

Bit ArrowとPythonを利用した オンラインでの情報科教員研修

漆原 宏丞¹ 本多 佑希¹ 兼宗 進¹

概要：2022年度から開始される新教育課程では、小学校や中学校に続き、高等学校の情報科でもプログラミングの内容が必修となる。それに伴い、全国の自治体においてもプログラミングを題材とした教員研修が行われている。本発表では、オンラインプログラミング環境「Bit Arrow」を利用して実施した教員研修について報告する。

1. はじめに

2022年度から、高等学校の情報科では必修教科目「情報I」と選択科目「情報II」の両方で、プログラミングの内容が扱われる [1]。それに合わせて、自治体の教員研修でもプログラミングが扱われている [2] [3] [4] [5]。

今回は、情報Iの内容を扱う情報科の教員研修を紹介する。研修では前半に小中中でどのような内容を扱うのかという座学的な内容を扱い、後半では情報Iで扱うプログラミングを体験する演習を行った。本稿では、この後半の演習の内容を中心に報告する。

プログラミング言語は教員研修資料でも扱われているPythonを用いた。1時間40分ほどの演習の中では、前半でPythonの言語仕様を含めた基礎的な内容を扱い、後半ではアルゴリズムやシミュレーション、顔認識などを扱った。

プログラミング環境には、環境構築が不要で、オンラインから手軽に扱えるBit Arrow [6]を採用した。本稿では、研修で扱った内容と事前アンケート、事後アンケートの結果を紹介するとともに、アンケート結果や授業の様子を報告する。

2. 実施した演習

2.1 研修の概要

研修は2021年2月8日にオンラインで実施した。参加者は54名であった。3月にも同じ内容の研修を予定している。

時間は3時間である。時間配分は、最初に「小中学校で行われるプログラミング教育」と「情報Iおよび情報IIの

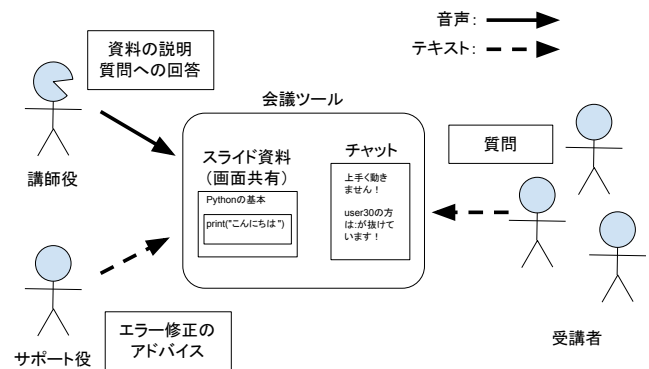


図1 演習の形態

内容」「情報入試」の解説を40分間で行った。その後、プログラミングの演習を90分間行い、最後にまとめを行った。

2.2 演習の体制

今回はオンラインでの研修となったため、質問やそれに対する回答、エラーに対する補助などを工夫した。Bit Arrowは生徒の状況を一覧することができる教員用ページを提供している [7]。そのため、このページを活用しながら、エラーが出ている受講者に対して、どう修正すべきかを助言することにした。

今回は、講師役の名が研修資料の内容を音声で説明しながらチャットの質問に回答し、サポート役の2名がBit Arrowの教員ページを確認しながらエラーの修正方法を各受講者に対してチャットで助言することにした。図1に演習の形態を示す。

2.3 演習の環境

今回の研修で使ったBit ArrowのPythonでは、クライアント側の端末内でPythonのプログラムをJavaScriptに

¹ 大阪電気通信大学
Osaka Electro-Communication University

前半
(1) Bit Arrow の使い方
(2) 文字の出力
(3) 変数 + 数値演算
(4) データ型
(5) 入力 + 型変換
(6) 条件分岐 + 比較演算 + インデント
(7) 論理演算
(8) モジュールインポート + 乱数
(9) 反復
(10) 制御構造の入れ子
(11) リスト
(12) 関数定義
(13) 引数の受け取り
後半
(14) 線形探索
(15) シミュレーション
(16) 画像認識で顔検出 (プログラムは配布)

