

# コンピュータ・プログラムを書こう： 中学生を対象としたプログラミング講座

掛下 哲郎<sup>1</sup> 草場 聡宏<sup>2</sup> 杉町 信行<sup>1</sup>

**概要：**日本の中学校では 2021 年度からコンピュータ・プログラミングが必修化される。これを展望して、佐賀大学教育学部 附属中学校の生徒を対象としてプログラミング講座を行った。講座を受講した 33 名の中学生と 1 名の教員は、18 名のプログラミング未経験者を含むプログラミング初学者だったが、Code Studio のコース 2 を活用したコンテスト形式の講座では、90 分間の制限時間内に 30~70 問の問題を解いた。その結果、順次実行や入れ子のないループの学習を行うことができた。講座後のアンケートでも、課題はやや難しかったものの、プログラミングはとても楽しかったため、より高度な課題に挑戦したい、との意見が多数を占めた。本論文では、講座の紹介および生徒から得たデータの分析を行い、中学校でのプログラミング教育を推進するための考察を行う。

**キーワード：**プログラミング教育, 中学生, Code Studio, scratch

## Let's Enjoy Computer Programming: Programming Workshop for Junior High School Student

TETSURO KAKESHITA<sup>†1</sup> TOKIHIRO KUSABA<sup>†2</sup>  
NOBUYUKI SUGIMACHI<sup>†1</sup>

**Abstract:** Mandatory computer programming education begins at junior high school from April 2021 in Japan. We created a programming workshop for students at Saga University Junior High School in view of this change of the education curriculum. Among the 33 students and a teacher taking this workshop, 18 students have had no programming experience. However, the workshop utilizing Course 2 of the Code Studio was quite successful. Students solved 30-70 quizzes within 90 minutes' contest. As a result, the students learned programming containing sequence and simple loop. Most of the students answered that although the quizzes are somewhat difficult, computer programming is quite enjoyable so that they are willing to challenge more advanced quizzes. In this paper, we introduce the workshop and analyze the data collected from the students. Such experience is valuable to promote programming education at junior high school in Japan.

**Keywords:** Programming Education, Junior High School Student, Code Studio, scratch

### 1. はじめに

学習指導要領の改訂に伴い、2021 年度から中学校でもプログラミング教育が全学年で必修化される[1]。これを展望して、佐賀大学教育学部附属中学校（以下、附属中学校と略記する）の生徒 33 名および教員 1 名を対象とするプログラミング講座を実施し、良い評価を得たので報告する。

本プログラミング講座は、附属中学校 校友会が企画した講座「佐賀大学の先生の授業を受けてみよう」の一環として開講した。本講座は、自分の将来を考え始めた中学生時代に、大学での様々な専門分野の講義を受けることによって、学問的な探求の芽を育てるとともに、将来について考える機会を与えることを目的としており、佐賀大学の教員が自主的に協力することで運営されている。

著者らは、今回初めて「コンピュータ・プログラムを書こう」と題する 2 時間の講義を開講したが、今回の経験を通じて様々な知見を得ることができた。また、授業に当た

っては様々な工夫を施したため、生徒や授業に同席した保護者・教員の評価も高かった。

以下、2. ではプログラミング演習の際に用いた Code Studio を紹介する。3. ではプログラミング講座の内容と工夫点をまとめる。4. では講座を受講した生徒のプロフィールを分析する。分析結果は、一般的な中学生のコンピュータリテラシーを把握する上で有用だと考えられる。5. では、プログラミング講座の学習効果を分析する。多くの生徒はプログラミング未経験者だったが、90 分の制限時間内に 30~70 問の問題を解いた。また、講座後のアンケートでも、課題はやや難しかったものの、プログラミングはとても楽しかったため、より高度な課題に挑戦したい、との意見が多数を占めた。6. では生徒、教員および保護者から収集した意見を紹介する。肯定的な意見が多数得られたことから、今回の取り組みは成功だと考えられる。7. では全体をまとめ、プログラミング教育の導入に向けた課題や展望を示す。

<sup>1</sup> 佐賀大学 理工学部 情報部門  
Saga University

<sup>2</sup> 西九州大学 こども学部  
Nishi Kyushu University

## 2. Code Studio

Code.org は 2013 年に米国で立ち上げられた非営利団体である。全ての学校（初中等教育）において、全ての生徒がコンピュータサイエンスを学ぶ機会を提供し、女性やマイノリティが参加する機会を増やすことを目的としている。

Code Studio (<https://studio.code.org/courses>) は、Code.org が運営する無料のプログラミング教育サイトであり、scratch[4]をベースとするプログラミングを Web 上で段階的に学習できるように構成されている。Code Studio はコンピュータサイエンス入門や Hour of Code から構成されている。コンピュータサイエンス入門は難易度の異なる 4 つのコース（下記）から構成されており、全てを学習するには約 20 時間を要する。

**コース 1:** 小学校 1 年生を対象としており、ドラッグ&ドロップ操作、ステップの順次実行、デバッグ、ループの基本概念を学ぶ。

**コース 2:** 小学校 2~5 年生を主な対象としており、ステップの順次実行、ループ、条件文、デバッグについて学ぶ。

**コース 3:** コース 2 を終了した生徒を対象としており、関数、条件文、入れ子のループ、デバッグについて学ぶ。主に小学校 4~5 年生を想定している。

**コース 4:** コース 2 と 3 を終了した生徒を対象としており、両コースで学んだ基本概念を組み合わせるプログラミングを行うスキルを学ぶ。本コースを終了すると、生徒はループやパラメータを持つ関数を含む、複雑なスキルを要するプログラミングが行えるようになる。主に小学校 4 年生~中学生を想定している。

