

# Python を利用した数学コンテスト問題への対応 —科学の甲子園全国大会の生徒解答事例—

愛知県立旭丘高等学校 田中 紀子

Tanaka Noriko, Aichi Prefecctural Asahigaoka High School

## 1 はじめに

科学の甲子園は、高等学校等（中等教育学校後期課程、高等専門学校を含む）の生徒チームを対象として、国立研究開発法人科学技術振興機構（JST）が、平成23年度より創設し、理科・数学・情報における複数分野の競技を行う取り組みである。全国の科学好きな高校生が集い、競い合い、活躍できる場を構築しつつ、科学好きの裾野を広げるとともに、トップ層を伸ばすことを目指しているコンテストである。

令和3年3月に実施された科学の甲子園全国大会における実技競技問題では、Python を利用して解く数学の問題が出題された。

本稿では、問題の概要と、生徒の解答のいくつかを示すとともに、今後のプログラミング教育の在り方について考えることを目的とする。

## 2 研究内容

### 2.1 科学の甲子園実技問題

#### (1) 事前公開資料

この実技競技では、事前公開資料が配布された。事前公開資料には、次のように書かれている（以下抜粋）。

当日出題される問題の6割程度はプログラムにより計算を行わなければ解答が得られません。この事前公開資料を熟読して事前にプログラミング環境に習熟しプログラミングを学習した上で、全国大会では実力が発揮できるように準備してください。

1. 競技人数 1 チーム 3 名
2. 競技時間 100 分
3. 競技概要

競技会場に用意されている競技に使用できるパソコン（以下、競技用 PC と呼ぶ）から実技競技サイトへログインして、情報に関する問題に取り組み 100 分間

で解答する。さまざまな分野から問題は出題される。問題の取り組む順番は自由である。正解したチームに得点が与えられる。問題の配点はそれぞれ異なる。出題される問題を解くには知識だけでなく、計算が必要となる。効率よく計算結果を得るためには、競技用 PC でプログラミングを行うこともできる。決められた時間内に、正解した問題で得られた合計得点を競う。

#### 4. 競技用 PC について

競技用 PC として 2 台配付する。

・競技用 PC それぞれにプログラミングができる環境がインストールされている（「5. プログラミング環境について」を参照）

- ・競技用 PC はインターネットに接続されていない。
- ・競技用 PC に新たにアプリケーションをインストールすることはできない。
- ・競技用 PC に外部記憶装置を接続することはできない。
- ・競技用 PC の OS は Windows10

#### 5. プログラミング環境について

競技用 PC には以下のプログラミング環境があり、使用することができる。

- ・プログラミング言語 Python3.8.5
- ・IDE※は Visual Studio Code、PyCharm、Anaconda(Jupyter Notebook、Spyder) が使用できる。

事前公開資料には、「引数に応じた階乗プログラムの例」と「階乗関数を使用したプログラム例」も掲載されていた。

本校の出場者は 3 週間で Python を独学で学び、大会に出場した。

## (2) 当日配布資料

当日配布資料は、次である (以下抜粋)。

### 1.1 競技趣旨

社会では科学技術上の問題を解決するにあたり、コンピューターを使って複雑な計算を行うことが多い。一方、理科・数学・情報で様々な科学現象を学ぶ機会では、プログラミングによって科学技術上の問題を解決する場面がまだまだ少ない。そこで、今回の実技競技では、プログラミングによる問題解決の有用性を感じてもらうことを意図している。

### 1.2 競技構成

競技は、下表に示す問題が出題される。問題は、どの問題から解いてもよい。

カテゴリー 問題の名称 配点

【素数】 問 1 100 問 2 200

【竹内関数】 問 100

【2 進法】 問 100

【和が決められた数列】 問 1 100 問 2 100 問 3 100

【約数】 問 1 200 問 2 200

【部分文字列】 問 300

【整数】 問 1 300 問 2 300 問 3 300

【ドレミ暗号】 問 1 100 問 2 200

【文字列操作】 問 300

【数列操作】 問 1 100 問 2 200

### (3) 生徒のプログラミング

問題のなかには、手計算でできるものと Python を使わないとできないものがあった。手計算の方が早いものも含まれていた。3人のグループの力を合わせて解く問題で、パソコンは2台だったため、生徒はどちらで解くと早いかを見極め、手分けをして解答をしていった。ここでは生徒が Python で行った解答をいくつか紹介する。

ア, 【素数】

<問題>

素数とは、約数が 1 とその数自身しかない 2 以上の自然数である。2 から 11 までの最初の 5 個の素数を印刷すると 2, 3, 5, 7, 11 であり、「1」は 2 回印刷される。2 から 997 までの最初の 168 個の素数を印刷すると「1」は何回印刷されるか、回数を求めよ。

<解答>

```
def primes(n):
    p = list(range(2,n+1))
    for i in range(2,n+1):
        p = [j for j in p if j % i != 0 or j == i]
    return p
print("".join(map(str,primes(997))).count("1"))
```

