

B2-6

本研究の内容は
前任校で実施した
ものです

プログラミング教育における 言語統一型と言語選択型での学習効果の比較

◆ 井 手 広 康 ◆

愛知県立旭丘高等学校 教諭

k619154u@gmail.com



どのようなプログラミング教育を実施していますか？

プログラミング教育を通してどのような生徒を育てたいですか？



主体的・対話的で深い学び



目の前の生徒は、**主体的に**プログラミングを学んでいますか？



目の前の生徒は、**対話的に**プログラミングを学んでいますか？

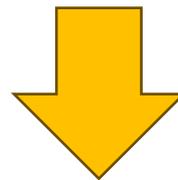


目の前の生徒は、プログラミングが**深い学び**になっていますか？



個別最適な学びと**協働的な学び**の一体的な充実を通して、

「主体的・対話的で深い学び」を実現する



「自立した学習者」へ



中央教育審議会：「令和の日本型学校教育」の構築を目指して ～全ての子どもたちの可能性を引き出す、個別最適な学びと、協働的な学びの実現～（答申）（令和3年1月26日）
文部科学省初等中等教育局教育課程課：学習指導要領の趣旨の実現に向けた個別最適な学びと協働的な学びの一体的な充実に関する参考資料（令和3年3月版）



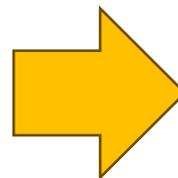


目の前の生徒は、**主体的に**プログラミングを学んでいますか？



【言語統一型】

全体でプログラミング言語を**統一**



【言語選択型】

各生徒がプログラミング言語を**選択**

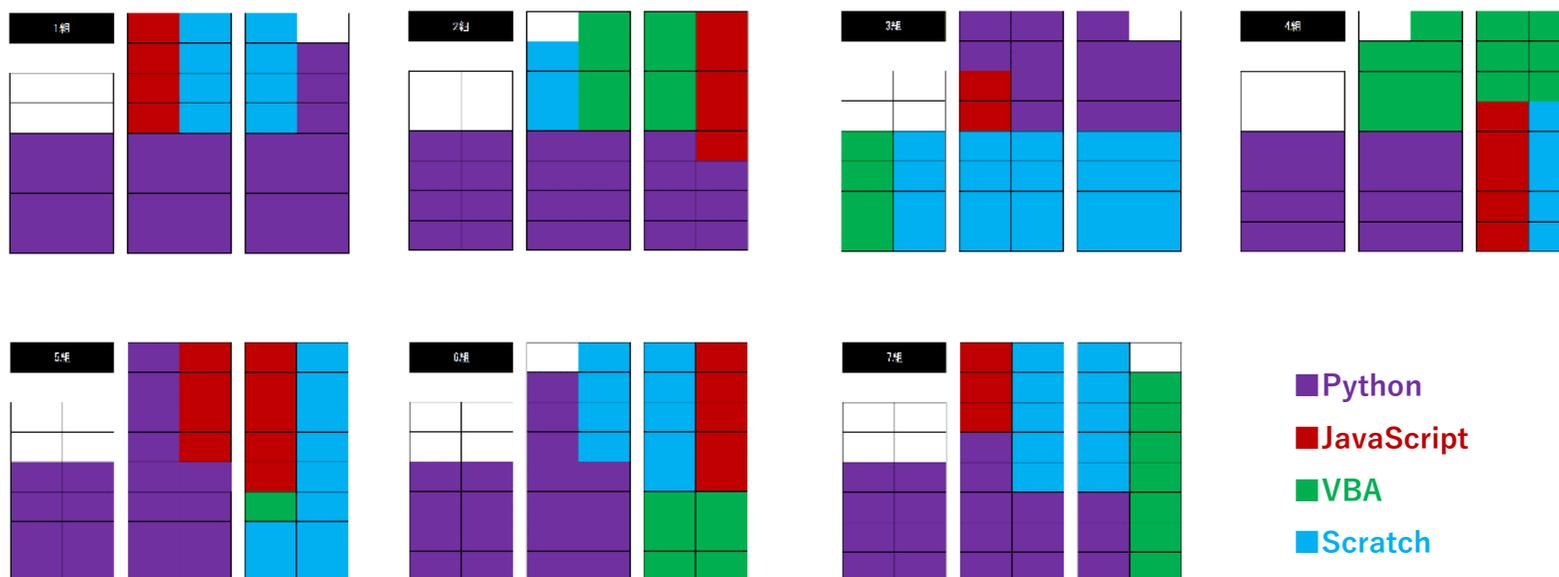
個別最適な学び

自身の**興味関心・進路希望・将来の夢**から言語を選択させる





目の前の生徒は、**対話的に**プログラミングを学んでいますか？



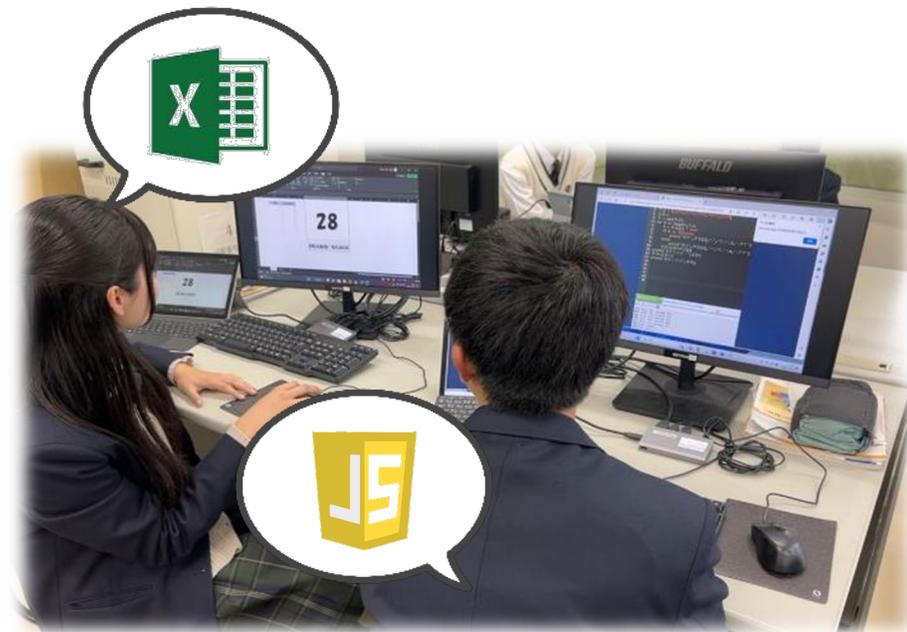
協働的な学び

会話自由，席の移動自由，**互いに教え合いながら**課題に取り組む





目の前の生徒は、プログラミングが**深い学び**になっていますか？



自立した学習者

プログラム制作・発表会を通して、**プログラミングの楽しさを共有**する





どのようにして生徒は言語を選択したのか？



プログラミング言語を選択するまでの過程



プログラミングの授業について **10分**



言語の簡単な説明 **10分**

Python 難易度：★★☆

入門から機械学習まで大人気のプログラミング言語

```

tokyo = ["浅草", "浅草", "浅草", "浅草"]
tokyo = ["1305,695,1440,180"]
season = 0
month = 0
for m in range(4):
    season = season + Tokyo[m]
    kizuna = season / 1000
    print("季節調整:", kizuna)
    print("北風調整")
for m in range(4):
    print(Tokyo[m], ":", Tokyo[m] / kizuna)
    
```

▼使用される分野

- データサイエンス
- 機械学習
- Webアプリ
- デスクトップアプリ

(a) Pythonで書かれたプログラム

JavaScript 難易度：★★★

Webブラウザ/サーバで活躍するプログラミング言語

```

var Tokyo = ["浅草", "浅草", "浅草", "浅草"];
var Tokyo = [1305, 695, 1440, 180];
var season = 0;
var month = 0;
for (let m = 0; m < 4; m++) {
    season = season + Tokyo[m];
}
var kizuna = season / 1000;
alert("季節調整:" + kizuna);
alert("北風調整");
for (let m = 0; m < 4; m++) {
    alert(Tokyo[m] + ":" + Tokyo[m] / kizuna);
}
    
```

▼使用される分野

- Webデザイン
- Webアプリ・コンテンツ
- Webサーバ

(c) JavaScriptで書かれたプログラム

VBA Visual Basic for Applications 難易度：★★★

仕事を強力にサポートするOfficeアプリ専用の言語

```

Sub マシナリ調整()
Dim Tokyo As Variant, Tokyo As Variant
Dim season As Integer, month As Integer
Dim kizuna As Double, Tokyo As Variant
Tokyo = Array(1305, 695, 1440, 180)
season = 0
For m = 0 To 3
    season = season + Tokyo(m)
Next m
kizuna = season / 1000
MsgBox "季節調整:" & kizuna
MsgBox "北風調整"
For m = 0 To 3
    MsgBox Tokyo(m) & ":" & Tokyo(m) / kizuna
Next m
End Sub
    
```

