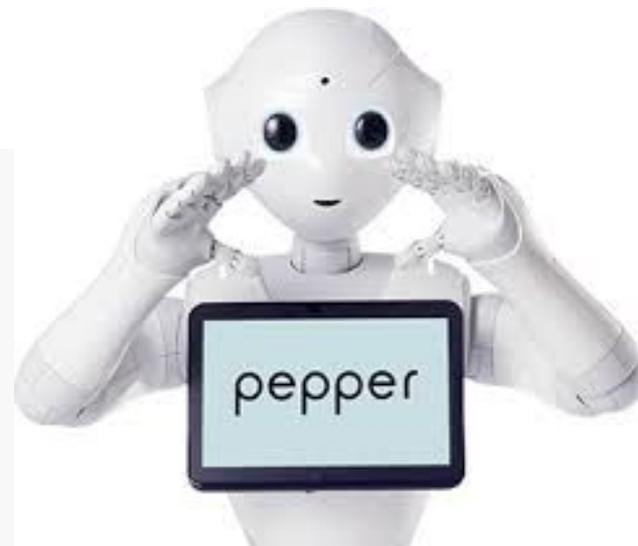
# Pythonの事例の一部



**YouTube**



**Instagram**



**Pepperの  
感情エンジン**

# 何を教えるプログラミング

五大装置 論理演算 論理回路 真理値

誤差 計算誤差 オーバーフロー

センサ アクチュエータ

アルゴリズム 制御構造 順次 分岐 反復 変数 流れ図

リスト 乱数 関数 引数 戻り値 WebAPI

線形探索 二分探索 選択ソート クイックソート

動的モデル 静的モデル 確定モデル 確率モデル

シミュレーション モンテカルロ法 一様乱数

自然現象 放物運動 生命体増殖 ランダムウォーク

# 今後のPython実践

- ・統計ツール作成
- ・生徒がいじれる ”余白”
- ・生徒を惹きつける教材作り



ご清聴ありがとうございました

## 20191227実践事例報告会「Pythonで実践プログラミング」サンプルコード

### (1) Python基本項目

# 文字の出力

```
print("情報の科学")
```

# 変数

```
a = 2
```

```
print("情報の科学" + str(a) + "学期")
```

# 文字の入力

```
b = int(input("誕生月を入れて : "))
```

```
b
```

# 乱数

```
import random
```

```
c = random.randint(1,100) #1から100の中から1つcに代入する
```

```
c
```

# 分岐 (選択)

```
d = 9
```

```
if d%2 == 0:
```

```
    print("偶数だよ")
```

```
elif d%2 == 1:
```

```
    print("奇数だよ")
```

# 反復 1 (有限の繰り返し)

```
for e in range(5):
```

```
    print("おいしい")
```

# 反復 2 (無限の繰り返し)

```
while(True):
```

```
    break
```

## (2) 数当てゲーム

```
import random
ans = random.randint(1,5)
i = 0
print("1から5までの間の数字を予想してね!")

while(True):
    i+=1
    num = int(input("予想した数字を入力してね! : "))
    if num == ans:
        break
    elif num > ans:
        print("大きいよ!")
    elif num < ans:
        print("小さいよ!")

print("\n当たりだよ!" + str(i) + "回で当てられたね")
```

## (3) ユークリッドの互除法

```
a = input('a=')
b = input('b=')
if a < b:
    a,b = b,a

def gcd(x,y):
    if y == 0:
        return x
    while y > 0:
        x %= y
        x,y = y,x
    if y == 0:
        return x

print(a,'と',b,'の最大公約数=',gcd(int(a),int(b)))
```

#### (4) 顔の認識

```
import cv2
cascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
img = cv2.imread('photo.jpg')
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
faces = cascade.detectMultiScale(gray, scaleFactor=1.5)

for(x,y,w,h)in faces:
    cv2.rectangle(img,(x,y),(x+w,y+h),(0,0,255))

cv2.imwrite("photoM.jpg",img)
```

## (5) 高校数学に挑戦

# 共通 (式の表示をTeXに沿って整形します)

```
from sympy import *
from google.colab.output._publish import javascript
javascript(url="https://cdnjs.cloudflare.com/ajax/libs/mathjax/2.7.3/latest.js?config=default")
init_printing(use_latex='mathjax')
```

# 式の展開

```
a,b,c = symbols('a b c')
ex1 = (a-b)*(b-c)*(c-a)
expand(ex1)
```

# 因数分解

```
x,y = symbols('x y')
ex2 = (x**4-7*(x**2)*(y**2)+y**4)
factor(ex2)
```

# グラフの作成

```
x = symbols('x')
plot(x**3)
```

# 解を求める

```
x = symbols('x')
ex3 = x**2-2*sqrt(3)*x+4
solve(ex3,x)
```

# 微分

```
x = symbols('x')
ex4 = x**4 + 2*x**3 + 3*x + 4
diff(ex4)
```

# 積分

```
x = symbols('x')
itg = Integral(1/sqrt(x**2+1))
rslt1 = itg.doit()
rslt1.rewrite(log)
```

## (6) タイピングゲーム

```
# -*- coding: utf-8 -*-
import random
import sys
import pygame

def select_word():
    word_list = ['apple','banana','cherry','melon','orange']
    num_of_elements = len(word_list)
    i = random.randint(0,num_of_elements - 1)
    return word_list[i]

def cut_head_char(word):
    return word[1:]

def is_empty_word(word):
    return not word

def main():
    pygame.init()
    screen = pygame.display.set_mode((720,480))
    bg = pygame.image.load("haikei.jpg").convert_alpha()
    rect_bg = bg.get_rect()
    pygame.mixer.music.load("sample.wav")
    pygame.mixer.music.play(1)
    font_big = pygame.font.SysFont(None,150)
    word = select_word()
    while True:
        screen.fill((200,200,200))
        screen.blit(bg,rect_bg)
        sf_word = font_big.render(word,True,(0,0,0))
        center_x = screen.get_rect().width / 2 - sf_word.get_rect().width / 2
        screen.blit(sf_word,(center_x,200))
        pygame.display.update()
        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                pygame.quit()
                sys.exit()
            if event.type == pygame.KEYDOWN:
                if chr(event.key) == word[0]:
                    word = cut_head_char(word)
                    if is_empty_word(word):
                        word = select_word()

if __name__ == '__main__':
    main()
```

### <ご注意>

○すべてのサンプルにおいて、動作を保証するものではありません。

○掲載スペースの都合上、長いコードが改行されてしまっている場合があります。ご容赦ください。

○プログラミングの際、あらかじめプログラム上で使用するモジュール（ライブラリ）が開発環境に導入されているか確認してください。

導入されていない場合、pip命令などで該当のライブラリを導入してから、プログラムを実行してください。

### <参考文献>

- ・日向俊二「やさしいPython入門 第2版」カットシステム.2018
- ・日経ソフトウェア編「いろいろ作りながら学ぶ！Python入門」日経BPマーケティング. 2019
- ・田中賢一郎「ゲームを作りながら楽しく学べるPythonプログラミング」インプレス R&D.2017