

大学入学共通テストを見据えた プログラミング教育の言語選択に関する考察

井手 広康¹

概要：令和4年度から始まる「情報Ⅰ」の教科書にPython, JavaScript, VBA, Scratchの四つのプログラミング言語が使用されたことから、多くの学校がいずれかのプログラミング言語を使用して授業を行うことになる。本研究では、クラスごとに四つのプログラミング言語を使い分けて同じ内容でプログラミング教育を実践し、最後に大学入学共通テスト「情報」サンプル問題の第2問「プログラミング」を全員に解答させた。サンプル問題の解答結果や事後アンケートの結果から、どのプログラミング言語を授業で使用していても、共通テスト「情報」を解答する上で大きな影響がないことが明らかとなった。ただし、Scratchについては他のプログラミング言語と比較して解答に時間を要することが予測されるため、共通テスト「情報」のDNCLに慣れるための演習が必要であると考える。

キーワード：情報Ⅰ, プログラミング教育, プログラミング言語, 大学入学共通テスト, DNCL

Consideration on Language Selection of Programming Education towards the Common Test for University Admissions

HIROYASU IDE^{†1}

Abstract: Since the four programming languages of Python, JavaScript, VBA, and Scratch were used in the textbook of "Information Study I" starting from April 2022, many schools decided to use one of the programming languages for teaching. In this study, we practiced programming education with the same content by using four programming languages for each class, and finally asked everyone to answer the second question "programming" of the sample problem "Information Study" of the Common Test for University Admissions. From the answer results of the sample problem and the results of the post-questionnaire, it became clear that no matter which programming language was used in the class, there was no significant effect on answering the Common Test for University Admissions "Information Study". However, it is expected that it will take more time to answer Scratch compared to other programming languages, so it is necessary to practice to get used to the DNCL of the Common Test for University Admissions "Information Study".

Keywords: Information Study I, Programming Education, Programming Language, The Common Test for University Admissions, DNCL

1. はじめに

令和4年度より高等学校において年次進行で実施される平成30年告示高等学校学習指導要領[1]では、教科「情報」は必修科目「情報Ⅰ」と選択科目「情報Ⅱ」で構成される。このうち必修科目である「情報Ⅰ」は、現行の学習指導要領における「社会と情報」と「情報の科学」が統合された形となり、「情報デザイン」や「プログラミング」、「データの活用」といった単元が必修化された。また、教科「情報」の入試化を巡っては、令和3年3月24日に大学入試センターが公開した「平成30年告示高等学校学習指導要領に対応した令和7年度大学入学共通テストからの出題教科・科目」[2]では、大学入学共通テスト（以下、「共通テスト」と表記）に「情報」を出題する指針が示され、令和3年7月30日に大学入試センターが公開した「令和7年度大学入学共通テスト実施大綱の予告」[3]において、共通テストに「情報」が導入されることが正式

に決定した。さらに、令和3年9月29日に大学入試センターが公開した「令和7年度大学入学共通テストに係る大学入学共通テスト実施大綱の予告（補遺）」[4]では、共通テスト「情報」の試験時間が60分であること、令和3年12月17日に同センターが公開した「令和7年度大学入学共通テストに係る大学入学共通テストにおける得点調整の対象教科・科目について」[5]では、試験科目「情報」の経過措置科目として「旧情報（仮）」が設置され、必要に応じて得点調整がされることが決定した。

このような背景を受け、令和4年度より始まる「情報Ⅰ」では、共通テストを意識した授業を行う必要がある。共通テストに向けて参考となる資料としては、大学入試センターが令和2年11月に関係各所に通達した大学入学共通テスト「情報」試作問題（検討用イメージ）[6]（以下、「試作問題」と表記）、令和3年3月に公開された大学入学共通テスト「情報」サンプル問題[7]（以下、「サンプル問題」と表記）が挙げられる。いずれの問題にも「実際の問題セットをイメージしたものや試験時間を考慮したものではない」という旨の注意書きはあるが、共通テスト「情報」を

¹ 愛知県立小牧高等学校
Aichi Prefectural Komaki High School

第2問 次の文章を読み、後の問い(問1~3)に答えよ。

Mさんは、18歳になって選挙権が得られたのを機に、比例代表選挙の当選者を決定する仕組みに興味を持った。そこで各政党に配分する議席数(当選者数)を決める方法を、友人のKさんとプログラムを用いて検討してみることにした。