▼使用される分野

- 事務処理の自動化
- 簡単な業務システム
- ExcelやWord等のOfficeアプリ

(d) VBAで書かれたプログラム

Scratch 難易度：★☆☆

ブロックをつなぎ合わせてプログラムを作成する教育用言語

▼使用される分野

- プログラミングの入門
- 教育現場での活用（主に小学校）
- ゲーム制作

(f) Scratchで書かれたプログラム



自由時間 **20分**

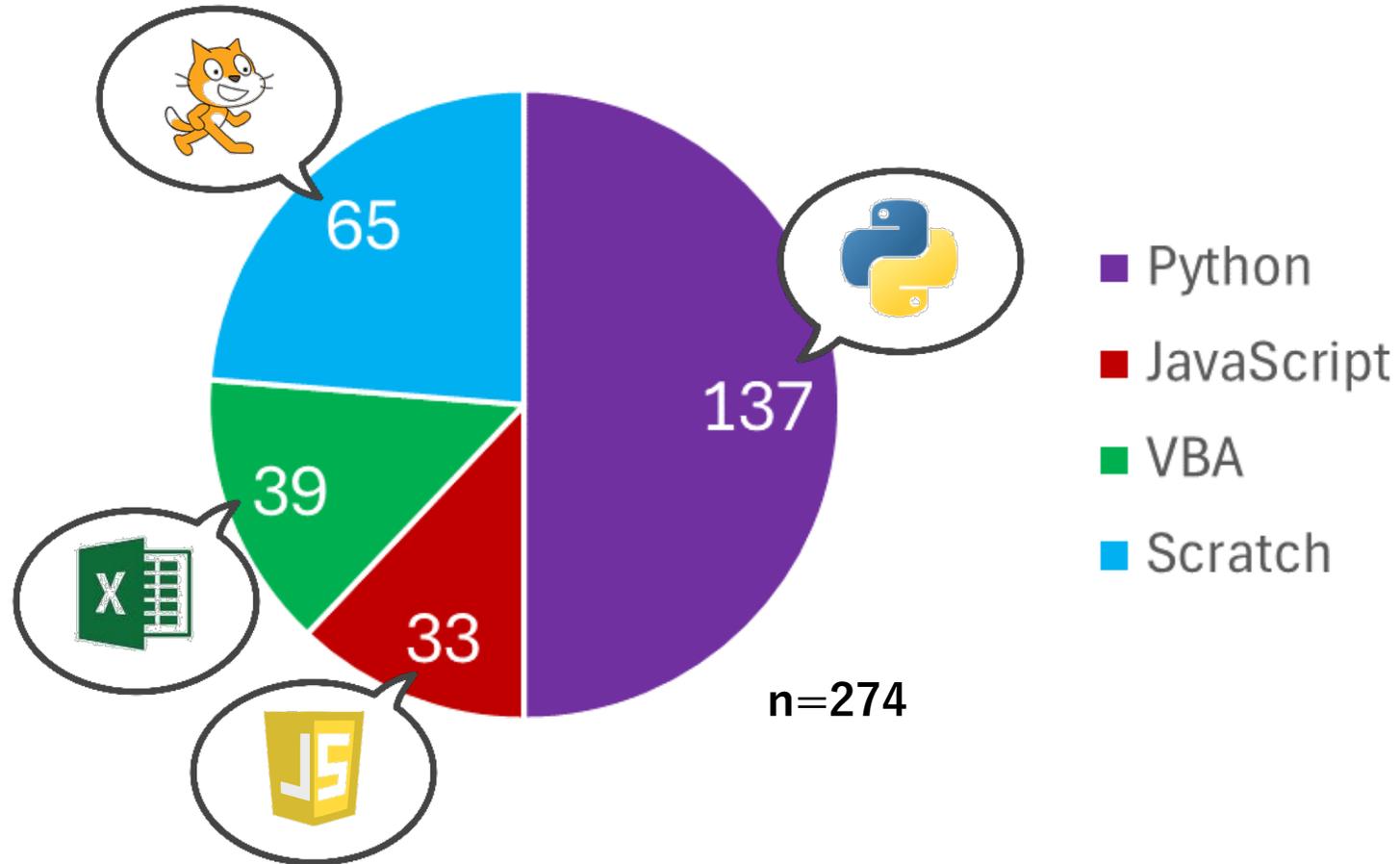


言語の回答（アンケート含む） **10分**

※後日の回答の修正も可



生徒が**選択**したプログラミング言語の内訳

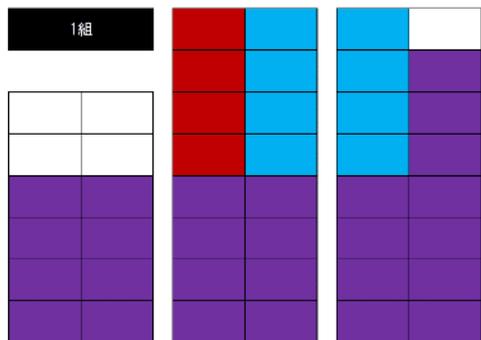


組	Python		JavaScript		VBA		Scratch	
	(A)	(B)	(A)	(B)	(A)	(B)	(A)	(B)
1	40	27	—	4	—	0	—	8
2	40	23	—	5	—	8	—	3
3	—	13	40	2	—	4	—	20
4	—	16	40	5	—	13	—	5
5	—	20	—	9	39	1	—	10
6	—	19	—	5	40	6	—	9
7	—	19	—	3	—	7	40	10
計	80	137	30	33	79	39	40	65

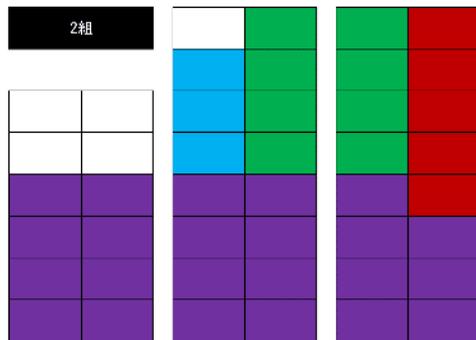
※実践(A)…言語統一型（2021），実践(B)…言語選択型（2023）



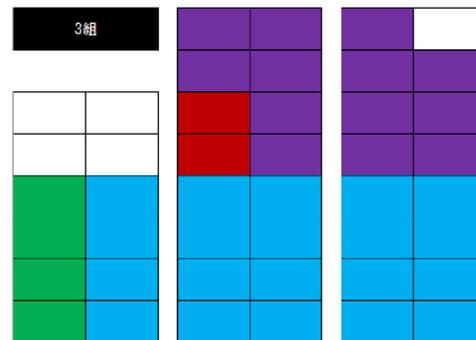
クラス別の授業の**座席表**



座席表：1組



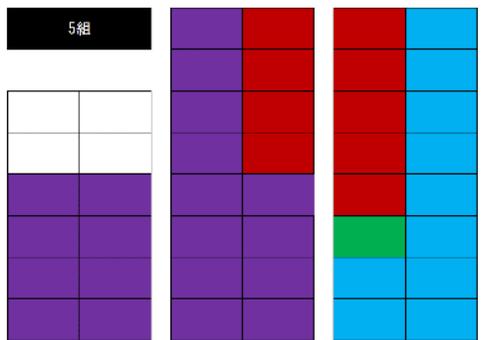
座席表：2組



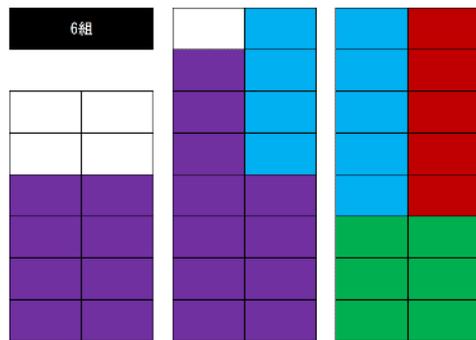
座席表：3組



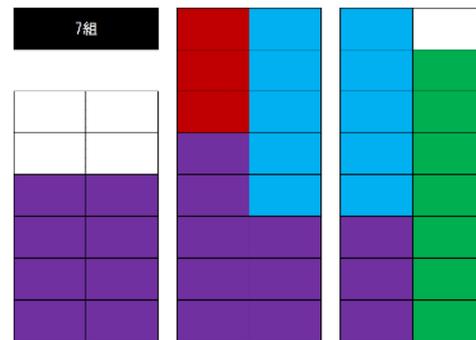
座席表：4組



座席表：5組



座席表：6組



座席表：7組