一方、Hour of Code はその名のとおりに 1 時間程度で終了する複数のチュートリアルから構成されており、一般的にはこちらの方が有名である。

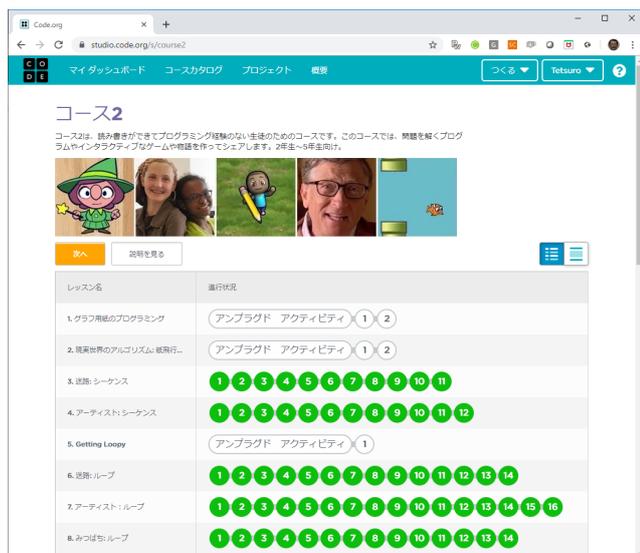


図 1 コンピュータサイエンス入門 コース 2  
Figure 1 Course 2 (Introduction to Computer Science).

今回のプログラミング講座では、生徒が講座終了後に自宅等で演習の続きに取り組むことを期待して、コンピュータサイエンス入門コース 2 (図 1) を使用した。コース 2 は 19 のレッスン（うち 8 つのレッスンはアンプラグド アクティビティ）から構成されている。プログラミングが必要なレッスン、課題の種類および問題数を表 1 に示す。コース 2 は合計で 139 の問題を含む。問題は徐々に難易度が上がるように構成されており、親しみやすいテーマを設定することで生徒の興味も持続しやすい。

表 1 コース 2 のプログラミングレッスン

Table 1 Programming Lessons for Course 2.

レッスン名	課題の種類	問題数
3. 迷路	順次実行	11
4. アーティスト	順次実行	12
6. 迷路	ループ	14
7. アーティスト	ループ	16
8. みつばち	ループ	14
10. みつばち	デバッグ	11
11. アーティスト	デバッグ	12
13. みつばち	条件文	15
16. フラッピー	ゲームの作成	10
17. プレイラボ	お話の作成	11
19. アーティスト	入れ子のループ	13

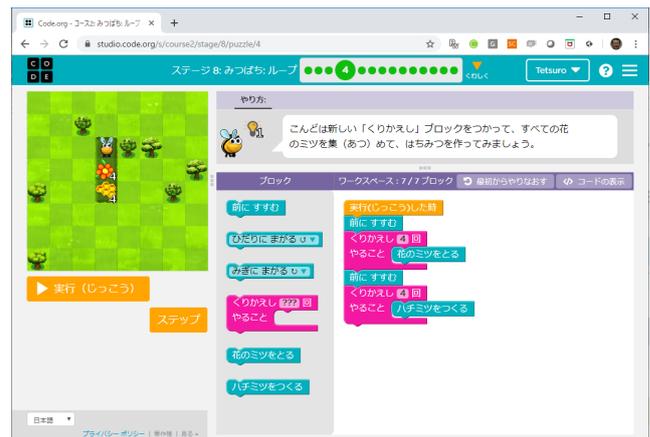


図 2 定型課題の例

Figure 2 Programming Quiz (Example).

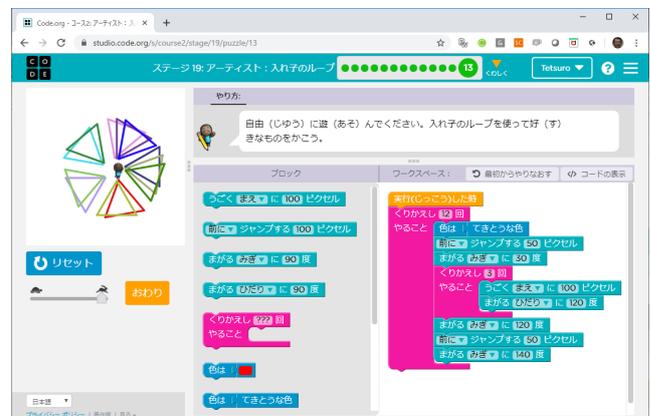


図 3 自由創作課題の例

Figure 3 Free Play Programming Quiz (Example).

コース 2 の課題は定型課題 (計 134 問, 図 2) と自由創作課題 (計 5 問, 図 3) に大別される. どちらも問題文は画面右上に示されており, 中央のブロックを右側のワークスペースにドラッグ&ドロップすることでプログラムを作成する. 図 2 のワークスペースに表示された「7/7 ブロック」は「使用ブロック数/正解のブロック数」であり, 生徒が自らのプログラムをより簡潔にするためのヒントになる.