問1 次の文章の空欄「ア」~「ウ」に入れる最も適当なものを、後の解答群のうちから一つずつ選べ。同じものを繰り返し選んでもよい。

Mさん: 表1に、最近行われた選挙結果のうち、ある地域のブロックについて、各政党の得票数を書いてみたよ。

	A党	B党	C党	D党
得票数	1200	660	1440	180

Kさん: 今回の議席数は6人だったね。得票の総数を議席数で割ると580人なので、これを基準得票数と呼ぶのがいいかな。平均して1議席が何票分の重みがあるかを表す数ということで。そうすると、各政党の得票数が何議席分に相当するかは、各政党の得票数をこの基準得票数で割れば求められるね。

Mさん: その考え方に沿って政党ごとの当選者数を計算するプログラムを書いてみよう。まず、プログラムの中で扱うデータを図1と図2にまとめてみたよ。配列Tomeiには各政党の党名を、配列Tokuhyoには各政党の得票数を格納することにしたよ。政党の数は4つなので、各配列の添字は0から3だね。

i	0	1	2	3
Tomei	A党	B党	C党	D党

i	0	1	2	3
Tokuhyo	1200	660	1440	180

Mさん: では、これらのデータを使って、各政党の当選者数を求める図3のプログラムを書いてみよう。実行したら図4の結果が表示されたよ。

```

(01) Tomei = ["A党", "B党", "C党", "D党"]
(02) Tokuhyo = [1200, 660, 1440, 180]
(03) sousuu = 0
(04) giseki = 6
(05) mを0から「ア」まで1ずつ増やしなが繰り返す:
(06) sousuu = sousuu + Tokuhyo[m]
(07) kizyunsuu = sousuu / giseki
(08) 表示する("基準得票数:", kizyunsuu)
(09) 表示する("比例配分")
(10) mを0から「ア」まで1ずつ増やしなが繰り返す:
(11) 表示する(Tomei[m], ":", 「イ」 / 「ウ」)
    
```

図3 得票に比例した各政党の当選者数を求めるプログラム

基準得票数: 580
 比例配分
 A党: 2.068966
 B党: 1.137931
 C党: 2.482759
 D党: 0.310345

図4 各政党の当選者数の表示

「ア」~「ウ」の解答群

① 0 ② 1 ③ 2 ④ 3 ⑤ 4 ⑥ 5 ⑦ 6 ⑧ Tomei[m]
 ⑨ Tokuhyo[m] ⑩ sousuu ⑪ giseki ⑫ kizyunsuu

図1 大学入学共通テスト「情報」サンプル問題(第2問-問1)

Figure 1 The Common Test for University Admissions "Information Study" sample problem (Question 2-1)

イメージする上で貴重な資料となっている。試作問題とサンプル問題のプログラミング言語には、Pythonの表記に似た疑似言語が使用されており、サンプル問題とともに公開された「サンプル問題のねらい」[8]には「高等学校の授業で何らかのプログラム言語を用いて実習した生徒であれば容易に理解できるものである」と明記されている。ただし、大学入試センターは「大学入学共通テストで使われるからといって、試験対策のためにDNCLを使って授業が行われることがないようにと考えている」[9]としている。また、実際に授業で扱うプログラミング言語の具体例について文部科学省や大学入試センターは公言していないが、「情報I」の教科書にPython, JavaScript, VBA, Scratchの四つのプログラミング言語が使用されたことから、多くの学校が上記のいずれかのプログラミング言語を使用して授業を行うことになる[10]。

ただし、試作問題とサンプル問題に使用されたDNCLがPythonの表記に似ていることが影響して、「プログラミング言語はPythonを使用した方が共通テストに有利に働くのではないか」という声や、反対に「Scratchはブロック型プログラミング言語であるため、共通テストでは不利にな

るのではないか」という声を聞くことがある。また、特定のプログラミング言語の選択が共通テストの解答に及ぼす影響に関して調査した文献は現在のところ見当たらず、上記のような情報は推測の域を出ない。

そこで本研究では、クラスごとにPython, JavaScript, VBA, Scratchの四つのプログラミング言語を分けて同じ内容でプログラミング教育を実践し、最後にサンプル問題の第2問「プログラミング」を全員に解答させた。以下、本稿では、サンプル問題の解答結果や事後アンケートの結果から、授業で扱うプログラミング言語によってサンプル問題の解答に関して差が生じるのかどうか検証した結果について述べる。