図 2 演習のカリキュラム

変換して実行する「クライアント実行」と、プログラムをサーバ側で実行する「サーバ実行」が存在する。今回は、細かい説明を省くため基本的にはサーバ実行を用いた。今回の内容では、表 2 の (15)、および (16) はサーバ実行でしか動作しないが、それ以前は全てクライアント実行で動作する。実際に授業で扱う際には、多くの生徒が同時に大量に通信を行うとネットワークに負荷がかかる可能性があるため、クライアント実行を上手く使いつつ、必要なときだけサーバ実行を使うというモデルが望ましい。

2.4 前半の内容

演習で扱ったカリキュラムを図 2 に示す。前半は「Python の基本」を扱い、後半は「その内容を使った応用」という構成にした。

前半では、基本的に情報 I の学習指導要領に沿った題材とした。プログラミングが初めての受講者がいることを想定して、言語仕様や制御構造などを初歩から扱うことにした。ただし、演習で使える時間は限られていたことから、演習では数行単位のプログラムを、スモールステップを意識しつつ細かい単位で扱った。

前半で扱ったプログラム例を図 3、図 4、図 5 に示す。

2.5 後半の内容

後半ではプログラムの細かい説明は省き、そのプログラムで何をしたいのか、どういう流れで目的を達成しているのかということに重きを置きつつ演習を行った。後半の最後には、プログラミングの可能性を感じてもらうために顔認識のプログラムを扱った。プログラムの複雑さを考慮して、プログラムを配布し、パラメータを変更してもらう程度にとどめた。

```
1 x = 5
2 if x <= 5 :
3     print("5以下")
```

図 3 前半のプログラム例 (条件分岐)

```
1 for i in range(1, 10):
2     if i % 2 == 0:
3         print(i)
```

図 4 前半のプログラム例 (制御構造の入れ子)

```
1 def p( m, n ) :
2     print( m * n )
3
4 p( 3, 2 )
```

図 5 前半のプログラム例 (関数)

図 6 に演習内で扱った線形探索のプログラムを示す。使われているプログラムの概念は前半で扱ったもののみであるため、プログラムの内容で躓くことはなく、リストに対してどのような操作を行うのかというアルゴリズムの本質に集中できると考えた。

また、図 7 に演習内で扱ったサイコロのシミュレーションのプログラムを示す。1 から 6 までの値を乱数で生成することで、サイコロを振るという現実の事象をシミュレートする。そして、100 回サイコロを振った際の各出目の出現回数を確認する。こちらも使われているプログラムの概念は前半で扱ったもののみである。そのため、N が増えることによって結果にどのように影響するのかを予想しシミュレートするなど、シミュレーションで学ぶべき部分に集中できると考えた。図 8 に、図 7 の実行結果を示す。

図 9 で、演習内で扱った、カスケード分類器を用いた顔検出のプログラムを示す。情報 I の内容からは外れているため、紹介程度にとどめた。この例はプログラムも長く、新たな概念が多数登場するため、受講者にプログラミングしてもらおうのではなく、Bit Arrow の機能を使ってプログラムを配布することにした。受講者にはプログラムを実行してもらおうとともに、パラメータを変更し、人物の顔をより検出できるように調整してもらった。図 10 に、図 9 の実行結果を示す。

3. 事前アンケート

3.1 概要

研修を行う前に、受講者に事前アンケートに答えてもらった。事前アンケートの設問の一覧を表 1 に示す。

3.2 事前アンケートの設問 1 の回答

図 11 に、事前アンケートの設問 1 の回答を示す。殆どの受講者が普通科 (社会と情報) を担当しており、普通科 (情報の科学) を担当している。兼任で情報の科学を担当

```

1 import random
2
3 ary = []
4 for i in range(10):
5     ary.append(random.randint(1, 10))
6
7 print("ary", ary)
8 target = 5
9
10 for i in range(10):
11     if target == ary[i]:
12         print(i)
    
```

図 6 後半で扱ったプログラム例 (線形探索)

```

1 import random
2 import matplotlib.pyplot as plt
3
4 deme = []
5 N = 100
6 for i in range(N):
7     r = random.randint(1, 6)
8     deme.append(r)
9
10 for i in range(1, 7):
11     c = deme.count(i)
12     plt.bar(i, c)
13 plt.show()
    
```

図 7 後半で扱ったプログラム例 (シミュレーション)