- Python (137名)
- JavaScript (33名)
- VBA (39名)
- Scratch (65名)





どのようにして授業を実施したのか？



基本的な授業の流れ

Step **一** 萬

全体説明

(約**10**分)

条件分岐
条件分岐は、条件式によって処理を変えるアルゴリズム(流れ)のこと。

Python	共通テスト用プログラム表記
<pre> tensu = 70 if tensu > 60: hantei = "OK" else: hantei = "NG" print(hantei) </pre>	<pre> tensu = 70 もし tensu > 60 ならば hantei = "OK" そうでなければ hantei = "NG" 表示する(hantei) </pre>

例：条件によってOKかNGを表示するプログラム

Step **二** 萬

例題演習

(約**20**分)

例題
次のプログラムを実行し、結果を表示させなさい。

	Python	共通テスト用プログラム表記	備考
問3	<pre> ten = 90 if ten >= 80: print("Great") elif ten >= 60: print("Good") else: print("Not Good") </pre>	<pre> ten = 90 もし ten >= 80 ならば 表示する("Great") そうでなければ もし ten >= 60 ならば 表示する("Good") そうでなければ 表示する("Not Good") </pre>	5つの分岐 >=, >, <, <=, >=

※ elif を利用して複数の条件式を処理できる(いくつでも追加が可能)

Step **三** 萬

発展課題

(約**20**分)

発展
5つ以上の分岐があるプログラムを作成しなさい(条件式は何でも構いません)。また、変数をさまざまな値に変更して、それぞれの分岐が処理が正しいか、プログラムを実行して確認しなさい。



愛知大会参加者限定
28,000円 → **0円** !!