2. 試作問題とサンプル問題の概要

試作問題は全8問から構成され、このうち第5問がプログラミング問題に該当する。第5問はシフト暗号(シーザー暗号)による暗号解読をテーマにした問題であり、頻度分析プログラムを作成して暗号文の解読を行っている。第5問におけるプログラムのまともなものは大きく二つあり、一つは「出現頻度を求めるプログラム」、もう一つは「暗号文

を復号するプログラム」である。問題文は T さん, M さん,

表 1 本実践におけるプログラミングの授業内容

時間	授業内容
1 時間目	変数
2 時間目	条件分岐
3 時間目	繰り返し (for 文)
4 時間目	繰り返し (while 文)
5 時間目	配列/リスト
6 時間目	関数
7 時間目	総合演習
8 時間目	サンプル問題 (第 2 問)

先生の 3 人による会話形式で進められ、プログラムの空欄に入る適切なコードを解答群から選択して解答する形式となっている。

一方、サンプル問題は三つの大問から構成され、第 1 問が「情報社会、情報デザイン、画像処理、ネットワーク」、第 2 問が「プログラミング」、第 3 問が「データの活用」に関する問題となっている。これらの大問のうち、第 2 問「プログラミング」の間 1 を図 1 に示す。第 2 問は、比例代表選挙の当選者を決定する仕組みに関して、各政党に配分する議席数 (当選者数) を決める方法をプログラムする内容となっている。第 2 問におけるプログラムのまとまりは大きく二つあり、一つは図 1 の右上に示した「得票に比例した各政党の当選者を求めるプログラム」、もう一つは「各政党の当選者数を求めるプログラム」である。問題文は M さん, K さん, 先生の 3 人の会話形式で進められ、試作問題と同様に、プログラムの空欄に入る適切なコードを解答群から選択して解答する形式となっている。

試作問題及びサンプル問題では、図 1 のように Python の表記に似た疑似言語が使用されているが、令和 4 年 1 月現在、共通テスト「情報」に使用されるプログラミング言語の仕様書は公開されていない。ただし、試作問題及びサンプル問題に使用されている疑似言語は、「情報関係基礎」に出題されている共通テスト手順記述標準言語 (DNCL) [11] の表記に近い。さらに、サンプル問題とともに公開された「サンプル問題のねらい」には、「問題の中で使用するプログラム言語は、高等学校の授業で多様なプログラム言語が利用される可能性があることから、公平性を鑑みて、大学入試センター独自の日本語表記の疑似言語としている」との記述がある[8]。これらのことから、共通テスト「情報」にも共通テスト用手順記述標準言語 (DNCL) の表記に近いプログラミング言語が使用されると考えられる。そのため本稿では、令和 7 年度からの共通テスト「情報」に使用される言語についても DNCL と表記する。

表 2 プログラミング言語の割り当てと実行環境

クラス	人数	言語	実行環境
1 組	40	Python	Google Colaboratory
2 組	40		
3 組	40	JavaScript	Google Chrome デベロッパーツール
4 組	40		
5 組	39	VBA	Excel VBE
6 組	40		
7 組	40	Scratch	オンライン版 Scratch (Scratch 3.0)

3. 本実践の概要

3.1 プログラミングの授業内容

本実践は、愛知県立小牧高等学校普通科 1 年生 7 クラス 279 名を対象に、令和 3 年度の 2 学期に「社会と情報」において 8 時間 (50 分×8 コマ) かけて実践した。なお「社会と情報」において、生徒らは本実践ではじめてプログラミングを学習することになる。ここで、本実践の 8 時間分の授業内容を表 1 に示す。1 時間目から 6 時間目までの 6 回で、変数、条件分岐、繰り返し、配列/リスト、関数について学習する。次の 7 時間目に、これまでに学習した内容を踏まえた総合演習を行う。最後に 8 時間目において、サンプル問題の第 2 問「プログラミング」を解答し、その後、事後アンケートに回答する。