イ, 【竹内関数】

<問題>

竹内関数とは、3 個の引数  $x, y, z$  を持つ以下のように帰納的（再帰的）に定義される関数 Tarai である。

$x \leq y$  のとき,  $\text{Tarai}(x,y,z)=y$

それ以外するとき,

$Tarai(x,y,z)=Tarai(Tarai(x-1,y,z),Tarai(y-1,z,x),Tarai(z-1,x,y))$   
関数 Tarai を Python で以下のプログラム tarai として計算できるようにした。

```
def tarai(x, y, z):  
    if x <= y:  
        return y  
    else:  
        return tarai(tarai(x-1,y,z),tarai(y-1,z,x),tarai(z-1,x,y))
```

tarai(14,7,0) を計算するとき、上記の tarai が何回呼び出されるか、回数を求めよ。ただし最初の呼び出しは含めないものとする。

<解答>

```
def tarai(x, y, z):  
    tarai.count += 1  
    if x <= y:  
        return y  
    else:  
        return tarai(tarai(x-1,y,z),tarai(y-1,z,x),tarai(z-1,x,y))  
tarai.count = 0  
tarai(14,7,0)  
print(tarai.count - 1)
```

ウ、【部分文字列】

<問題>

文字をいくつか並べたものを引用符「」でくくって文字列を表すものとする。'aab' は長さ 3 の文字列、'' は長さ 0 の文字列である。ある文字列からいくつか文字を取り除いて順番を変えずに並べたものを、もとの文字列の部分文字列という。取り除くときは全部取り除いてもよいし 0 個取り除いてもよい。例えば 'aab' の部分文字列は '', 'a', 'b', 'aa', 'ab', 'aab' の 6 個であり、'abc' の部分文字列は '', 'a', 'b', 'c', 'ab', 'ac', 'bc', 'abc' の 8 個である。文字列 'abracadabra' の部分文字列の個数を求めよ。

<解答>

```
import itertools as itr  
s = "abracadabra"  
print(sum([len(set(itr.combinations(s,i)))\n            for i in range(0,len(s) + 1)]))
```

## 2.2 実技問題に関する生徒の感想

- 事前公開資料に「さまざまな分野から問題は出題される。」と書かれていたため、どのような分野の問題があるか不安だったが、実際はほとんど数学の問題で安心した。
- 【竹内関数】が一番難しいと感じた。答えが出るのに1, 2分かかり、正しいかどうか不安だった。
- 【整数】ゴールドバッハ予想の問題の大会後に発表された解答について…模範解答はあまりよくないと思う。素数の問題(問1, 問2)で作った式を解答に使うべきだ。

## 3 考察

プログラミングによる問題解決の有用性を感じてもらふ意図は達成できているか、言語はPythonでよかったか、高校生に必要なプログラミングのリテラシーは何かの3点について考察することで、プログラミングコンテストの在り方やプログラミング教育の在り方について考える一助としたい。

○「プログラミングによる問題解決の有用性を感じてもらふ意図」は達成できているか。

(1) ある程度は達成できている。

(2) コンテスト問題としては、「有用性を感じさせる」だけではなく、「何か(知識・技能あるいは資質・能力)の差異に対して順位をつける」視点が入るはず。それは何であったか(何であるべきか)。プログラミング的思考であるべきか。

(3) 出題者は「順位決定用の問題」を3問設けた。同点の場合順位決定用の問題を先に正解したチームを上位とするとした。出題者は100分で18問完答できる県代表はいないと考えたようだ(競技中の場内アナウンスによる)。「完答できたチームから順位を上位とする。完答できたチームがいなければ得点の高いものを上位とし、同点の場合は順位決定用問題を先に正解したチームを上位とする。」のほうが、参加者の理解が得られやすい。

(4) 問題のほとんどが数学的思考を要する問題であった。測れたものは、数学的思考力とPythonの活用力であるか。

○言語はPythonでよかったか。

(1) 本校では課題研究(情報)でBasicをやっている。多くの高校生は現在Pythonを学んでいない。多くの高校教員はPythonを教えたことがない。

(2) 高等学校新学習指導要領の共通教科【情報】では、必修科目として情報1、選択科目として情報2がある。情報1のコンピュータとプログラミングの領域では、python, JavaScriptの実装例の説明がある。

(3) 今回の問題の出題が Python or 手計算 の 2 者択一の形となったが、最適か。言語が選べた方がフェアだった可能性があるか。

(4) 思考力・創造性を測るのであれば、科学の甲子園第 2 回大会の様な「大会専用のプログラミング言語」でもよかったか。Python の問題であることは 1 カ月程度前に事前資料によって参加生徒の知ることとなった。特定のプログラミング言語に対し、どこかですでに学んでいた生徒がいるチームとそうでないチームに、大会前から差異があると考えられるが、よいか。

○高校生に必要なプログラミングのリテラシーは何か。

(1) やはり数学が大切か。(「ことば」としての「数学」)

(2) 育成すべきリテラシー教育と、共通テストへの出題に対する対応としての教育は、ほぼ同等か。

(3) プログラミング言語は Python ? JavaScript ? R ?

参考) SSH 生徒研究発表会 (2021,8/4,8/5,8/20 ; 各 SSH 校が 1 校 1 本ずつ発表を持ち寄る)

数学・情報分野 2021 年発表本数: 数学 18 本, 情報 15 本 情報はほとんど Python で書いていた。年々情報分野の割合が増えている。

## 参考文献

[1] 科学の甲子園ホームページ

<https://koushien.jst.go.jp/koushien/pre/index.html> (R3 年 11 月 24 日取得)

[2] 文部科学省: 高等学校学習指導要領 (平成 30 年告示) 解説【数学編 理数編】, 2018.