ご自由に加工して
お使いください！

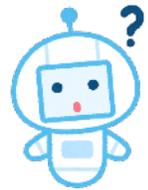
<https://drive.google.com/drive/folders/1hOeC6sS4UR2bQPdS6ukulgbAxR9cuB5u?usp=sharing>

約 **7** スライド × **4** 言語 × **10** 回分 = 約 **280** スライド



授業回	言語統一型（2021）	言語選択型（2023）
1	メッセージ・変数	メッセージ
2	条件分岐	変数 変更点①
3	繰り返し（for文）	条件分岐
4	繰り返し（while文）	繰り返し（for文）
5	配列／リスト	繰り返し（while文）
6	関数	配列／リスト①
7	総合演習①：数当てゲーム	配列／リスト② 変更点②
8	総合演習②：サンプル問題	関数
9	—	総合演習①：プログラム制作
10	—	総合演習②：発表会 変更点③

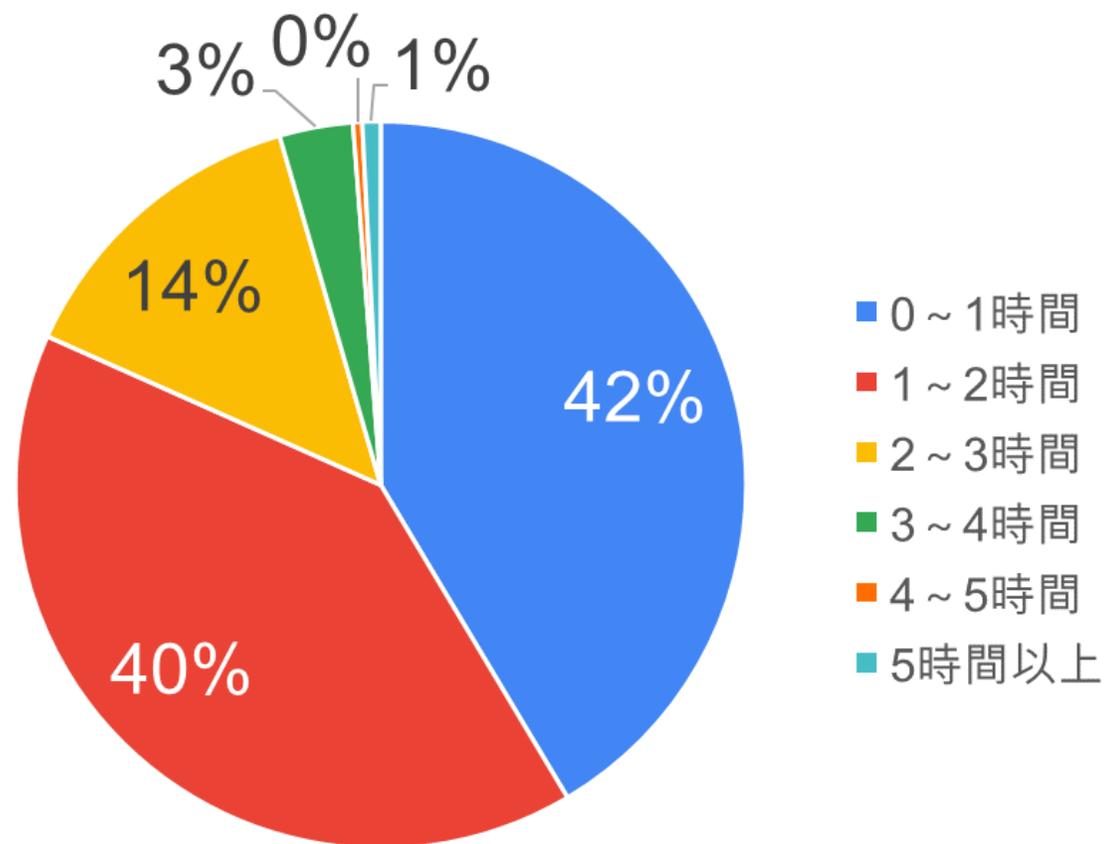




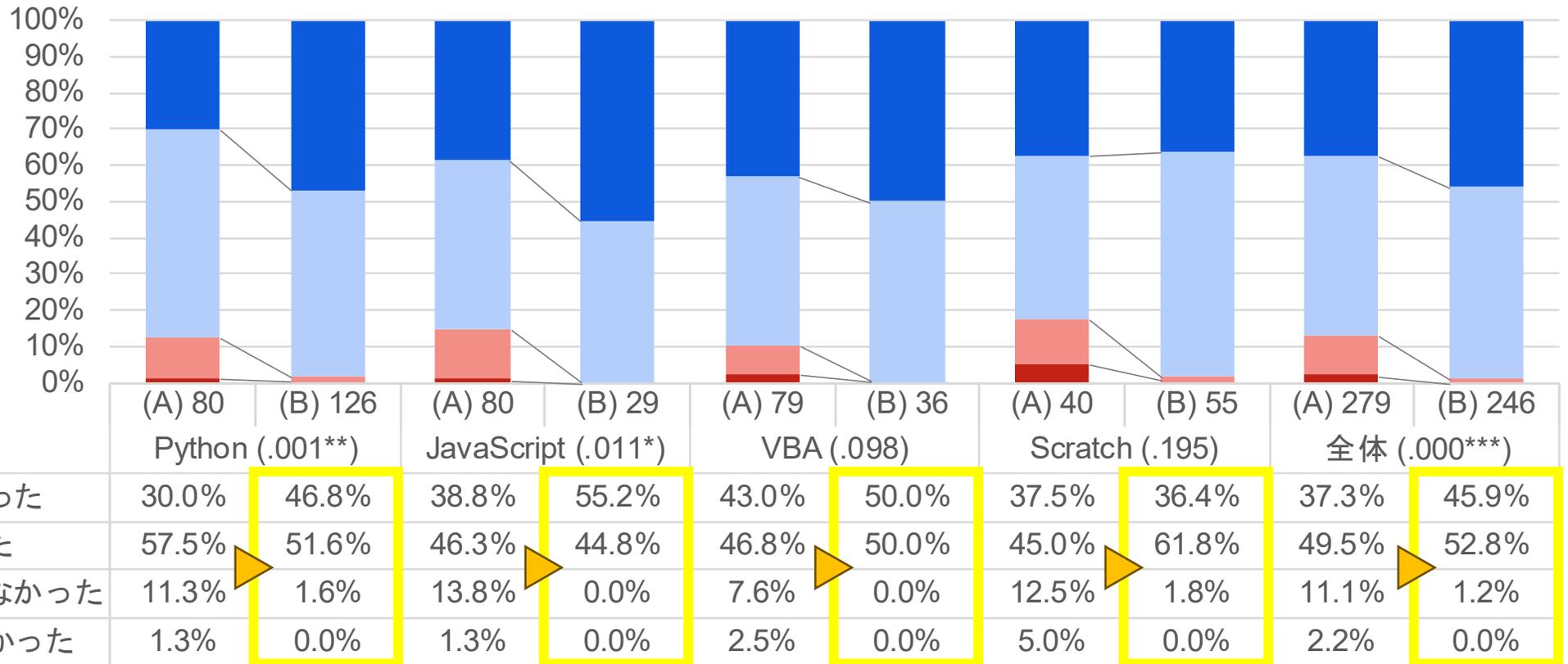
生徒の様子はどうかだったのか？