最後の 8 時間目の授業でサンプル問題 (第 2 問) に解答するため、7 時間目までの授業においてサンプル問題 (第 2 問) に出題されている範囲を学習しておく必要がある。先行研究[12]では、試作問題及びサンプル問題のプログラミング分野に出題された内容を、①変数、②代入、③インクリメント、④メッセージ、⑤異なるデータ型の結合、⑥算術演算子、⑦比較演算子、⑧論理演算子、⑨条件分岐 (if 文)、⑩繰り返し (for 文)、⑪繰り返し (while 文)、⑫入れ子 (ネスト)、⑬配列/リスト、⑭添字 (インデックス)、⑮関数の 15 個に分類している。本実践では、これらすべての項目に関する演習を授業に組み入れるように授業計画を立てた。なお、表 1 の授業内容に記載のない②~⑧、⑫、⑭の項目については、7 時間目までの授業において適宜演習に加えている。

3.2 使用するプログラミング言語

本実践では、表 1 に示した授業内容を、クラスごとに複数のプログラミング言語に分けて実践した。使用したプログラミング言語は「情報 I」の教科書に使用されている Python, JavaScript, VBA, Scratch の四つである。ここで、使用したプログラミング言語の割り当てと、プログラミング言語の実行環境について表 2 に示す。

<pre>import random pc = random.randint(1,10) while True: you = input("数字を入力") you = int(you) if you > pc: print(you,"よりもアよ") elif you < pc: print(you,"よりもイよ") else: print("正解!") break</pre>	<pre>pc = 1 から 10 までの乱数 ずっと繰り返す: you = 入力("数字を入力") you = 数値(you) もし you > pc ならば: 表示する(you,"よりもアよ") もし you < pc ならば: 表示する(you,"よりもイよ") そうでなければ: 表示する("正解!") 繰り返しを抜ける</pre>
---	--

(a) Python の授業プリント (抜粋)

<pre>pc = Math.floor(Math.random() * 10 + 1); while (true){ var you = prompt("数字を入力"); you = Number(you); if (you > pc){ alert(you + "よりもアよ"); }else if (you < pc){ alert(you + "よりもイよ"); }else{ alert("正解!"); break; } }</pre>	<pre>pc = 1 から 10 までの乱数 ずっと繰り返す: you = 入力("数字を入力") you = 数値(you) もし you > pc ならば: 表示する(you,"よりもアよ") もし you < pc ならば: 表示する(you,"よりもイよ") そうでなければ: 表示する("正解!") 繰り返しを抜ける</pre>
---	--

(b) JavaScript の授業プリント (抜粋)

<pre>pc = Int(Rnd * 10 + 1) Do you = InputBox("数字を入力") you = Val(you) If you > pc Then MsgBox you & "よりもアよ" ElseIf you < pc Then MsgBox you & "よりもイよ" Else MsgBox "正解!" Exit Do End If Loop</pre>	<pre>pc = 1 から 10 までの乱数 ずっと繰り返す: you = 入力("数字を入力") you = 数値(you) もし you > pc ならば: 表示する(you,"よりもアよ") もし you < pc ならば: 表示する(you,"よりもイよ") そうでなければ: 表示する("正解!") 繰り返しを抜ける</pre>
---	--

(c) VBA の授業プリント (抜粋)

	<pre>pc = 1 から 10 までの乱数 ずっと繰り返す: you = 入力("数字を入力") もし you > pc ならば: 表示する(you,"よりもアよ") もし you < pc ならば: 表示する(you,"よりもイよ") そうでなければ: 表示する("正解!") 繰り返しを抜ける</pre>
--	--

(d) Scratch の授業プリント (抜粋)

図2 7時間目の授業「総合演習」に使用したプログラミング言語ごとの授業プリント

Figure 2 Lesson prints for each programming language used for the 7th hour lesson "Comprehensive Exercise"

表2に示したように、プログラミング言語は1, 2組がPython, 3, 4組がJavaScript, 5, 6組がVBA, 7組がScratchを使用している。なお、プログラミング言語の実行環境について、VBAとScratchは表2に示したもののしかないが、PythonとJavaScriptについては表2に示したものの以外にもさまざまな実行環境が存在する。先行研究[13]によると、クラスごとに複数の実行環境を分けてPythonによるプログラミングを実践した結果、使用した実行環境によって教育効果に大きな差が生じなかったとしている。そのため本実践では、PythonとJavaScriptに関して環境構築が容易または不要な実行環境をそれぞれ使用している。