生徒が作成したプログラムは「実行」ボタンを押すと直ちに実行され, 結果もすぐに確認できる. 定型課題の実行結果が題意に合わない場合にはアドバイスも表示されるため, 生徒はそれを見ることで, 自力で修正もできる. 課題によっては「ステップ」ボタン (ステップ実行) や速度調節スライダが表示され, プログラムの実行を制御できる.

なお, 画面左下で数多くの言語を選択できるため, 外国人でも学習できる. 本サイトの日本語は英語版の翻訳のため, 不自然な表現や誤訳も散見されるが, 中学生でもほぼ問題なく理解できるレベルである.

### 3. プログラミング講座

講座「佐賀大学の先生の授業を受けてみよう」では, 20 講座が開講された. 多くの講座では受講者数が 15~25 名だったが, 本講座は 33 名の生徒と 1 名の教員が受講した. 受講者の規模は典型的な中学校の 1 クラス分に相当するため, 本講座を通じて得られた知見は, 中学校で同様の授業を行う際にも参考になると考えられる.

本講座の授業風景を図 4 に示す. 当日は生徒と教員の他に生徒の保護者 8 名が授業を参観した.



図 4 授業風景

Figure 4 Class Overview.

#### 3.1 講座の内容

本講座は 2019 年 9 月 7 日 (土) 10:00~12:00 に佐賀大学で開催された. 2 時間の講座の予定は以下の通りであった.

- 講義「コンピュータ・プログラミングとは?」(15 分)
- 佐賀大学 LAN (eduroam) への接続 (15 分)
- 説明「Code Studio で学ぶプログラミング」(15 分)
- プログラミング演習 (コンテスト形式, 60 分)
- アンケート回答, 終わりに (15 分)

受講者が中学生であることを考慮して講義や説明の時

間は極力短縮し, プログラミング演習の時間を長めにした.

実際の講座では生徒が開始時刻よりも早く来たため, 佐賀大学 LAN への接続を講義前に済ませた. また, Code Studio の説明は, 生徒に PC を操作させながら行ったため, 生徒は自主的にプログラミング演習を始めた. その結果, プログラミング演習の時間を約 90 分間取ることができた.

#### 3.2 事前準備

受講者にはインターネットに接続できるノート PC の持参を求め, 受講者名簿に基づいて佐賀大学学内 LAN 接続用のユーザー ID とパスワードを事前に取得した.

自宅のノート PC を借りることができなかった生徒が 3 名居たため, 個人賠償責任保険への加入を条件に大学のノート PC を貸し出すことにした.

また, プログラミング演習のサポートを行うため支援員 (准教授 1 名, 助教 2 名) を配置した. 支援員はそれぞれ 11 名の生徒を担当した. また, 支援員はコンピュータサイエンス入門 (コース 2) の課題を全て解いておき, 生徒の質問に直ちに回答できるように準備した.

さらに, 生徒のコンピュータリテラシーの程度を調査するとともに, 本講座に対するフィードバックを得るため, アンケートを準備した.

#### 3.3 プログラミングコンテスト

プログラミング演習は, 競争の要素を含めることで生徒の意欲を引き出すことを狙い, コンテスト形式で行うこととした. そのため, コンテストのルールを設定した.

課題の複雑さや難易度が段階的に上がることを考慮し, 受講者には基本的にレッスン 3「迷路」から順に課題に取り組ませる. ただし, 自由創作課題については, 生徒の判断により飛ばして先に進んでも良いこととした.

定型課題は Code Studio が自動採点する. 最小限のブロック数で問題を解いた場合は 1 点/問題, より少ないブロック数の正解がある場合は 0.5 点/問題の評価を与えることとし, 合計得点によってベストプログラマ賞の選考を行う.

一方, 自由創作課題については, プログラム完成時に支援員が 2 段階評価することとした. 評価基準は「完成し工夫点が見られる (2 点)」「完成し一応動作する (1 点)」「その他 (0 点)」である. 各支援員は担当する生徒からの申告を受けてプログラムを評価する. 評価結果に基づいてベストクリエイター賞の選考を行う.

プログラミングコンテストの評価は本講座終了後に集計し, 成績上位者 (全体の約 35%) には賞状を授与した.

### 4. 受講者のプロフィール

本節ではアンケート回答を基に 33 名の生徒の PC 利用経験やコンピュータリテラシー, プログラミング経験などをまとめる. アンケートは本講座を受講した生徒全員から回収した.

図 5 には受講者の学年比を示す. どの学年でも男女比は

おおよそ半々であった。

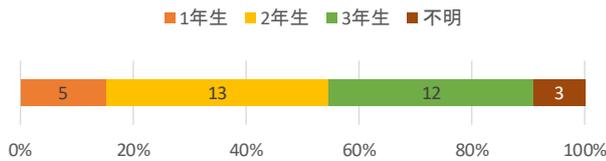


図 5 受講者の学年比  
Figure 5 School year of participants.