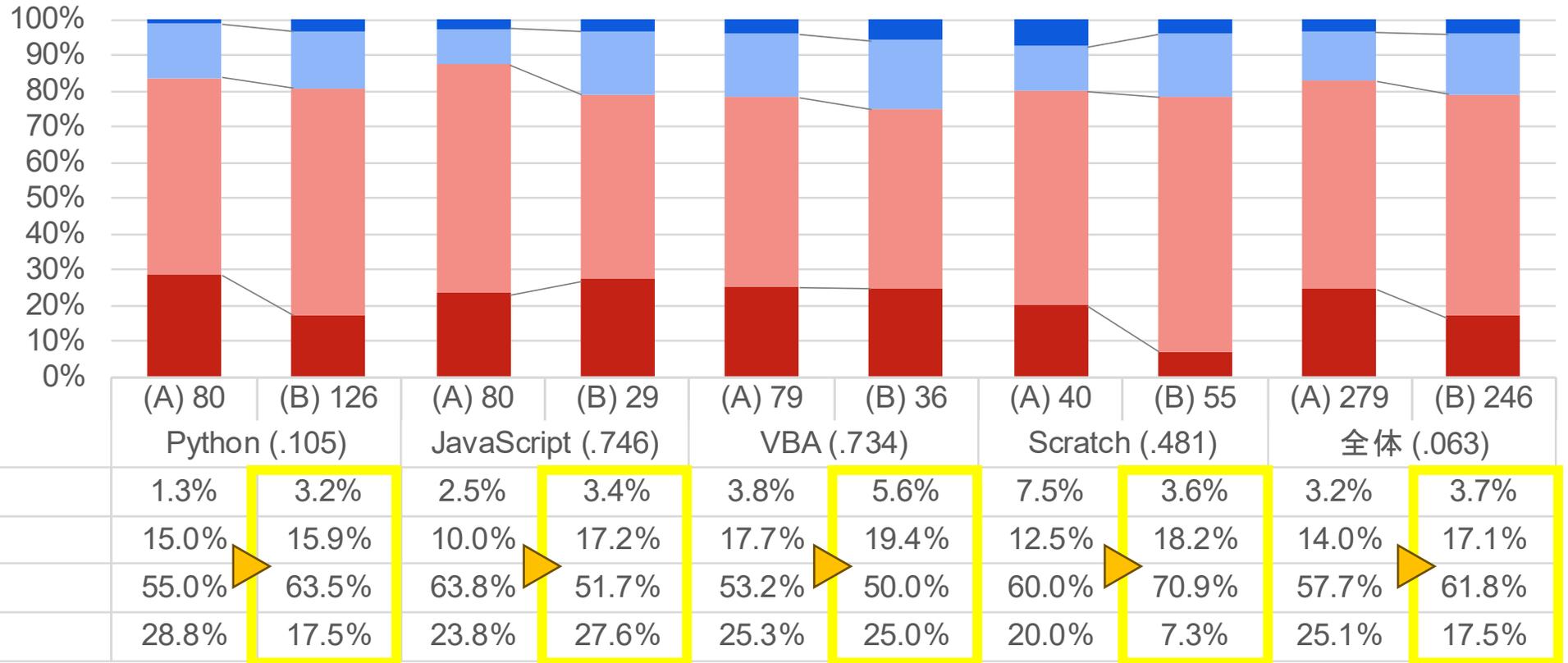
プログラム制作にかけた時間（n=246）



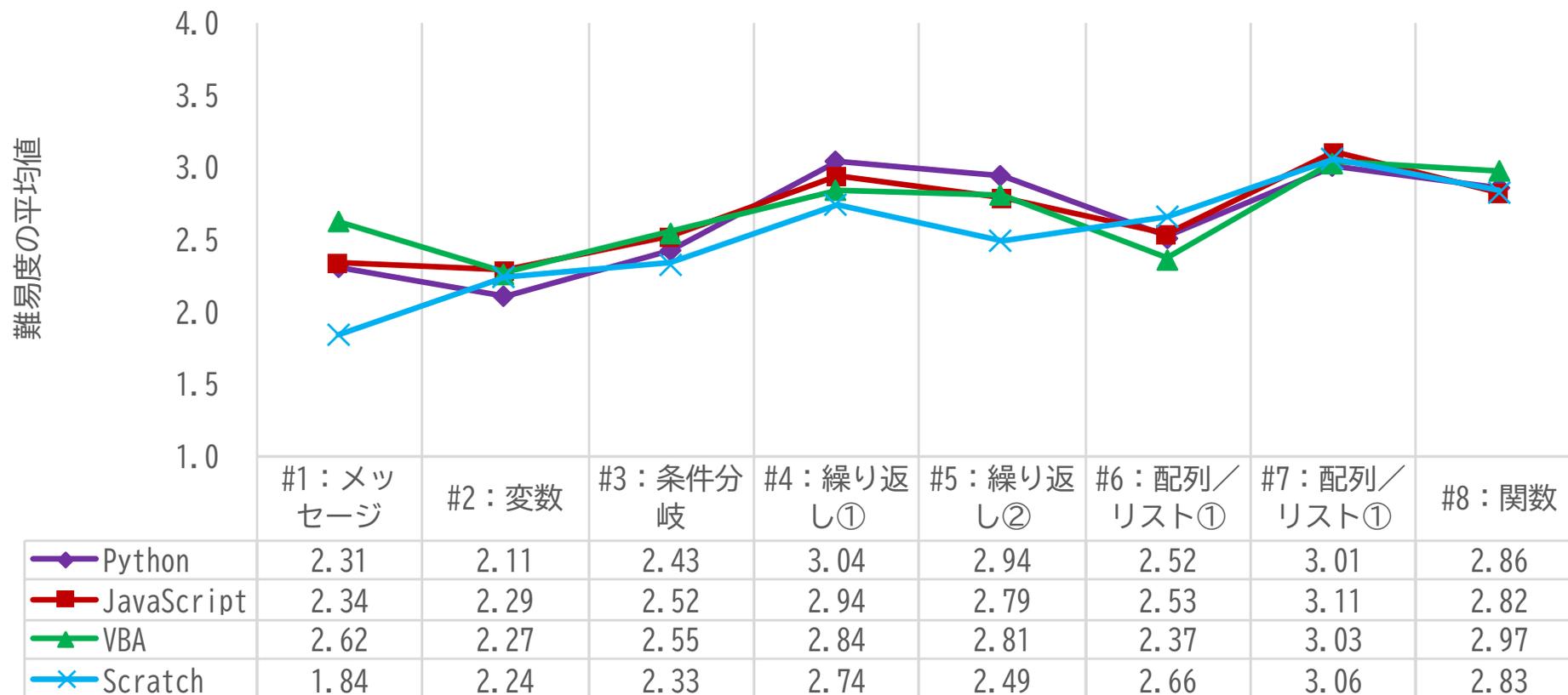
プログラミングの「楽しさ」に関するアンケート結果



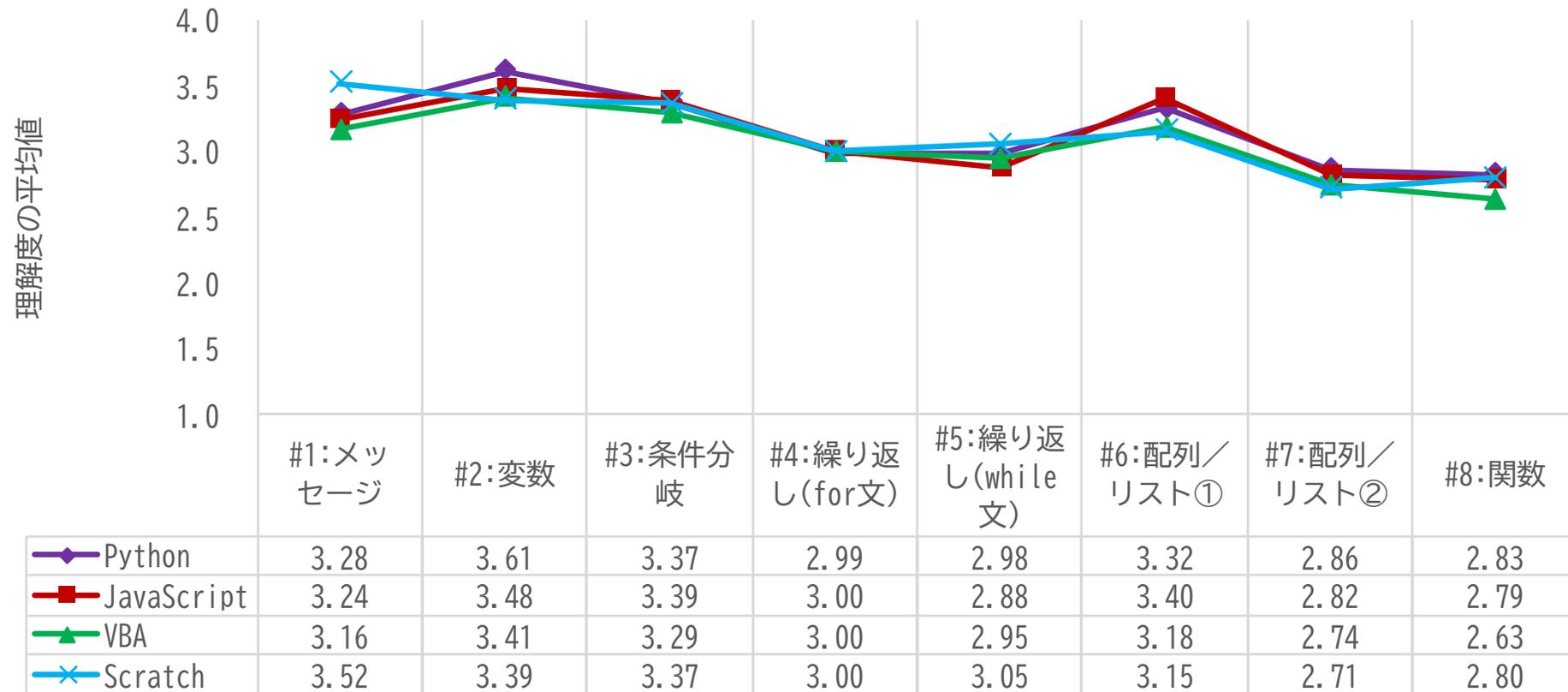
プログラミングの「難易度」に関するアンケート結果



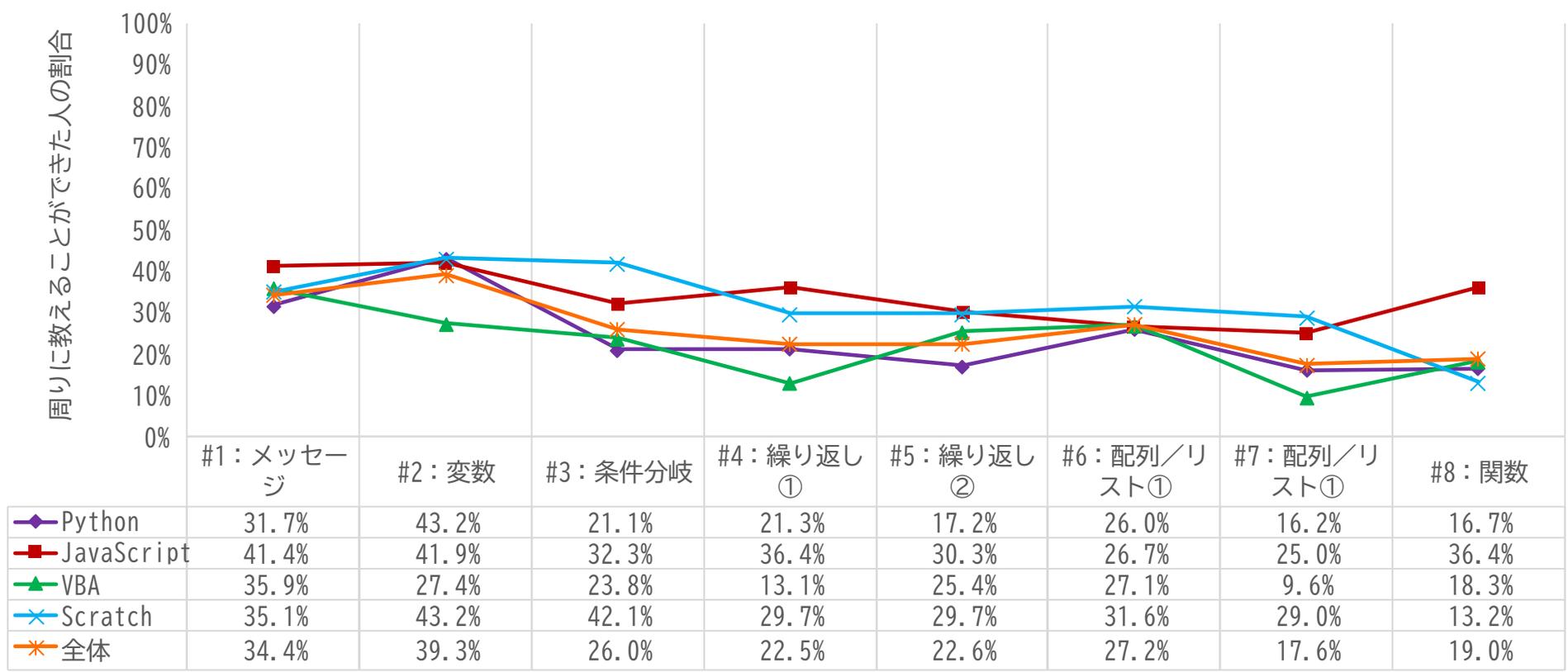
プログラミングの「難易度」に関するアンケート結果



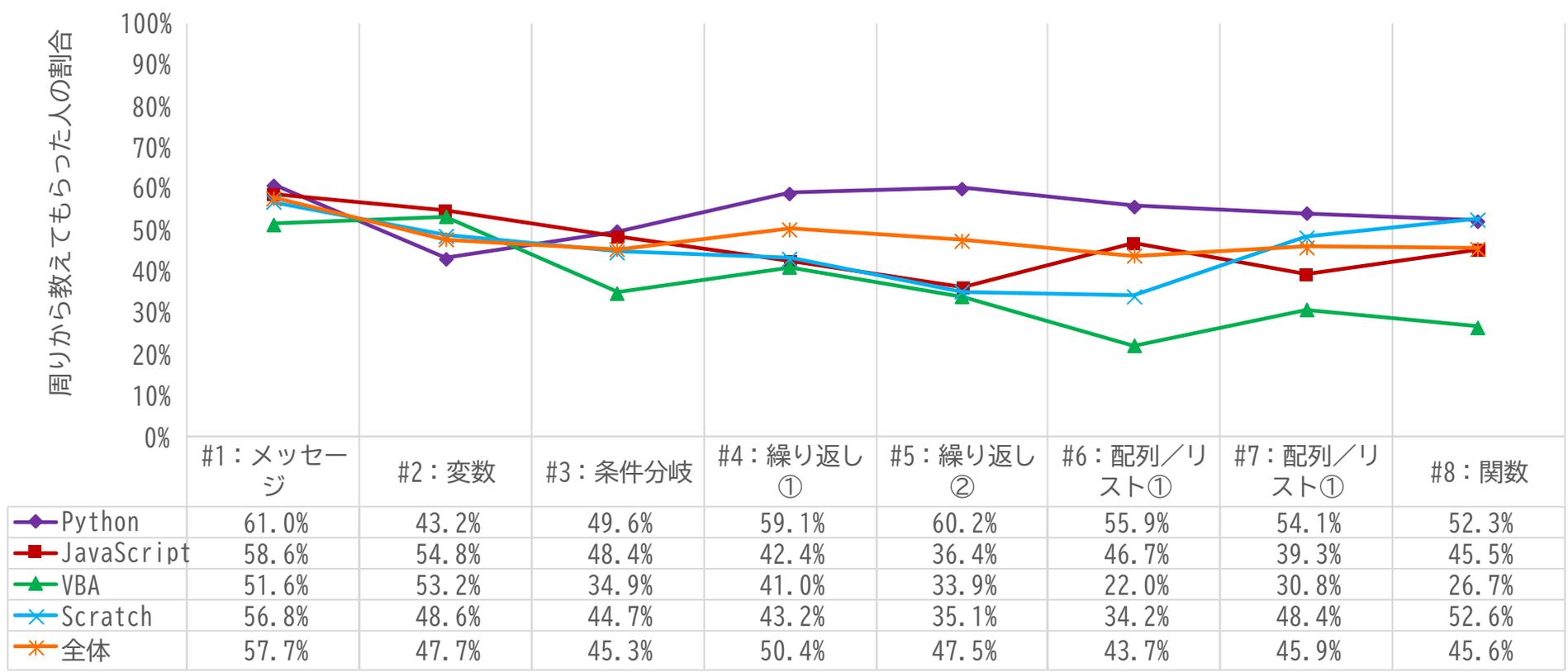
プログラミングの「理解度」に関するアンケート結果



周りの生徒に「教えることができた人」の割合



周りの生徒から「教えてもらった人」の割合



授業後の生徒の感想例

-  隣の席の子と話しながら教えあいながらできました。楽しく、時に難しく多少苦戦しましたが、自分で考えて作ることができました。なぜこうなるかと疑問になりながら、違う数字を入れて理解を深めれたと思いました。普段ふれることができないから楽しかったです。