3.3 授業プリント

本実践では、表2に示したように四つのプログラミング言語を使用しているため、プログラミング言語ごとに異なる授業プリントを用意している。ただし、プログラミング言語は異なるが、授業プリントの内容は表1に示したものと同じである。ここで、7時間目の授業「総合演習」に使

用した、プログラミング言語ごとの授業プリント(抜粋)を図2に示す。図2のように、授業で扱うプログラミング言語を授業プリントの左側、これをDNCLで表したプログラムを右側に示している。このように、授業で扱うプログラミング言語とDNCLとの対応を常に比較できる状態にしているため、サンプル問題(第2問)を解く8時間目までにDNCLだけの演習は特に行っていない。

4. 結果の考察

4.1 サンプル問題における平均点の比較

サンプル問題(第2問)は三つの問から構成され、問1はア〜ウの3問、問2はエ〜スの10問、問3はセ〜テの6問となっている。サンプル問題に配点は記載されていないが、1問1点の19点満点と設定して合計点を算出した。ただし、大学入試センターが公開している「情報サンプル問題 正解表」[14]には、イ・ウ、タ・チ、ツ・テは「全問正解の場合のみ点を与える」との記載があるため、これら

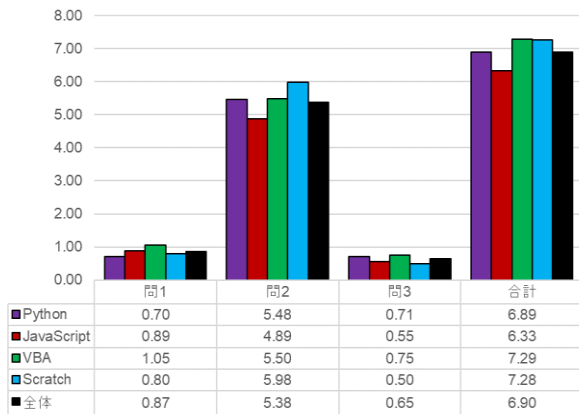


図3 サンプル問題（第2問）における平均点
 Figure 3 Average scores in the sample problem Q-2

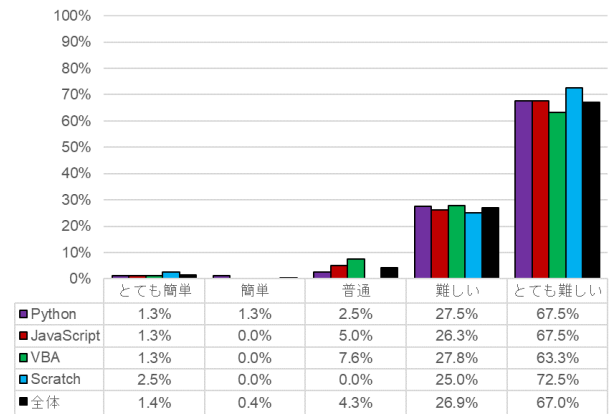


図4 サンプル問題（第2問）における難易度
 Figure 4 Difficulty in the sample problem Q-2

表3 各プログラミング言語のZ検定におけるZ値

Table 3 Z-value in Z-test for each programming language

言語	問1	問2	問3	合計
Python	0.16	0.80	0.41	0.98
JavaScript	0.87	0.18	0.25	0.19
VBA	0.12	0.77	0.22	0.37
Scratch	0.69	0.25	0.21	0.54

は二つが正解の場合のみ2点としている。ここで、プログラミング言語ごとの各問における平均点を図3に示す。なお、サンプル問題の解答者数は279名（有効回答率100%）である。図3の全体の傾向として、平均点が高い順にVBAの7.29点、続いてScratchの7.28点であり、両者にはほとんど差がない結果となっている。さらにScratchに続いてPythonの6.89点、最後にJavaScriptの6.33点となっている。なお、全体の平均点は6.90点であった。

ここで、各プログラミング言語のZ検定におけるZ値を表3に示す。表3のように、いずれのZ値も有意水準5%を下回ることがなく、有意な差は認められなかった。図3を見ると、プログラミング言語ごとに若干の差があることがわかるが、表3のZ検定の結果からは、授業で扱うプログラミング言語によってサンプル問題（第2問）の解答結果に有意な差は生じないといえる。

