

Pythonを用いたプログラミング並びに データサイエンス導入カリキュラム実施の試み

齋藤朗宏・隈本覚・池田欽一・平山克己
北九州市立大学経済学部

1. はじめに

北九州市立大学経済学部経営情報学科(以下本学科)は、その名の通り経営学並びに情報科学を主たる専攻内容としている。そのため、カリキュラムには各種プログラミング、システム開発にかかわる授業が設定されてきた。ただ、本学科の学生は文科系学生に分類されるため、プログラミング導入授業は、理科系大学生に対するそれと比較して困難が多い。その理由について佐野(1998)では東京外語大を例にとり、「一般数学や数理論理学に関する講義がないために、演算子、論理演算子、条件式などのプログラミング用語の理解が難しいものと思われる」と指摘している。この点を踏まえ、本学科においては、隈本他(2014)、齋藤他(2015)で既報の通り、プログラミング入門のさらに前段階の授業として、2013年度より情報科学入門という科目を開講し、プログラミング言語 Scratch を使用した導入教育を実施してきた。

近年ではさらに、情報科学を含む複合分野として、データサイエンス教育に関する関心も高まっている。Davenport & Patil(2012)でデータサイエンティストを「21世紀最もセクシーな職業」と表現したことを一つのきっかけとして注目が高まり、大学教育の現場においても、データサイエンスを専攻する学部・学科は増えつつある。

これらの状況を踏まえ、本学科では2019年度より、プログラミング言語 Python を主に使用する言語と設定し、プログラミング並びにデータサイエンス導入カリキュラムを設定している。ここでは、このカリキュラムと実施状況について報告する。

1.1. 高等学校教育課程における情報教育の現状

社会の情報化への対応として、平成11(1999)年高等学校学習指導要領改訂により、高等学校に普通教科「情報」が新設、必修科目とされることとなった。この決定に伴い、2005年4月高等学校入学学生より「情報A」、「情報B」、「情報C」が実施された。それぞれの科目は、澤田(2008)によると、「教科書や実際の指導内容を踏まえ、教職員や専門家の間では情報Aを基礎的科目、情報Bを理系指向の科目、情報Cを文系指向の科目とする見方が強い」とされている。これらの科目の中で、プログラミングが実施されているのは情報Bのみであ

った。必修化はされた一方で、大学入試における主要科目に入っていないこともあってか、基礎的内容である情報 A の選択率が圧倒的に高く、佐藤(2012)によると、教科書発行部数ベースで見た場合、2003 年度段階で 83.9%、2012 年度段階でも 72.4%が情報 A を選択している状態であった。

この教科「情報」は、平成 21(2009)年高等学校学習指導要領改訂により整理統合され、2013 年 4 月高等学校入学生より、「情報と科学」と「社会と情報」の 2 科目のうちいずれかが必修という形になった。その内容としては、佐藤(2012)では、情報 A の内容のうち情報機器の取り扱いの部分が義務教育に移り、情報 A と情報 C から社会と情報、情報 A と情報 B から情報と社会と説明している。つまり、「情報と科学」ではプログラミングを扱い、「社会と情報」では扱っていないことになる。この両科目の設置割合は、重田他(2015)による調査では、社会と情報 57.2%、情報の科学 18.1%、代替科目への置き換え 23.2%となっており、やはりプログラミングを伴う科目の設置割合は低いことがわかる。

さらに平成 30(2018)年高等学校学習指導要領改訂により、2022 年 4 月高等学校入学生より、上記 2 科目からの選択制から、必修科目「情報 I」と選択科目「情報 II」という形に改められることになっている。この制度変更による大きな変化として、高等学校におけるプログラミング教育が必修化されたという点が挙げられる。

こういった状況の変化を踏まえると、現時点では主にプログラミング経験のない学生を対象とした教育を行いつつ、近い将来(2025 年 4 月)にはプログラミング教育を受けてきた学生が入学してくることに備え、より高度なプログラミング教育を実施する準備を行うことが大学の情報系学科には求められていると言える。

1.2. 大学におけるデータサイエンス教育

先述のようなデータサイエンスに関する関心の高まりを踏まえ、内閣府・文部科学省・経済産業省の 3 府省が連携し、2021 年に「数理・データサイエンス・AI 教育プログラム (リテラシーレベル)」認定制度が制定された。同制度の目的を文部科学省は「大学等の正規の課程であって、学生の数理・データサイエンス・AI への関心を高め、かつ、数理・データサイエンス・AI を適切に理解し、それを活用する基礎的な能力を育成することを目的として、数理・データサイエンス・AI に関する知識及び技術について体系的な教育を行うものを文部科学大臣が認定及び選定して奨励することにより、数理・データサイエンス・AI に関する基礎的な能力の向上を図る機会の拡大に資すること」と述べており¹、データサイエンス等を専門としない学部・学科における底上げを重視していることが確認できる。