-  プログラミングと聞いてすごい難しそうなイメージがあったけれど、共通テストまでの範囲なら意外とわかりやすかったし楽しくできたので良かったです。また、わかりやすいものも多かったのでプログラミングに対する意識やイメージが変わりました。
-  後半に行くにつれて「なんでエラーなの!？」「どこが間違ってるんだ…？」と疑問に感じるところが増えてきたけど、家や学校で自分の入力したプログラミングを一つ一つ確認して、「ここがこうなったから結果が表示されなかったのか！」とゆっくりですが確認することができました。最後のプログラミングではいくつかのプログラミング要素を組み合わせたプログラムを組むことができたので良かったです！

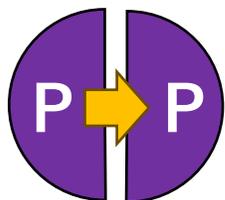


もしもう一度プログラミングをやるとしたらどの言語がいいか

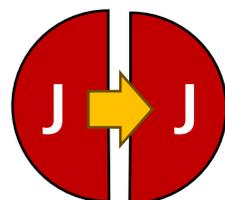
		東	南	西	北
次回		Python 130名	JavaScript 34名	VBA 30名	Scratch 52名
今回					
東	Python 126名	73.8%	7.9%	4.8%	13.5%