4.2 サンプル問題における難易度の比較

サンプル問題（第2問）を解答させた後、事後アンケートにおいてサンプル問題の難易度に対して「とても簡単」、「簡単」、「普通」、「難しい」、「とても難しい」の五択で回答させた。この結果を図4に示す。なお、本アンケートの回答者数は279名（有効回答率100%）である。図4の全体の傾向として、割合が大きい順に「とても難しい」の67.0%、「難しい」の26.9%、「普通」の4.3%、「とても簡単」の1.4%、「簡単」の0.4%となっている。

表4 難易度の分布における適合度検定の結果

Table 4 Goodness-of-fit test in difficulty distribution

言語	検定統計量	H ₀	H ₁
Python	2.41 < 9.49	○	×
JavaScript	0.41 < 9.49	○	×
VBA	2.48 < 9.49	○	×
Scratch	2.41 < 9.49	○	×

ここで、全体の割合に対する各プログラミング言語の適合度検定（カイ二乗検定）の結果を表4に示す。なお、この場合の帰無仮説H₀は「全体の難易度の分布と各プログラミング言語における難易度の分布は一致する」、対立仮説H₁は「全体の難易度の分布と各プログラミング言語における難易度の分布は一致しない」となる。また、有意水準5%におけるカイ二乗分布（自由度4）の棄却限界値は9.49であり、表4における「○」は採択、「×」は棄却を表す。表4のように、適合度検定（カイ二乗検定）の結果、いずれのプログラミング言語においても帰無仮説H₀を採択（対立仮説H₁を棄却）していることがわかる。そのため、全体の難易度の分布と各プログラミング言語における難易度の分布は一致し、授業で扱うプログラミング言語によってサンプル問題（第2問）に対して感じる難易度に差は生じないといえる。

4.3 サンプル問題における解答時間の比較

サンプル問題（第2問）を解答させた後、事後アンケートにおいてサンプル問題の解答時間に対して「短い」、「ちょうどよい」、「長い」の三択で回答させた。この結果を図5に示す。なお、サンプル問題（第2問）の解答時間は30分で設定しており、本アンケートの回答者数は279名（有効回答率100%）である。図5の全体の傾向として、割合が大きい順に「ちょうどよい」の67.0%、「長い」の17.6%、「短い」の15.4%となっている。

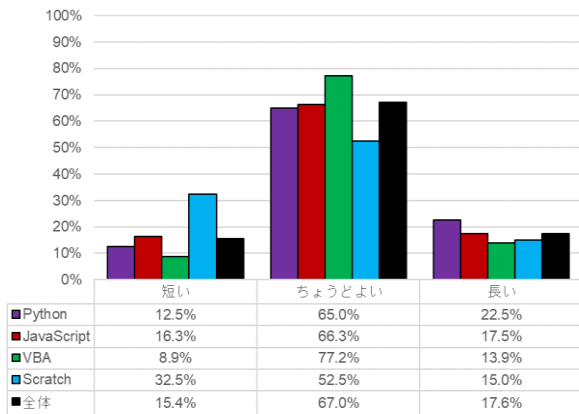


図5 サンプル問題（第2問）における解答時間

Figure 5 Answer time in the sample problem Q-2

表5 解答時間の分布における適合度検定の結果

Table 5 Goodness-of-fit test in distribution of answer time

言語	検定統計量	H ₀	H ₁
Python	1.60 < 5.99	○	×
JavaScript	0.04 < 5.99	○	×
VBA	4.02 < 5.99	○	×
Scratch	8.99 > 5.99	×	○

ここで、全体の割合に対する各プログラミング言語の適合度検定（カイ二乗検定）の結果を表5に示す。なお、この場合の帰無仮説 H₀ は「全体の解答時間の分布と各プログラミング言語における解答時間の分布は一致する」、対立仮説 H₁ は「全体の解答時間の分布と各プログラミング言語における解答時間の分布は一致しない」となる。また、有意水準 5% におけるカイ二乗分布（自由度 2）の棄却限界値は 5.99 であり、表5における「○」は採択、「×」は棄却を表す。表5のように、適合度検定（カイ二乗検定）の結果、Python, JavaScript, VBA が帰無仮説 H₀ を採択（対立仮説 H₁ を棄却）し、Scratch が対立仮説 H₁ を採択（帰無仮説 H₀ を棄却）していることがわかる。そのため、全体の解答時間の分布と各プログラミング言語における解答時間の分布は、Python, JavaScript, VBA のみが一致し、これらのプログラミング言語はサンプル問題（第2問）に対して感じる解答時間に差は生じないといえる。