生徒のパソコン使用頻度を図 6 に示す。学年ごとの使用頻度では、使わない生徒 3 名のうち 2 名が 1 年生（1 名の学年は不明）であり、1 年生は高々週 1 回程度の使用頻度である。これに対し、3 年生の 75%，2 年生の 70% が週 1 回以上使用している。

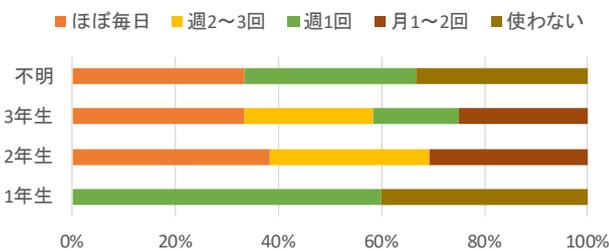


図 6 パソコンの使用頻度  
Figure 6 Frequency to Use Personal Computer.

インターネットの閲覧に関しては、どの学年においても「いつも閲覧」が過半数をこえており、ほぼ全員が何らかの形で閲覧している（図 7）。

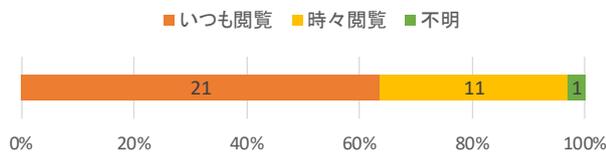


図 7 インターネットの閲覧  
Figure 7 Browsing of the Internet.

表 2 には LINE の送信についての回答を示す。1 年生の母数は少ないが、LINE を自分で送信できるのは、1 年生が 60% なのに対し、3 年生は 75% に達する。電子メールの送信についても LINE の場合とほぼ同様である。また、インターネットをいつも閲覧している生徒の中で LINE あるいは電子メールを送信できる生徒の割合は 76% だが、時々閲覧する生徒の場合は 45%（電子メールは 36%）に留まる。

表 2 LINE の送信（人数）

Table 2 Sending of the LINE (number of persons).

	自分で送信できる	使わない
1 年生	3	2
2 年生	8	5
3 年生	9	3
不明	2	1

MS-Word, MS-Excel の活用スキルに関する回答を図 8 及

び図 9 に示す。MS-Word を使ったことがない生徒は全体の 33% であり、MS-Excel の場合は 46% であった。1 年生の大半はどちらも使っていないが、高学年になるほど、各種の機能を使っている生徒の割合が高い。

図 8 MS-Word の活用スキル  
Figure 8 Skill Level to Use MS-Word.

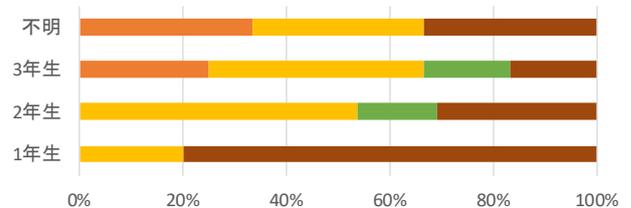


図 8 MS-Word の活用スキル  
Figure 8 Skill Level to Use MS-Word.

図 9 MS-Excel の活用スキル  
Figure 9 Skill Level to Use MS-Excel.

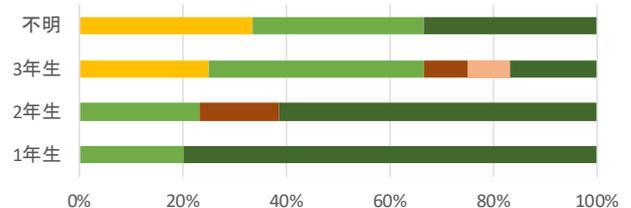
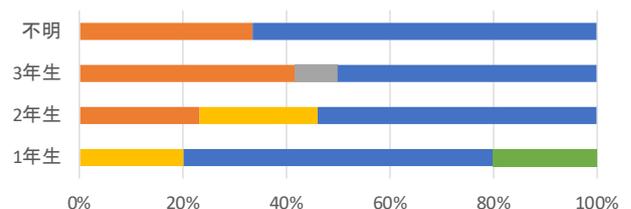


図 9 MS-Excel の活用スキル

Figure 9 Skill Level to Use MS-Excel.

図 10 プログラミング経験  
Figure 10 Programming Experience.



プログラミング経験に関する生徒の回答を図 10 に示す。他人の作業を観察しただけの生徒を含む未経験者は全体の 67% であり、1 年生はほぼ全員が該当した。100 行以上のプログラムを作成したことのある生徒はいなかったが、100 行未満のプログラム作成の経験者は、技術科の授業で経験した生徒を含めると 2 年生で 23%、3 年生で 50% だった。

## 5. プログラミング講座の効果

本節では、生徒が正解したパズル数およびアンケート回答に基づいて、本講座の教育効果を分析する。その際に 4.

で収集した受講者のプロフィールとの関係も併せて分析する。講座の効果に関するアンケート項目は、以下の通りである。

- 正解した課題数（クリアしたステージおよびパズル番号を問い、回答に基づいて算出した）
- 課題の難易度について
- プログラミングの楽しさについて
- 自由創作課題の楽しさについて
- より高度な課題に取り組む意欲

### 5.1 クリアしたパズル数に基づく達成度分析

2. で述べたように、今回実践したコース 2 には合計で 139 の問題がある。我々は、生徒が回答した「クリアしたレッスン名（ステージ）とパズル番号」に基づき、「クリアしたパズル数」を算出した。なお、回答者中 3 名は同項目に未回答だったため、分析対象は 30 名である。

