

神奈川県高等学校教科研究会情報部会
令和4年度実践事例報告会

2022/12/26

創造的写経プログラミング授業のすすめ

埼玉県立川越南高等学校
情報科 春日井 優

自己紹介

春日井 優（かすがい ゆう）

埼玉県立川越南高等学校 情報科（数学科で採用）

東洋大学総合情報学部 非常勤講師（2013年、2016年、2019年、2022年）
情報科指導法Ⅰ・Ⅱ、情報化社会と職業、情報化社会と倫理

授業実践発表

モデル化とシミュレーションの活用による問題解決を行う協働学習
テキストマイニングと機械学習の活用による問題解決を行う協働学習
データの活用（カイ2乗検定）の活用による問題解決を行う協働学習

執筆

文部科学省高等学校情報科「情報Ⅰ」「情報Ⅱ」研修用教材
「情報Ⅱ」教科書（実教出版）など

創造的写経プログラミング授業のすすめ

はじめに

創造的写経プログラミング？

辞書によると

創造：**新しいものを初めて作り出す**こと

的： 名詞に付いて、形容動詞の語幹をつくる

㊦ そのような性質をもったものの音をます

㊧ (略)

㊨ そのようなよすの、そりつ、などの意を表す

相反する考え方

写経：**経文を書写すること**。また、書写した経文。

仏典の保存や仏典書写による功德などを目的とする

(goo辞書より)

写経プログラミングにありがちな生徒の感想

プログラミングをたくさん経験して、
文字を速く打つことができるようになりました。

写経：経文を書写すること。また、書写した経文。

仏典の保存や**仏典書写による功德**などを目的とする

(goo辞書より)

プログラミングを学習する目的?????

プログラミングの学習

プログラミング教材は、正解を答える形式が多いのでは？

→ 傍用問題集、ネット教材など

Scratchを使ったプログラミングでは、
いろいろ試して創造的なものが作れたのに・・・



プログラミングの学習

問題として与えられるので、正答できるようにするだけ。

プログラミングも正解主義になってしまう？

プログラミングの問題を解けるようになりました！
でもプログラムを作って役に立つんですか？



そういえば、
同じ発想の授業実践があったはず！



キミのミライ発見 事例59

乱数を利用するJavaScriptプログラミング授業の実践
～プログラミングで「好きなもの」を作る活動の工夫
東京都立町田高校 小原 格先生

<https://www.wakuwaku-catch.net/jirei1759/>

創造的写経プログラミング授業のすすめ

授業の概要

単元での授業

時限	項目	具体的な内容
夏休みの宿題	Progateでの予習	基本構造、リスト、辞書、関数
1時限	基本構造とリスト(配列)	最寄駅からの運賃を表示する
2時限	関数の利用	各駅の乗降人員を■でグラフにする
3・4時限	簡単なアルゴリズム	最も乗降した人員を調べる 券売機で釣銭を金種ごとに枚数を調べる
5・6時限	基本的なアルゴリズム	線形探索・二分探索の考え方 交換法・選択法による整列の考え方を理解する
7時限	乱数を利用したプログラム	乱数を用いて、疑似サイコロを振る 乱数を用いて、くじ引きをする
8・9時限	創造的写経プログラミング	グループでプログラミングを行う 完成させて、発表する

授業環境

Googleのアカウントを1人1IDずつ割り当てている

Google Colaboratoryを利用可能

Google Classroomを使って、ベースとなるプログラムを配布

1～6時限の授業の流れ

Google Classroomを使って、ベースとなるプログラムを配布する

プログラムの流れ、処理の考え方を確認する

プログラムで特に理解したい箇所の空欄を補充する

1～7時限の授業の流れ

Google Classroomを使って、ベースとなるプログラムを配布する



プログラムの流れ、処理の考え方を確認する



プログラムで特に理解したい箇所の空欄を補充する

ここでは正解がある問題を考える

ここで生徒に示したプログラムが写経のもとになる

授業で生徒に示したプログラムの問題①

【問題】

南大塚駅から西武新宿線を利用したときの運賃を求めたい。南大塚駅から各駅までの距離は、下の表1のようになっている。また、西武鉄道の運賃体系は表2のようになっている。これらの表をもとに、空欄をうめてプログラムを完成させなさい。

表1 南大塚から各駅までの距離

駅名	距離
南大塚	—
新狭山	2.6
狭山市	5.3
入曽	8.3
新所沢	12.2
航空公園	13.4
所沢	15.0

表2 距離に対するきっぷでの運賃

距離	運賃
4.0km 以下	150 円
4.1km 以上 8.0km 以下	180 円
8.1km 以上 12.0km 以下	210 円
12.1km 以上 16.0km 以下	250 円

南大塚の駅に入場するには、入場券 150 円がかかる。

授業で生徒に示したプログラム①

【プログラム】南大塚駅から各駅までの運賃を表示するプログラム

```
station = ['南大塚', '新狭山', '狭山市', '入曽', '新所沢', '航空公園', '所沢']  
distance = [0.0, 2.6, 5.3, 8.3, 12.2, 13.4, 15.0]
```

```
for i in range(0, 7, 1):  
    if distance[i] <= 4.0 :  
        fare = 150  
    elif distance[i] <= 8.0:  
        fare = 180  
    elif distance[i] <= 12.0 :  
        fare = 210  
    else :  
        fare = 250  
    print(f'{station[i]}駅まで{fare}円です。')
```