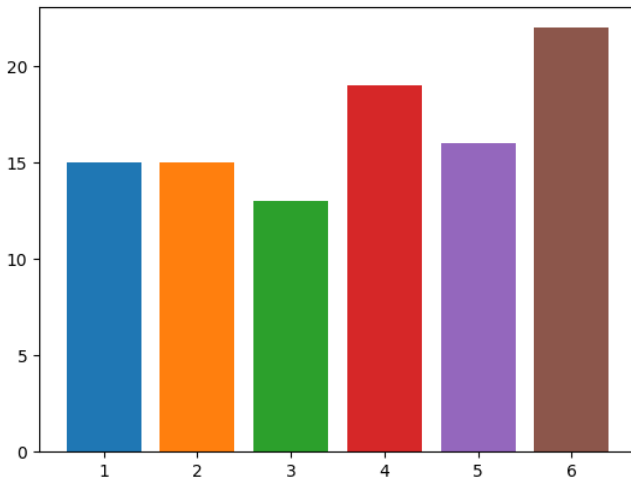


図 8 図 7 の実行結果

している受講者も複数見受けられた。

3.3 事前アンケートの設問 2 の回答

図 12 に、事前アンケートの設問 2 の回答を示す。プログラミング言語とは異なるが、HTML を設問の回答の中に加えた。プログラミング言語を扱ったことがない受講者も 1 名見受けられたが、概ね全ての受講者が何かしらの言語を経験したことがあることが分かった。

```

1 import cv2
2
3 img = cv2.imread("class/picture.jpg")
4 gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
5
6 cascade = cv2.CascadeClassifier("class/
7     haarcascade_frontalface_default.xml")
8
9 faces = cascade.detectMultiScale(gray,
10     scaleFactor=1.3, minNeighbors=2)
11
12 for x, y, w, h in faces:
13     img = cv2.rectangle(img, (x, y), (x + w, y + h),
14         (150, 0, 255), 2)
15
16 cv2.imshow("", img)
    
```

図 9 後半で扱ったプログラム例 (顔検出)



図 10 図 9 の実行結果

表 1 事前アンケートの設問

設問番号	内容
1	所属と担当科目を教えてください
2	勉強や仕事で、どのようなプログラミング言語を使ったことがありますか？
3	授業で、どのようなプログラミング言語を教えたことがありますか？
4	授業でプログラミングを教えることについての気持ち
5	プログラミング授業について、今までの経験や、今後の授業への期待や不安、研修で学びたいことなど

所属と担当科目を教えてください
54 件の回答

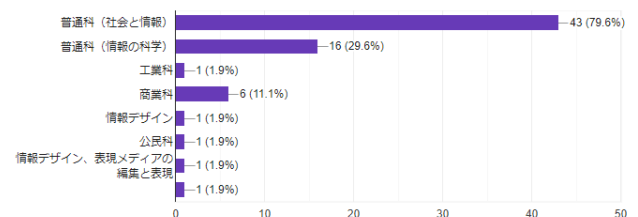


図 11 設問 1 の結果 (事前)

勉強や仕事で、どのようなプログラミング言語を使ったことがありますか？

54件の回答

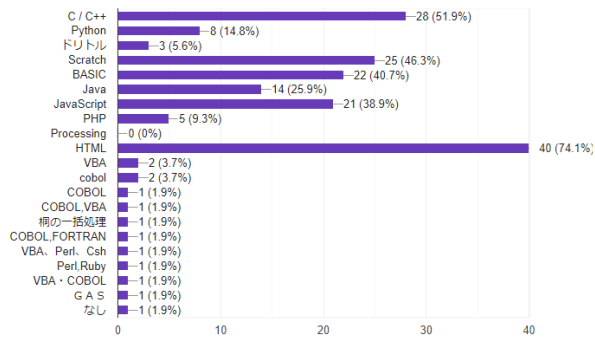


図 12 設問 2 の結果 (事前)