上記のような状況の変化を踏まえると、文系であり、かつ情報教育を標榜している学科としては、学科全体への基礎的な教育と、より専門性を高めたい学生に対する高度な教育との両輪が求められていると言える。

¹ https://www.mext.go.jp/a_menu/koutou/suuri_datascience_ai/00002.htm

2. 北九州市立大学経済学部経営情報学科におけるカリキュラムの変更

上記の状況、また、各分野における統計・データ解析の応用が進む状況を踏まえ、本学科では 2019 年度新入生より新カリキュラムを導入した。その中で、データサイエンスに関連する主な変更は以下の通りである。

- 旧カリキュラムにおける選択科目「経営統計」を必修科目「経営統計 I」として 1 年生 2 学期に配置。
 - 旧カリキュラムにおける 3 年生選択科目「企業データ分析」を選択科目「経営統計 II」として 2 年生 1 学期に配置。
 - 旧カリキュラムにおける 1 年生必修科目「情報科学入門」と「プログラミング I」を整理し、Python をメインとした必修科目「情報科学入門」として 1 年生 2 学期に配置。
 - 新設科目として、選択科目「マーケティング・サイエンス」を 3 年生 1 学期に配置。
- これらの科目の関係性を表したのが図 1 である。

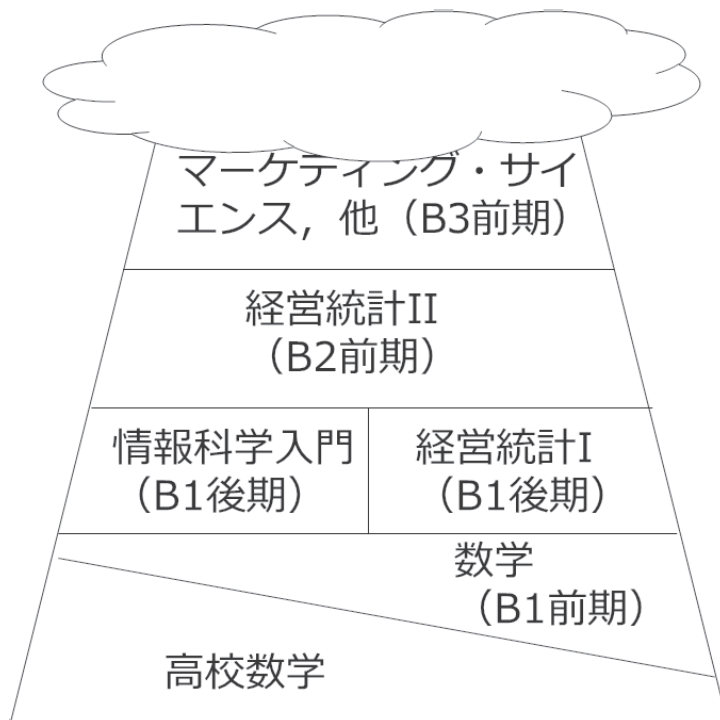


図 1 データサイエンスに関連する各種科目の関係性

この図はデータサイエンスという学問分野を 1 つの山に見立て、土台となる科目を積み上げていくことで、より高度な学びへと到達できることを示している。高校数学を土台とし、特に経済学・経営学における応用上重要である微分法、積分法を中心に「数学」において深

める。そして、「情報科学入門」で構造化プログラミングの考え方とプログラミング言語 Python の基礎を学び、同時期に「経営統計 I」で統計学の基礎的な理論を学ぶ。これらの学修を統合する科目として、「経営統計 II」にて、Python を用いた統計データ解析の基礎を学ぶ。最後に発展的な科目として、「マーケティング・サイエンス」にてデータサイエンス諸分野の中でも、特にマーケティングに着目した分析モデルを学び、それぞれが将来的に専門とするであろう場面、分野に応用出来る技術を身につけるといった流れになっている。

2.1. Python とデータサイエンス

Python は、1990 年代初めに Guido van Rossum により開発された言語²であり、特に近年の人気上昇は著しい。オランダの TIOBE 社が発表している、プログラミング言語に関する検索結果を基にした人気言語ランキング、「TIOBE Index」でも、Python は 2021 年 10 月に長年首位を維持してきた C 言語や Java を抑えて初めて首位に立ち、その後 2022 年 1 月に至るまで首位を堅持している³。