授業で生徒に示したプログラム②

【プログラム】本川越駅から所沢駅までの乗降人員のグラフを表示するプログラム
(関数を使って処理を行い、戻り値がない場合)

```
def graph(name, number):          # nameは駅名、numberは乗降人員
    box = round(number / 2000)    # round(数値)で四捨五入できる
    print(name, end=':')         # end=':'を付けると改行しないで:を表示
    for i in range(0, box, 1):
        print('■', end='')
    print(number)

stations = ['本川越 ', '南大塚 ', '新狭山 ', '狭山市 ',
            '入曽 ', '新所沢 ', '航空公園', '所沢 ']
passengers = [ 43048, 14620, 15973, 32066,
               14168, 42857, 20480, 86613 ]

for i in range(0, 8, 1):
    graph(stations[i], passengers[i])
```

授業で生徒に示したプログラム③

リストを利用して、等しくない確率でくじの結果を出力するプログラム

```
import random
a = ['あたり', 'おいしい', 'はずれ']
r = random.random()
print(r)
if r < 0.2:
    print(a[0])
elif r < 0.5:
    print(a[1])
else:
    print(a[2])
```

他にもいくつかのプログラムを示している

創造的写経プログラミング授業のすすめ

創造的写経プログラミング

創造的写経プログラミングとは

創造的：授業で学習したプログラムをもとに
それを発展させて新しいプログラムを作り出すこと

写経：もともになるプログラムを模倣する
プログラムを模倣することによる
プログラミングの理解の深化を目的とする

要するに、学習したプログラムを発展させて
生徒自身のオリジナルのプログラムを作成すること

創造的写経プログラミングの課題

【課題】

これまでに学習したプログラムを改変したり、応用したりして、何かが便利になったり、何かの役に立つようなプログラムを作ってください。

【条件】

グループで1つのアイデアにまとめ、それをプログラムする
全員が必ず自分のPCで動作させる

(エラーが出たら、基本的にグループで解決する

+ やむを得ない場合には、教員からアドバイスする)

8・9時限の授業の流れ

学習したプログラムを改変してできることをグループで考える



プログラムを作成する（全員が自分のPCで動作させる）



プログラムを作成するにあたって決めた設定と
プログラムの動作を文章で記述させる

ここでは正解がない問題を考える

授業で学習したプログラムを

生徒がグループで考えて作る

生徒が作成したプログラム①

【プログラム】 夕飯のメニューを乱数で決めるプログラム（一言コメント付き）

```
import random

a = ['そば', 'グラタン', 'オムライス', 'チャーハン', 'おにぎりセット']
r = random.random()
print(r)

if r < 0.1:
    print(a[0])
    print('天ぷら一緒にどうですか。')
elif r < 0.25:
    print(a[1])
    print('エビグラタンにしませんか。')
elif r < 0.45:
    print(a[2])
    print('デミグラスorケチャップ')
elif r < 0.7:
    print(a[3])
    print('目指せパラパラ')
else:
    print(a[4])
    print('~味噌汁を添えて~')
```

生徒が作成したプログラム②

【プログラム】 10回連続でガチャを引くプログラム

```
import random
a = ['星5', '星4', '星3']
for i in range(0,10,1) :
    r = random.random()
    print(r)
    if r < 0.006 :
        print(a[0])
    elif r < 0.051 :
        print(a[1])
    else :
        print(a[2])
```


生徒が作成したプログラム③

【プログラム】連続してガチャを引くプログラム

(必ず15回のうちにレア度が高い星5が出るようにし

10回のうちにレア度がやや高い星4が出るようにしている)

(コメントは春日井が追加)

```
# ポイント
s=1600
# ポイントに応じたガチャを引ける回数
kai=s//160

import random

# 星5がgcou回続けて出ていないときに星5が出る確率
go =[0.006, 0.006, 0.006, 0.006, 0.006,
      0.006, 0.006, 0.006, 0.006, 0.006,
      0.006, 0.2,  0.2,  0.2,  1   ]

# 星4がycou回続けて出ていないときに星4が出る確率
yon=[0.05,  0.05,  0.05,  0.05,  0.05,
      0.05,  0.05,  0.05,  0.05,  1]
```

```
# 星5が出ていない連続回数
gcou=0

# 星4が出ていない連続回数
ycou=0

for i in range(0,kai,1):
    print(i+1,'回目')

    r=random.random()

    if r<go[gcou]:
        print('星5')
        gcou=0
        ycou=0

    elif r<yon[ycou]:
        print('星4')
        gcou=gcou+1
        ycou=0

    else:
        print('星3')
        gcou=gcou+1
        ycou=ycou+1
```

生徒が作成したプログラム④

【プログラム】 あっち向いてホイで
コンピュータと対戦するプログラム

```
#あっち向いてホイ
```

```
import random
```

```
a = ['右', '左', '上', '下']
```

```
r=random.random()
```

```
hito=input('好きな方向を入力してください！')
```

```
print(r)
```

```
if r < 0.2:  
    print(a[0])  
    pc=a[0]  
elif r < 0.5:  
    print(a[1])  
    pc=a[1]  
elif r < 0.8:  
    print(a[2])  
    pc=a[2]  
else:  
    print(a[3])  
    pc=a[3]  
  
if hito==pc:  
    print('勝ち')  
else:  
    print('負け')
```


学習の効果

- プログラムを使う状況を生徒が考え、あわせてそのプログラムを作るために必要な条件やパラメータを生徒が決めてプログラムをする
 - ⇒ 共通テスト試作問題でのリード文に書かれているようなことを生徒自身で考える機会になる
- プログラムを作るのに必要なロジックを主体的に考えられる
 - ⇒ アルゴリズムの理解が深まる
- 生徒は複雑なことを案外やりたいと考えている
 - ⇒ 予想以上に複雑で長いプログラムを作る