授業で、どのようなプログラミング言語を教えたことがありますか？

54件の回答

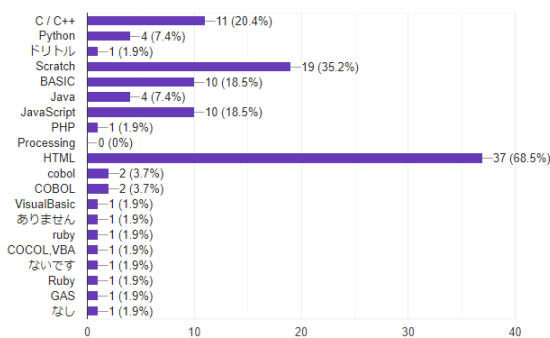


図 13 設問 3 の結果 (事前)

授業でプログラミングを教えることについての気持ち

54件の回答

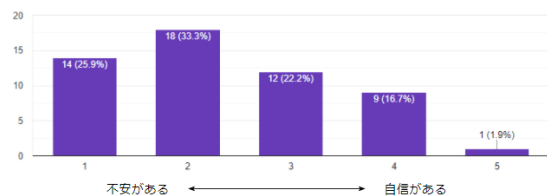


図 14 設問 4 の結果 (事前)

3.4 事前アンケートの設問 3 の回答

図 13 に、事前アンケートの設問 3 の回答を示す。プログラミング言語とは異なる HTML を省くと、Scratch や C/C++、BASIC や JavaScript などを授業で経験したことがあるとわかった。一方、Python は 7.4%と低く、今回の研修は大きな意味を持つことが分かった。

3.5 事前アンケートの設問 4 の回答

図 14 に、事前アンケートの設問 4 の回答を示す。この結果から、過半数の受講者はプログラミングを授業で扱うことに関して不安があると感じていることが分かった。

3.6 事前アンケートの設問 5 の回答

図 15 に、事前アンケートの設問 5 の回答の中から、特徴

的な回答を示す。エラーの対処ができるかどうか不安であったり、時間数に関しての不安が多く見られた。また、どこまで入試を意識しながら授業を組み立てたら良いかが気になるという回答や、プログラミング言語の移り変わりが激しいことから、今教えた概念が入試や大学で通用するのかを気にする回答もあった。

4. 演習の様子

4.1 概要

演習で用いた Bit Arrow では、授業ごとに生徒を登録し、管理する機能が提供されている。この機能を使って、受講生がどれだけエラーを起こしたかなどを確認した。

4.2 エラーと実行回数の割合

事前アンケートの結果から、プログラミングを一度も経験したことがない受講生は少ないことがわかっていたため、スペースや記号が全角になっていることや、文法的なエラーがあまり多くはないことが予想された。

演習中の全受講者の実行回数は合計 2481 回、エラーの回数は 463 回だった。各受講者の実行回数の平均は約 45 回で、1 時間 40 分の演習時間を考えると積極的に参加していたと思われる。各受講者のエラー率の平均は約 18.3%であった。補助として TA に参加した補助役の主観でも、あまりエラーも多くはなかったと思われる。

4.3 演習の完成度

各受講者がどれだけ演習を完成させることができたかを集計した。途中、離席したと思われる受講者や、後半だけ実際にプログラミングし、前半は見るだけで終わった受講者も数名見られた。エラーのまま終わっていたプログラムは全受講者の全プログラムのうち 4 件のみで、殆どの受講者がプログラムを完成させられていることがわかった。また、最後のシミュレーションの課題は、ファイルが無い受講者が 2 名、編集途中で終わっているという受講者が 1 名いた。最後の演習の入力が間に合わない、もしくは諦めてしまったのだと考えられる。しかし、54 名中 3 名と考えると多くはない。そのため、難易度としては丁度良く設定できたのではないかと考えている。

5. 事後アンケート

5.1 概要

研修を受けた後に、受講者に事後アンケートに答えてもらった。事後アンケートの設問の一覧を表 3 に示す。

5.2 事後アンケートの設問 1 の回答

図 15 に、事後アンケートの設問 1 の回答を示す。51% が 4、15.7% が 5、合計 66.7%と、多くの受講生が授業を行うイメージをある程度は掴めたと回答した。平均値は約

表 2 設問 5 の特徴的な回答 (事前)