この Python は、中でもプログラミング導入教育並びに AI・データサイエンスの分野において特に注目されている言語である。間辺他(2019)では、C 言語と比較した Python の特徴の 1 つとして、反復や分岐のブロックをインデントで作成することを挙げ、これにより括弧などでブロックを作成するやり方と比較して、わかりやすくストレスを感じずに学べる可能性があることを指摘している。このインデントを用いたブロック作成という特徴は、教育という観点で言えば、見やすいプログラム作成のためのインデントの習慣づけにも繋がるであろう。

文部科学省の高等学校情報科「情報 I」教員研修用教材⁴では、プログラミングの考え方、事例を Python で示しており、「情報 I・II」でもかなりの割合で用いられる言語になると予想できる。実際、井出(2021)では各教科書における使用プログラミング言語を比較しており、教科書会社 6 社中、Python を用いた教科書を出版していないのは 2 社のみであった。

また、Python はデータサイエンティスト向けの環境が発達しており、データサイエンティスト向けに開発されたディストリビューションである Anaconda⁵や、学生、データサイエンティスト、AI リサーチャーなどを想定する主な使用者として挙げている Web での実行環境 Google Colaboratory⁶など、言語を習得すればすぐに分析に取り組むことができる状況になっている。

このように、Python には「プログラミング導入教育に適している」、「データサイエンスに向いている」という 2 つの大きな特徴があり、それらはいずれも本学科における教育目的に合致している。

² <https://www.python.org/>

³ <https://www.tiobe.com/tiobe-index/>

⁴ https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1416756.htm

⁵ <https://www.anaconda.com/>

⁶ <https://colab.research.google.com/>

2.2. 情報科学入門

このような背景, 理念から, それまでの Scratch を用いた導入教育を再構成し, 新しい導入教育科目「情報科学入門」を 2019 年度より実施している. その各回の内容は以下の通りである.

- 第 1 回 ガイダンス 【情報科学】とは
- 第 2 回 プログラム手順の考え方 【フローチャート】【Flowgorithm】
- 第 3 回 プログラミング入門。【Python】の起動と文字の表示【文字コード】
- 第 4 回 【変数】、【型】と代入
- 第 5 回 Python での四則演算、【順次構造】のフローチャートとプログラム
- 第 6 回 【分岐構造】のフローチャートとプログラム
- 第 7 回 【Microsoft Excel】でデータを見る
- 第 8 回 【numpy】入門。Excel で作成したファイルの読み込み
- 第 9 回 【配列】の扱い方
- 第 10 回 【反復構造】のフローチャートとプログラム
- 第 11 回 【フローチャート】とプログラムの読み取り
- 第 12 回 【平均値】、【最大値】、【最小値】の計算
- 第 13 回 応用プログラミング
- 第 14 回 プログラミング実習
- 第 15 回 まとめ
(同科目シラバスより)

まずフローチャートを学び, その上で変数と型, 構造化プログラミングにおける 3 つの構造を学ぶ. そして各種計算を実際に行うといった流れは Scratch か Python かという違いを除き, そこまで大きな内容的変化とはなっていない. しかし, 隈本他(2014), 齋藤他(2015)でも議論しているように, フローチャートを描いてプログラムの構造を作成することと, 実際のコードを作成することとが初学者にとっては必ずしも対応するものではなく, その理解の難しさが課題であった. そこで本授業では, その間を繋ぐものとして, Flowgorithm⁷ というソフトウェアを導入し, そのソフトを用いてフローチャートの作成を行った. Flowgorithm は Devin Cook により 2014 年に発表されたフローチャート作成と実行を行うソフトウェアである. 図 2 左側のようなフローチャートを直感的に作成することができ, また, Scratch のようにフローチャートをそのまま実行することもできる. また, 日本語を含めた多数の言語の表示と処理にも対応している. 図 2 左側のフローチャートをそのまま実行すると図 2 右側のようになり, 変数 **a** に 1 から 5 までの値を代入して表示する繰り返しが実行され, 日本語も問題なく出力されていることが確認できる.