南	JavaScript 29名	34.5%	62.1%	0.0%	3.4%
西	VBA 36名	13.9%	11.1%	66.7%	8.3%
北	Scratch 55名	40.0%	3.6%	0.0%	56.4%



同じ言語を選んだ人の理由



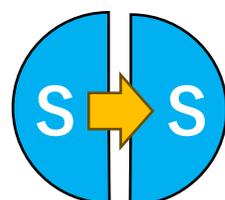
- 友達と教え合いながらやっとなぜかできるようになって、これからはもっと面白いプログラムを今度は一から自分で考えられるようになりたいから。



- 内容はとても難しかったが、最後に結果が出た時の達成感がすごかったので、自分は自分の選んだ言語がよかったと思いました。



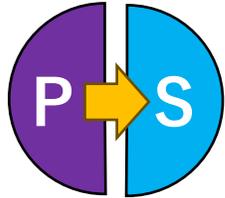
- ちゃんとプログラムをしているという感覚にもなるし、親に聞いたりすると会社でもVBAを使っているらしいので、将来に役に立つかなと思いました。



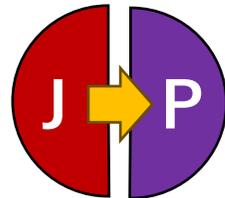
- 最後の発表の時に他の言語のプログラミングも説明を聞いたり、画面を見たけどScratchより難しそうだったから。



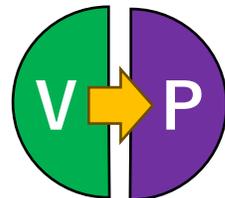