番号	回答内容
1	言語について本格的に勉強したことがないので、不安
2	詳しくないので、エラーが起こった時に対処する方法がわからない時が不安。
3	1人授業なので、サポートがない中で置いて行かれてる生徒の扱いについて
4	どこまでどのように教えれば良いのか俯瞰したい
5	入試のために座学中心で、点数を取れる授業をしていった方が良いのか？
6	たんなる写経にならない授業にしたい
7	1対40人でコードミスを含めて教えられるのか不安
8	プログラミングの授業において、指導方法や教材など
9	授業で高度なプログラミング、AI 関係を紹介、指導をしていきたいが自分の知識がなく、不安。
10	授業時間数が足りない
11	2単位という時間的制約からプログラミングは勿論内容を全て網羅するのが大変すぎる。これで、本当に共通テストが実施されたら2単位ではとてもやりきれない。
12	時代とともに変化していく言語選択に疑問を感じています。高校1年生で教えた言語が入試である2年後の段階で廃れてしまっているかもしれないし、大学卒業時には全く使われていないということも予想されます。

表 3 事後アンケートの設問

設問番号	内容
1	今回の研修で授業のイメージを掴めましたか？
2	研修の内容は難しかったですか？
3	研修の内容は理解できましたか？
4	Bit Arrow を使ってみて、 操作などはわかりやすかったですか？
5	Bit Arrow を授業で活用してみたいですか？
6	理解できなかった点があれば教えてください
7	研修で学べたこと、 印象に残ったことがあれば教えてください

今回の研修で授業のイメージを掴めましたか？
 51件の回答

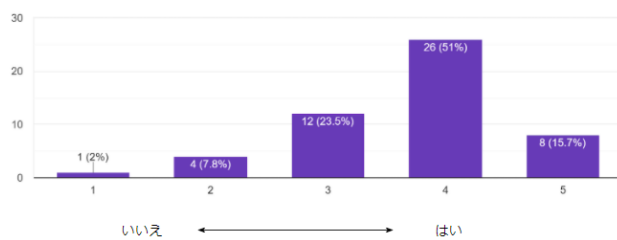


図 15 設問 1 の結果 (事後)

3.71 であるため、高い割合でイメージが掴めていることがわかった。短い時間に多くの概念を学ぶことになったが、情報 I で学ぶ内容は可能な限り多くを扱ったため、どのようなことを学ぶ必要があるのかという点ではイメージできたのだと考えている。

5.3 事後アンケートの設問 2 の回答

図 16 に、事後アンケートの設問 2 の回答を示す。23.5% が 1、31.4% が 2、合計 54.9% と、多くの受講者が難しくはなかったと回答した。平均値は約 2.49 であるため、難しすぎず、簡単すぎもしない丁度良い難易度だったと考えられる。この結果は、各受講者の演習の完成具合からも妥当だ

研修の内容は難しかったですか
 51件の回答

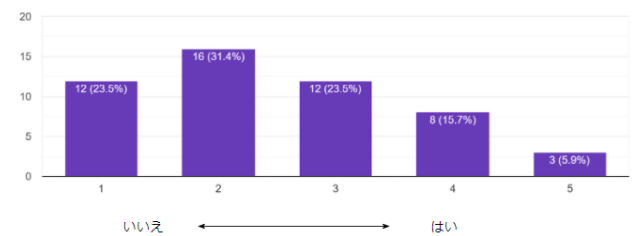


図 16 設問 2 の結果 (事後)

研修の内容は理解できましたか
 51件の回答

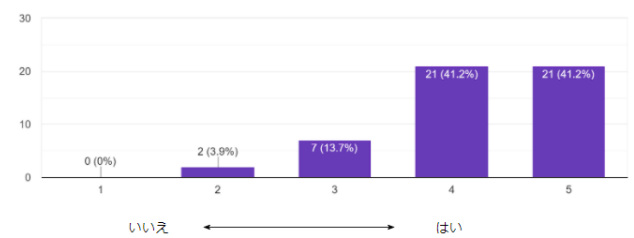


図 17 設問 3 の結果 (事後)

ろうと考えている。

5.4 事後アンケートの設問 3 の回答

図 17 に、事後アンケートの設問 3 の回答を示す。4 と 5 が共に 41.2% であり、平均値も約 4.20 であるため、高い割合で研修の内容は理解できたと回答した。前半は各概念ごとに短いプログラムをスモールステップで扱ったため、学びに躓きが少なかったのだと思われる。