#### クリアしたパズル数と学年の関係

図 11 に、クリアしたパズル数毎の生徒数分布を学年別に示す。クリアしたパズル数の最小値は 32、最大値は 67、平均値は 46.4、中央値は 46 であった。多くの生徒がプログラミング未経験者だったが、約 90 分の演習時間の間に約 2 分/問のペースでテンポよく問題を解いたことがうかがえる。

学年不明の 3 名を除き各学年の平均値  $m$  を求めたところ、1 年生 ( $n=5, m=43.2$ )、2 年生 ( $n=13, m=46.0$ )、3 年生 ( $n=9, m=48.0$ ) であった。クラスカル=ウォリス検定による多重比較を行ったが、学年間の平均値に有意な差はなかった ( $\chi^2 = 3.083, df=2, p=0.2141$ )。

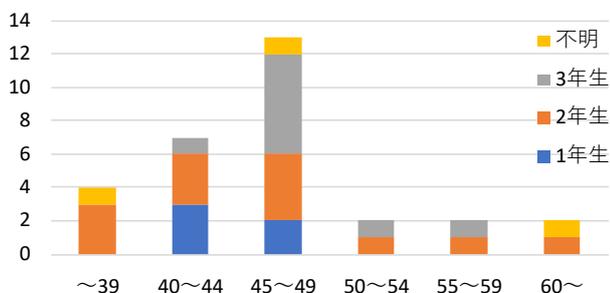


図 11 クリアしたパズル数毎の生徒数分布

Figure 11 Student Distribution vs Number of Cleared Puzzles.

中央値の生徒の場合、「7. アーティスト」のパズル 9 まで進んだことになる。このレベルは順次実行の課題が完了し、入れ子でない単純なループの学習が半分程度進んだレベルである（表 1）。最も進んだ生徒の場合、「10. みつばち」のパズル 4 まで完了した。この場合、ループの学習も完了し、デバッグ（与えられたプログラムの誤りを探して修正）の学習を開始した段階である。プログラミング未経験者が多数を占める中で、90 分程度のプログラミング演習としては、十分な成果を挙げたと考えられる。

#### クリアしたパズル数と PC 使用頻度の関係

PC の使用頻度別にクリアしたパズル数の差が生じるか否かを検討するために、「使わない」、「月に 1~2 回」を低頻度群、「週に 1 回」~「ほぼ毎日」を高頻度群として、クリアしたパズル数の差について検定を行った。両群の等分散性を確認するために F-検定を行ったところ、両群が等分散でないことが示された ( $F=2.9652, p=0.04634$ )。そこで Welch の t 検定を行ったところ、両群のクリアしたパズル数には有意な差はなかった ( $t=-0.9043, p=0.1936$ )。

#### クリアしたパズル数と MS-Word (MS-Excel) 活用経験の関係

MS-Word や MS-Excel の活用経験によってクリアしたパズル数の差が生じるか否かを検討する。

MS-Word を「使わない」と回答した生徒を未活用群 ( $n=10, m=41.8$ )、その他を活用群 ( $n=20, m=48.7$ ) として、クリアしたパズル数の差について検定を行った。F-検定を行ったところ、両群が不等分散とはいえないことが示された ( $F=0.4208, p=0.1843$ )。続けて、t 検定を行ったところ、両群のクリアしたパズル数の差は有意であることが示された ( $t=-2.044, df=28, p=0.02525$ )。

MS-Excel の活用スキルについても同様に分析する。MS-Excel についても「使わない」を未活用群 ( $n=14, m=41.6$ )、その他を活用群 ( $n=16, m=50.6$ ) として検定を行った。F-検定の結果、両群が不等分散とはいえないことが示された ( $F=0.6891, p=0.5061$ )。続けて、t 検定を行ったところ、両群の間には有意な差が見られた ( $t=-3.057, df=28, p=0.002435$ )。

MS-Word と MS-Excel の未活用群、活用群について回答数のクロス集計を表 3 に示す。一方のみの活用者は少なく、両方を活用する生徒と、どちらも活用しない生徒に二極化していることが分かる。Office ソフトウェアの活用状況によってプログラミング能力に差が出るのは自然なことと考えられる。

表 3 MS-Word と MS-Excel の活用度合いの関連

Table 3 Relationship between MS-Word and MS-Excel Usage.

		MS-Excel		合計
		未活用群	活用群	
MS-Word	未活用群	9	1	10
	活用群	5	15	20
合計		14	16	

#### クリアしたパズル数とプログラミング経験の関係

プログラミング経験が「未経験」、「見学」、「技術の授業で少し」の生徒を未経験群 ( $n=21, m=45.2$ )、「作成経験あり（100 行未満）」を経験群 ( $n=8, m=49.8$ ) として両群のクリアしたパズル数を比較する。本項目に対する未回答者が 1 名いたので、分析の対象は 29 名である。

まず F-検定を行い、両群が不等分散とはいえないことが示された ( $F=0.8122, p=0.6642$ )。そこで t 検定を行ったが、両群のクリアしたパズル数には有意な差はなかった ( $t=-$

1.1832,  $df=27$ ,  $p=0.247$ ). このことは、多少のプログラミング経験の差は、本講座のように通常の授業等とは形態の異なるプログラミング学習では問題にならないことを示唆している。

## 5.2 課題の難易度に対する回答分析

5.1 で行った「クリアしたパズル数」の場合と同じく、「課題の難易度」(取り組んだ感想)について分析する。なお、全回答者中 1 名は本項目に未回答だったため、分析対象者は 32 名である。