異なる言語を選んだ人の理由（次点の言語）



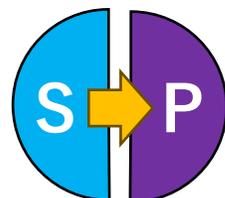
- Scratchはアニメーションや音などがあって楽しそうだなと思ったし、日本語で全部書いてあってわかりやすいなと思ったからです。



- 発表を聞いたのがPythonで、自分が使っていた言語と多少異なるので、やってみたくて思いました。



- VBAは家でできないので少し不便だった。Pythonは選んでいる人も多いので、色々な人に聞きやすいと思ったから。

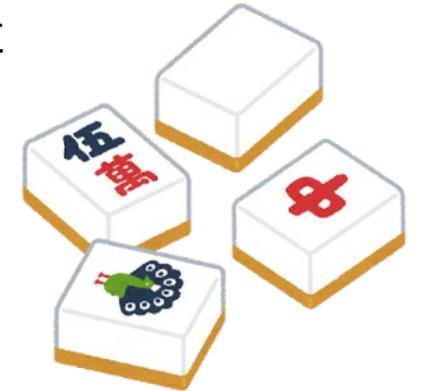
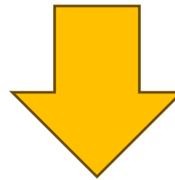


- Pythonが共通テストのプログラミング表記と一番似ていて、授業を通じて共通テスト対策もできるのではと思ったから。



授業実践のまとめ

プログラミング教育に**言語選択**と**発表会**を導入し、
主体的に生徒同士で学び合える環境を最優先した



個別最適な学びと**協働的な学び**の一体的な充実を通して、
「**主体的・対話的で深い学び**」を実現することができた