5.5 事後アンケートの設問 4 の回答

図 18 に、事後アンケートの設問 4 の回答を示す。平均値は 4.16 であり、概ね Bit Arrow の操作がわかりやすかったと回答した。操作がわかりやすいことは演習の円滑さに影響するため、演習での躓きが少なく、理解度が高かった

Bit Arrowを使ってみて、操作などは分かりやすかったですか？
 51件の回答

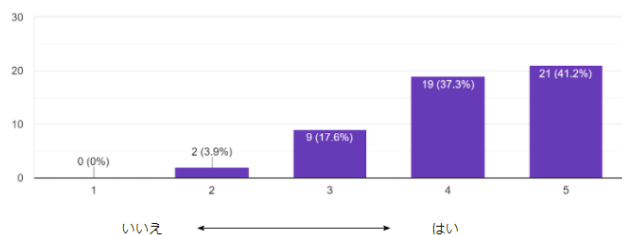


図 18 設問 4 の結果 (事後)

Bit Arrowを授業で活用してみたいですか？
 51件の回答

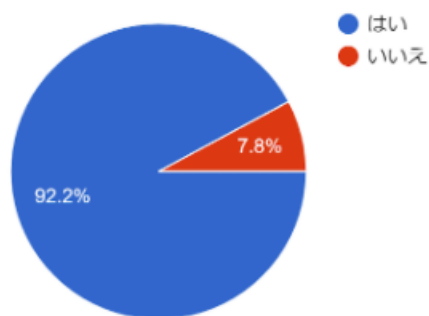


図 19 設問 5 の結果 (事後)

結果にも寄与していると考えられる。

5.6 事後アンケートの設問 5 の回答

研修の理解度とは直接関係はないが、Bit Arrow を授業で活用したいかを回答してもらった。図 19 に、事後アンケートの設問 5 の回答を示す。92.2%の受講者が Bit Arrow を授業で活用したいと回答した。操作がわかりやすかったという点から、実際に授業でも活用したいと答えたのだと考えられる。

5.7 事後アンケートの設問 6 の回答

表 4 に、事後アンケートの設問 6 の回答を示す。Bit Arrow の使い方をもっと知りたかったという回答が散見された。また、エラーの修正が難しく、エラー表示をわかりやすくしてほしいという回答も散見された。生徒が自分でエラーの修正ができると教員は授業に集中できるようになるため、エラーについて重要視する声が多かったのだと考えられる。

5.8 事後アンケートの設問 7 の回答

表 5 に、事後アンケートの設問 7 の回答を示す。ブラウザで書けることや生徒ごとのプログラムのエラー履歴などが見れるという点で Bit Arrow が良かったという回答が多く見られた。また、3 番の回答のように、画像認識に関する回答も散見された。オンライン研修だったため、オンラ

イン授業のときの参考にしたいという意見や、チャットでの質問がしやすかったという意見もあった。講師役が口頭が答えた質問もあったが、チャットで補助役が答えた質問に関しては、回答を遡って確認できるのが良かったという回答も見られた。

6. 考察

今回の研修で良かった点の中で、特徴的に感じたものを挙げる。

- (1) 基礎の概念を数行単位のスモールステップで学べたこと
- (2) 質問や返答がチャットで行えたこと
- (3) Bit Arrow が優れていたこと

(1) の点は、演習の途中から内容についてこれなかった受講者がほぼいなかったこと、難易度を問うアンケートで丁度良い難易度だったという回答をもとに考察した。少ない演習時間の中で内容を多く入れるための工夫だったが、これにより、それぞれの概念を効率的に学ぶことができたと考えている。

(2) の点は、自由記述のアンケートで、質問や返答を遡って確認できるのが良かったという意見をもとに考察した。対面での研修であれば、全ての説明が終わってから質問をする、解説を遮って質問をするしかない。解説を聞きながらリアルタイムで質問ができたのは、オンライン研修会がかつチャットで質問を受け付ける形態にしたためである。また、質問への返答を講師役が行いながら、補助役もチャットで返答したことも良かったと考えられる。これにより、返答もチャットに残るため、遡ってもう一度確認するということがやりやすい。