全体的な回答分布を図 12 に示す。「やや難しかった」の回答が多かった (32 名中 23 名 71.8%)。しかし、学年による回答分布の差は見られなかった。また、PC 使用頻度、MS-Word の活用経験、MS-Excel の活用経験、プログラム作成経験による影響についても検討したが、回答分布の差は見られなかった。

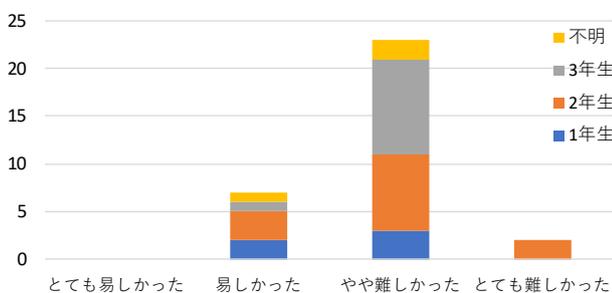


図 12 課題の難易度  
Figure 12 Difficulty of the Puzzle.

## 5.3 プログラミングの楽しさに対する回答分析

「プログラミングの楽しさ」について分析する。分析対象は 33 名である。学年別の「プログラミングの楽しさ」を図 13 に示す。「とても楽しかった」と回答した生徒が多かったこともあり、学年による回答の違いは見られなかった。「やや苦しかった」「とても苦しかった」と回答した生徒はいなかった。

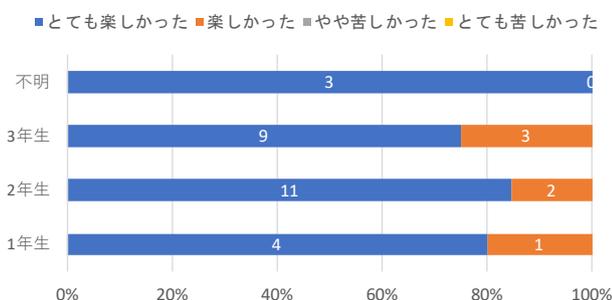


図 13 学年×プログラミングの楽しさ  
Figure 13 Grade × Enjoyment of Programming.

## 5.4 自由創作課題に関する分析

33 名の生徒を対象として自由創作課題の取り組みについて分析する。

学年別の「自由創作課題への取り組み」に対する回答分布を図 14 に示す。多くの受講者が自らの創造性を働かせて自由課題に取り組み、充実感を味わったようである。

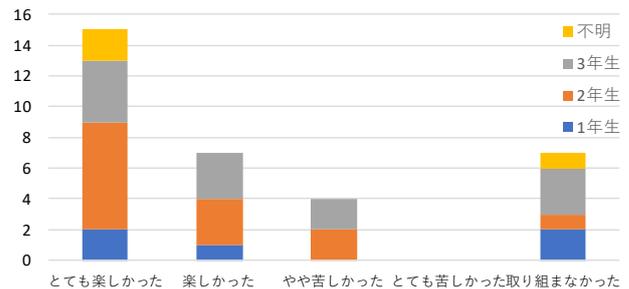


図 14 自由課題への取り組み  
Figure 14 Challenge for Free Play Puzzle.

自由創作課題については、生徒に自由に取り組みさせた。完成した時点で挙手をさせ、担当する支援員が評価した。評価基準は、一応完成した場合 1 点、完成し工夫が見られる場合 2 点とした。プログラミング演習の間に生徒が取り組んだ自由創作課題は、「アーティスト」の 2 問 (問題 4-10 および 7-12) であった。

問題 4-10 (レッスン 4 パズル 10) では、以下の操作を組み合わせて自由に絵を描かせた。ループは含まない。

1. 指定したピクセル数だけ [前/後] に動く。
2. [右/左] に指定した角度だけ曲がる。
3. 指定したピクセル数だけ [前/後] にジャンプする。
4. 12 色の中から色を指定する。
5. ランダムに色を指定する。

一方、問題 7-12 (レッスン 7 パズル 12) では、上記の 5 操作に加え、繰り返しを含む以下の 4 操作を組み合わせて自由に絵を描かせた。

6. 指定した回数だけ操作を繰り返す。
7. 線の太さを指定する。
8. 線の形状を指定する。
9. 線のアルファ値 (透明度) を指定する。

表 4 自由創作課題の評価分布

Table 4 Student Score for Free Play Programming.

評価：問題 4-10		評価：問題 7-12	
点数	人数	点数	人数
完成し、工夫が見られた	3	一応完成した	1
一応完成した	2	一応完成した	2
完成し、工夫が見られた	2	未完成・未着手	7
一応完成した	1	未完成・未着手	14
未完成・未着手	0	未完成・未着手	9
合計			
			33

これらの問題に対する評価分布を表 4 に示す。プログラミング演習の総時間が 90 分程度だったため、自由創作課題に取り組む時間は十分とは言えなかったと考えられる。2 つの問題とも未完成・未着手の生徒は 9 名だった。図 14 の結果と一致していないが、未完成の生徒が 2 名いたためである。

一方、短い時間ながらも、問題 4-10 を完成させ、かつ、

工夫が見られた生徒は 8 名いた。90 分の演習時間を延長すると、中学生の場合、集中力が続かなくなる可能性が高いが、日を改めて自由創作課題に取り組みさせることで、生徒の工夫を引き出すことも可能だと考えられる。