一方、Scratch は全体の解答時間の分布と比較して、「短い」と回答した割合 32.5% が全体の割合 15.4% と比較して多く、「ちょうどよい」と回答する割合 52.5% が全体の割合 67.0% と比較して少ないといえる。この理由について、図6に示す「授業プリントの左側と右側のプログラムではどちらの方が分かりやすかったか」の質問項目に対する回答（事後アンケートの結果）から考察する。図6のように、JavaScript と VBA は、授業で使用したプログラミング言語よりも DNCL の方が分かりやすかったと回答している割合

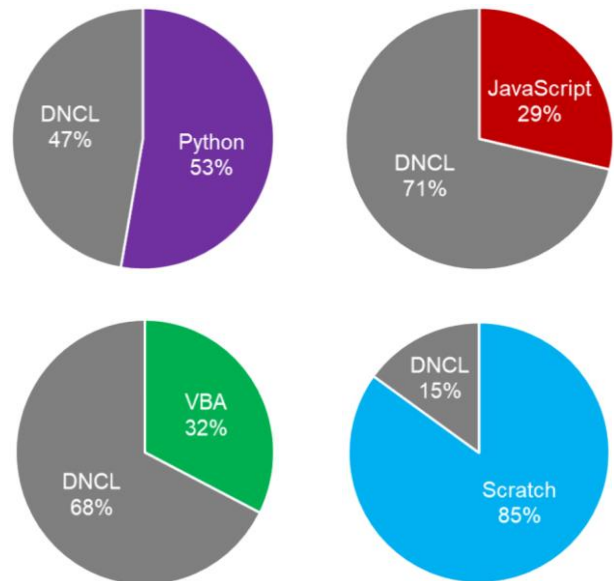


図6 どちらのプログラムが分かりやすかったか

Figure 6 Which program was easier to understand

が多いが、Python と Scratch に関しては、逆に DNCL よりも授業で使用したプログラミング言語の方が分かりやすかったと回答した割合が多い結果となっている。Python については、DNCL が Python の表記に似ていることや、DNCL を含むプログラミング言語の中で Python がもっとも短いコードでプログラムを記載できることが要因であると考えられる。また、Scratch については、ブロック型のプログラミング言語であることが要因して DNCL を読むことに時間がかかり、その結果、解答時間が「短い」と回答した生徒が多くなってしまったのではないかと推測する。

5. おわりに

本研究では、クラスごとに Python, JavaScript, VBA, Scratch の四つのプログラミング言語を分けて同じ内容でプログラミング教育を実践し、最後にサンプル問題の第2問「プログラミング」を全員に解答させた。さらに、サンプル問題の解答結果や事後アンケートから、授業で扱うプログラミング言語によってサンプル問題の解答に関して差が生じるのかどうか検証した。

まず、サンプル問題（第2問）の解答結果から、プログラミング言語ごとに多少の平均点の違いは見られたが、Z 検定からは平均点に有意な差があるとはいえない結果となった。なお、これはサンプル問題（第2問）の合計点だけでなく、問1～問3の各問に対しても同様の結果となっている。この結果から、少なくともサンプル問題（第2問）に関しては、授業で扱うプログラミング言語によって解答結果に有意な差が生じなかったといえる。

次に、サンプル問題（第2問）を解いて感じた難易度を

問うた事後アンケートの結果から、サンプル問題（第2問）が多く生徒にとって「とても難しい」と感じていることがわかった。また、各プログラミング言語で回答結果に多少の違いは見られたが、適合度検定（カイ二乗検定）からは有意な差があるとはいえない結果となった。この結果から、少なくともサンプル問題（第2問）に関しては、授業で扱うプログラミング言語によって問題を解いて感じた難易度に有意な差が生じなかったといえる。

最後に、サンプル問題（第2問）を解いて感じた解答時間を問うた事後アンケートの結果から、サンプル問題（第2問）の解答時間に設定した30分が多く生徒にとって「ちょうどよい」と感じたことがわかった。また、各プログラミング言語で回答結果に多少の違いは見られたが、適合度検定（カイ二乗検定）から Python, JavaScript, VBA については有意な差があるとはいえない結果となった。一方、Scratch については適合度検定（カイ二乗検定）から有意な差がある結果となり、サンプル問題（第2問）の解答時間に対して「短い」と感じた生徒が多く、「ちょうどよい」と感じた生徒が少なかったといえる。この結果については、Scratch がブロック型のプログラミング言語であるため、サンプル問題の DNCL を読むことに時間がかかってしまったことが要因ではないかと推測する。