(3) の点は、自由記述のアンケートや、Bit Arrow の操作がわかりやすかったかを問うアンケート、授業で Bit Arrow を活用したいかを問うアンケートの結果から一目瞭然である。ファイル作成やファイルの編集、実行などの基本操作が難しいほど、演習を進める効率が落ちてしまう。円滑に進められた要因の大きな一つとして、Bit Arrow が優れていたことが挙げられる。また、学生の実行ログをリアルタイムで確認できる機能も授業の円滑な進行に貢献した。補助役が実行ログを確認しながら、エラーの原因をリアルタイムでチャットで指摘できることも大きな利点だったと考えられる。

7. おわりに

教員研修で Bit Arrow を使った Python の研修を行った。情報 I の内容に沿った研修を行い、実行ログや、実施した事前アンケート・事後アンケートをもとに結果を分析して紹介した。Bit Arrow を使ったことや、概念を細かく扱い、演習を短いプログラム単位で扱ったことから演習が円滑に行えたものと思われる。

表 4 設問 6 の特徴的な回答 (事後)

番号	回答内容
1	Bit Arrow の授業での活用方法をもっと詳しく知りたかったです！
2	時間をかけて覚えればもう少し分かるようになるのではないかと思いますので、この後、復習します。
3	エラーの返しがわかりづらかったです。生徒が理解できるようなエラーメッセージだとありがたいです。
4	文字が全部合っているのにエラーが出てきた。 if や for のインデントが入力されていないことによるエラーだったので、インデントを視覚化できるようにしてほしい
5	線形探索から先はプログラムを受け取れる形にしてほしかったです。

表 5 設問 7 の特徴的な回答 (事後)

番号	回答内容
1	ブラウザで簡単にコードを書けるのが分かり、プログラミングに対するハードルが低くなったと思います。
2	生徒の実習履歴やエラーリストの確認ができるところがよさそう。
3	最後の画像認識プログラムは、python の真骨頂だと思います。認識の正確性を調整していましたが、与えられたプログラムであってもそこから、もっと使いやすくするなどのプログラムを修正・バージョンアップしていく作業を研修の中でもっと時間を割いてできると良いと感じました。
4	生徒の学習状況が見えるところに興味を惹かれました
5	オンラインの研修でしたが、質問もしやすく、また回答もログが残っていてわかりやすかったです。非常に勉強になりました。
6	生徒側でやった事で、生徒が陥りやすいミスを一通り体験できた。
7	実際に自分が生徒として授業を受けてみて、始まる前は難しいんだろうなと思っていましたが、実際に触ってみて出来た時には嬉しく思えたり、一つ一つ順番に行っていくことで分かりやすかったです。またすぐにエラーの個所をチャットで教えてもらうことができたのも良かったです。自分が授業をするときにも授業中に全員のエラーを見つけることは難しいと思いますが、エラー個所を一緒に見つけていくことで生徒の苦手意識を減らせると思いました。

参考文献

- [1] 文部科学省: 新学習指導要領 (平成 29 年), https://www.mext.go.jp/a_menu/shotou/new-cs/1383986.htm (2021/2/17 参照).
- [2] 文部科学省: 高等学校情報科「情報 I」教員研修用教材, https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1416756.htm (2021/2/17 参照).
- [3] 文部科学省: 高等学校情報科「情報 II」教員研修用教材, https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/mext_00742.html (2021/2/17 参照).
- [4] 沼崎拓也: 新学習指導要領で求められる情報科教員の育成・採用・研修, 情報処理学会, 情報処理, Vol.59, No.11, pp.1030-1033, 2018.
- [5] 鹿野利春: 情報科教員を目指すにあたって 新学習指導要領実施に向けて準備すべきこと, 情報処理, Vol.62, No.2, pp.84-87, 2021.
- [6] 長島和平, 長慎也, 間辺広樹, 兼宗進, 並木美太郎: Web ブラウザを用いたプログラミング学習支援環境 Bit Arrow の設計と評価, 情報処理学会論文誌「教育とコンピュータ」, Vol.4, No.1, pp.57-69 (2018).
- [7] 長島 和平, 堀越 将之, 長 慎也, 間辺 広樹, 兼宗 進, 並木 美太郎: プログラミング学習支援環境 Bit Arrow の教員支援機能の設計と試作, 情報処理学会, 情報教育シンポジウム (SSS2017), Vol.2017, No.18, pp.129-136 (2017).