### 5.5 より高度な課題に対する学習意欲

「5.3 プログラムの楽しさ」を今回の授業の満足度とすると、「より高度な課題に取り組みたいか」をプログラミングに対する意欲として分析する。生徒に提示した選択肢は、「ぜひ取り組んでみたい」「取り組んでみたい」「あまり取り組みたくない」の 3 つである。

学年別のプログラミングに対する意欲を図 15 に示す。「あまり取り組みたくない」と回答した生徒はいなかった。全体として極めて高い学習意欲が示された。

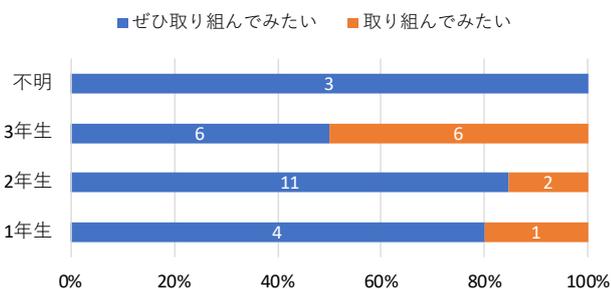


図 15 学年×より高度な課題に対する学習意欲  
 Figure 15 Grade × Motivation to Further Exercise.

また、満足すると自然に学習意欲が高まると思われる。これを確認するために満足度とプログラミングへの意欲に関する回答数のクロス集計を表 5 に示す。表 5 から「満足度が高い」と「意欲が高まった」の相関性を読み取ることができる。 $\chi^2$  検定を行ったところ、2 項目間に独立性はないことが分かった ( $\chi^2 = 15.331$ ,  $df=1$ ,  $p<0.000$ )。すなわち満足度と意欲は関係があることが示された。

表 5 プログラミングの楽しさ×学習意欲  
 Table 5 Enjoyment×Motivation for Programming.

	より高度な課題に		合計
	ぜひ取り組んでみたい	取り組んでみたい	
とても楽しかった	24	3	27
楽しかった	0	6	6
合計	24	9	33

### 5.6 考察

クリアしたパズル数をプログラミング講座の技能的な面での効果とするならば、その効果に学年による差はみられなかった。現在の中学生は、小学校時代にプログラミング教育の授業を受けてきたわけではなく、中学校の授業においても、現段階では実施されていないことから、ある意味当然と言える。PC の使用頻度により生徒の IT リテラシーに差があると思われるが、その差は、今回のクリアしたパズル数に影響を与えているとは言えない。しかし、Office 系ソフトウェア (MS-Word, MS-Excel) の活用度については、

クリアしたパズル数に有意な差が認められた。

次に課題の難易度(課題に取り組んだ感想)に関しては、72%の生徒が「やや難しい」と感じている。普段の授業等で経験したことのない学習体験であり、コンテスト形式を採用することで学習負荷をかけたことが原因と考えられるが、生徒の属性(学年、プログラミング経験)による差はみられなかった。

最後に、プログラムに対する楽しさ、今後に対する意欲については、受講したすべての生徒が「とても楽しかった」(82%)、「楽しかった」(18%)と答え、「より高度な課題」に対して、「ぜひ取り組みたい」(73%)、「取り組みたい」(27%)と肯定的に捉えている。その意味から、本講座は十分な成功を収めたと考えられる。

## 6. 生徒や保護者からのコメント

本講座を受講した生徒 33 名から収集したコメントの一部を以下に示す。基本的に極めてポジティブなコメントが多数得られており、生徒の学習意欲を引き出す上で十分な効果があったと考えられる。

- 時間を忘れるほど楽しかったです。
- とても楽しくて、知識も身に付けることができました。
- 家でも取り組んでみたいと思いました。
- 自分が最初に思っていたのよりも難しく結構苦戦した。でも楽しかったのもっとしたい。
- なかなか頭を使わないと解けないような問題があり、とても楽しかった。家でもこの続きをやってみます。
- 頭を使うことがたくさんあって、とてもいい機会になりました。家でもプログラミングをしてみたいです。
- 今回は、初めてプログラムを作成したが、思ったより易しく、とても楽しかった。家でも取り組みたいと思った。来年もぜひ来たいです。
- プログラミングは難しい印象があったが、意外と簡単で、プログラミングへの概念が変わった。
- 簡単なプログラムでも、様々な動きができることが分かった。
- 解いていくにつれて、少しずつ問題が難しくなっていくので、飽きずにずっとできた。
- 自分の考えたデザインをプログラミングするのも楽しかった。手が込んだことをすればするほど、細やかで具体的に動くので面白いなと思いました。
- プログラミングは難しいけれども、誰でも挑戦できて、人間の可能性を広げる重要なものだった。
- 先生の話が分かりやすかった
- 初めて行なったプログラミングだったけど、説明通りにすることで、分かりやすくプログラミングできた。
- コンピュータそのものが今後の社会のカギになり、それに関する仕事も増えると思った。
- IT 技術者のすごさを改めて感じた。「IT 技術者は社会の