以上の結果から、「情報 I」の教科書に使用されている Python, JavaScript, VBA, Scratch の四つのプログラミング言語に関して、どのプログラミング言語を授業で使用していても、共通テスト「情報」を解答する上で大きな影響はないものと結論付けることができる。ただし、Scratch については、他のプログラミング言語と比較して解答に時間を要することが予測されるため、共通テスト「情報」の DNCL に慣れるための演習が必要であると考えられる。今後は、令和4年度より始まる「情報 I」において、プログラミング教育の効果的な授業の展開方法や共通テストへの対策について実践及び考察していきたい。

なお、本研究の一部は、日本産業技術教育学会第64回全国大会、第17回情報教育合同研究会、日本情報科教育学会第13回フォーラムにおいて発表している。

謝辞

本研究は、JSPS 科研費 JP21H03962 の助成を受けたものです。

参考文献

- [1] 文部科学省：平成30年告示高等学校学習指導要領，https://www.mext.go.jp/content/1384661_6_1_3.pdf (参照 2022-1-10)。
- [2] 大学入試センター：平成30年告示高等学校学習指導要領に対応した令和7年度大学入学共通テストからの出題教科・科目について，https://www.dnc.ac.jp/kyotsu/shiken_jouhou/r7ikou.html (参照 2022-1-10)。
- [3] 大学入試センター：令和7年度大学入学選抜に係る大学入学共通テスト実施大綱の予告，<https://www.mext.go.jp/content/2022-1-10>。

- [4] 大学入試センター：令和7年度大学入学選抜に係る大学入学共通テスト実施大綱の予告(補遺)，https://www.mext.go.jp/content/20210929-mxt_daigakuc02-000005144_1.pdf (参照 2022-1-10)。
- [5] 大学入試センター：令和7年度大学入学選抜に係る大学入学共通テストにおける得点調整の対象教科・科目について，https://www.dnc.ac.jp/kyotsu/shiken_jouhou/r7ikou.html (参照 2022-1-10)。
- [6] 大学入試センター：平成30年告示高等学校学習指導要領に対応した大学入学共通テストの『情報』の試作問題(検討用イメージ)，https://www.ipsj.or.jp/education/9faeag0000012a50-att/sank_o2.pdf (参照 2022-1-10)。
- [7] 大学入試センター：サンプル問題『情報』問題，https://www.dnc.ac.jp/kyotsu/shiken_jouhou/r7ikou.html (参照 2022-1-10)。
- [8] 大学入試センター：サンプル問題『情報』ねらい，https://www.dnc.ac.jp/kyotsu/shiken_jouhou/r7ikou.html (参照 2022-1-10)。
- [9] 水野修治：大学入学共通テスト新科目案「情報」についてーこれまでの経緯と試作問題(検討イメージ)のねらいー，情報処理学会コンピュータと教育研究会第158回研究発表会招待講演，<https://www.wakuwaku-catch.net/kouen210303/> (参照 2022-1-10)。
- [10] 井手広康：「情報 I」教科書の動向，情報処理学会高校教科「情報」シンポジウム2021秋，<https://www.wakuwaku-catch.net/kouen211101/02/> (参照 2022-1-10)。
- [11] 大学入試センター：共通テスト手順記述標準言語(DNCL)の説明，<https://www.dnc.ac.jp/albums/abm00040701.pdf> (参照 2022-1-10)。
- [12] 井手広康：大学入学共通テスト「情報」サンプル問題を踏まえた情報 I の教科書におけるプログラミング分野の比較，情報処理学会情報教育シンポジウム2021 論文集，9-1，pp.246-253 (2021)。
- [13] 井手広康：高等学校におけるプログラミング環境の違いによって生じる教育効果に関する比較，日本産業技術教育学会論文誌，Vol.63，No.2，pp.269-278 (2021)。
- [14] 大学入試センター：サンプル問題『情報』正解表，https://www.dnc.ac.jp/kyotsu/shiken_jouhou/r7ikou.html (参照 2022-1-10)。