命を握っている」という話が良かった。

- 自分は将来、プログラミングに関係のある職業に就きたいと思っていたので、大変なためになった。

一方、保護者(8名)からは以下のコメントが得られた。

本講座は、ITやプログラミング教育に対する保護者の意識を変える上でも効果があったと考えられる。

- 生徒が真剣にプログラミングに取り組んでいて良かった。頭の柔らかい子供たちは、どんどん世界を広げてゆくとと思う。
- 日常生活で楽しむ程度にゲーム制作等をする事は、頭を鍛え、物事を論理的に考える訓練になると思った。
- 先生が丁寧に授業をされていて、プログラミングがとても楽しそうだった。プログラミングは、難しく考えず楽しんでやるものだと思った。
- やり方が分かったので、家で子供とやってみたい。コンテスト形式にしていたので、やりがいもあったと思う。
- コンピュータは、本当にアイデア次第で可能性は無限だと実感した。これからはPCを使う時に、どのようなプログラムが動いているか興味を持つと思う。

また、本講座に参加した附属中学校教員からも良好な評価を得ることができ、次のコメントを頂いた。「プログラミングの学習は考え方を学ぶものだと思います。来年からのプログラミングの学習でも(本講座と同様に)楽しく、簡単にできるようになればいいと思います。パソコンをもっと安く使うことができればいいですね。」

本講座について、附属中学校が生徒および保護者から取ったアンケートでの評価は、5段階評価の最高評価が90.5%(38名)に上った(N=42)。また、回答者の73.8%(31名)からは来年度も本講座を受講したいとの回答があった。

## 7. おわりに

Code Studio を用いると、インターネットに接続できるパソコンがあれば、誰でも無料でプログラミング学習を行うことができる。教師用アカウントを作成することで、生徒のアカウントを複数発行して、生徒の進捗状況を把握できる。小学生から高校生までの幅広い学年にも対応している。しかし、日本の学校インターネット環境では、フィルタリング(有害サイトアクセス制限)のため、Code Studio にアクセスできないケースも多く見られる。この問題を解決すれば、小中学校等での導入も比較的容易だと考えられる。

小中高校でのプログラミング教育については、様々な教材や指導案が提案されている[2,3]。Scratch[4]やビスケット[5]等、小学校からのプログラミング教育を志向したプログラミング言語や学習教材も多数提供されている。しかし、完成した教材を用いて他教科の学習を行うことを主目的としたものが多い。また、1からプログラミングを行うのは、未経験者にとってハードルが高い。その点、Code Studio は

数分程度で解けるプログラミング課題を多数用意し、段階的に難易度を高めることで、生徒が自然にプログラミング能力を身に付けられるよう配慮されている。自由創作課題を通じて、生徒のアイデアを実現することもできる。

プログラミングを学ぶことで、コンピュータを使いこなし、これまで手作業で行ってきた仕事を自動化できる。また、プログラミングに習熟すると、様々なアイデアをソフトウェアで実現できる[6]。さらに、プログラムがコンピュータに対する指示であることから、プログラミング教育のもう一つの目的は、他者に的確な指示が出せる人(つまりリーダー)を育成することにあると著者らは考えている。複雑化する現代社会において、他者と協力することで様々な価値を創造し、より良い社会を築くリーダー人材を育成するためにも、プログラミング教育の重要性は高い。

小学校等においては、プログラミング教育を担当する教員の多くがプログラミング未経験者であり、スキル不足が大きな課題である[7,8]。一方で、教員は極めて多忙であり、スキルアップに取り組む時間の確保が難しい[9]。本講座で使用した Code Studio は、教員のプログラミング能力を高めるためにも活用できる。著者らは、教員のスキルアップを目的としたプログラミング講座も実施した経験があるが、それについては稿を改めて報告することとしたい。

**謝辞** 本講座の指導員としてご協力頂いた前田明子さん(佐賀大学 助教)および村田雅也君(西九州大学 4年生)に謹んで感謝の意を表します。

## 参考文献

- [1] 林文部科学大臣, “Society 5.0 に向けた人材育成の推進”, 未来投資会議資料, 2018年5月.
- [2] 日本作業技術教育学会編, “小・中・高等学校でのプログラミング教育実践: 問題解決を目的とした論理的思考力の育成”, 九州大学出版会, 2019.
- [3] 文部科学省, 小学校を中心としたプログラミング教育ポータル, <https://miraino-manabi.jp/>, (参照 2019-10-20).
- [4] MIT Scratch Team, Scratch HP, <https://scratch.mit.edu/>, (参照 2019-10-20).
- [5] デジタルポケット, ビスケット, <https://www.viscuit.com/>, (参照 2019-10-20).
- [6] Resnick, M. et al., “Scratch: programming for all”, *Comm. ACM*, Vol. 52, No. 11, pp. 60-67, 2009.
- [7] 黒田 昌克, 森山 潤, “小学校段階におけるプログラミング教育の実践に向けた教員の課題意識と研修ニーズとの関連性”, 日本教育工学会論文誌, Vol. 41, No. Suppl., pp. 169-172, 2018.
- [8] 教育ネット, “小学校のプログラミング教育、先生の98%が「授業の実施に不安」”, 2019. <https://resemom.jp/article/2019/04/26/50334.html>, (参照 2019-10-20).
- [9] リベルタス・コンサルティング, “教員勤務実態調査(平成28年度)”, 文部科学省委託研究, 2018. [http://www.mext.go.jp/b\\_menu/houdou/30/09/1409224.htm](http://www.mext.go.jp/b_menu/houdou/30/09/1409224.htm), (参照 2019-10-20).