⁷ <http://www.flowgorithm.org/>

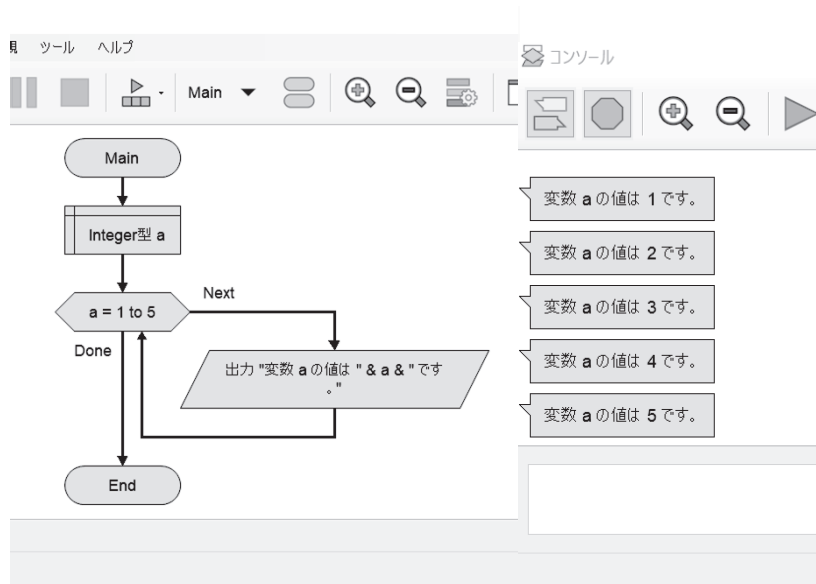


図 2 Flowgorithm で作成したフローチャートの例

ここまでであれば従来用いてきた Scratch でも十分に対応可能であるが、Flowgorithm では、これに加えて、Python も含めた多くのプログラミング言語で使用可能なコードを出力することができる。この点が、フローチャートからコード作成の間を繋ぐものとして、大きな意味があると考えている。実際に図 2 のフローチャートから Python のソースコードを作成した結果が図 3 である。このように、Python でそのまま実行可能なコードを簡単に作成することができる。

The screenshot shows a 'ソースコードビューア' (Source Code Viewer) window for Python. The code is as follows:

```

0  for a in range(1, 5 + 1, 1):
1  print("変数 a の値は " + str(a) + " です。")

```

図 3 Flowgorithm を用いて作成した Python のコード

2.3. 経営統計 II

経営統計 II では、経営統計 I と情報科学入門で学んだ内容を踏まえ、Python でデータ分析を実行することを主たるテーマとしている。この段階ではそこまで高度な分析手法を用いるわけではないため、基本的には Microsoft Excel などを使うことができれば分析自体は可能だが、Python を利用することで、統計理論をシミュレーションでも確認できるという点がメリットである。たとえば図 4 は、2 項分布の成功回数をシミュレートした結果である。

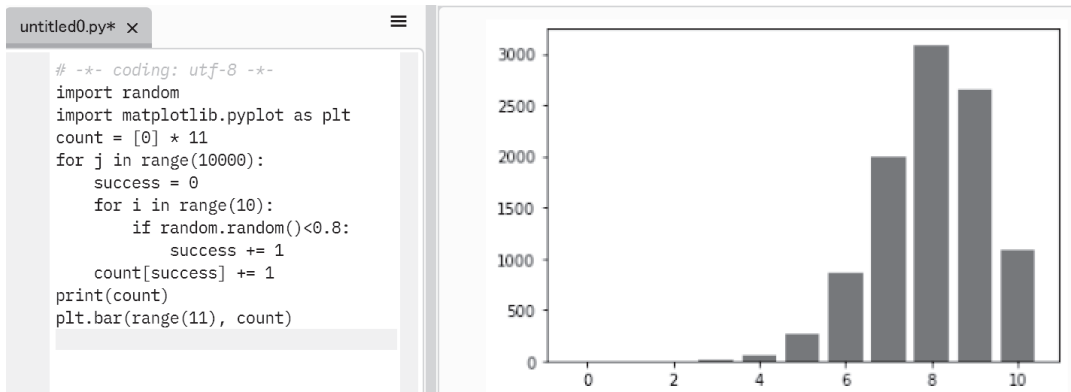


図 4 Python による 2 項分布の成功回数のシミュレーション結果

80%の確率で成功する試行を 0 から 1 の乱数で 0.8 未満になるか否かで表現し、10 回の試行から得られる成功回数を求める。その上で、それを 10000 回繰り返し、成功回数の分布を求めグラフ化している。この過程を順を追って実行することで、統計理論の理解を促すことを目的としている。

2.4. マーケティング・サイエンス

マーケティング・サイエンスは、本学科におけるデータサイエンス関連教育のゴールとすることを意図して設計した科目である。その観点から、経営系学科に所属する学生にとってデータサイエンスが応用できる分野としてマーケティングを取り上げ、マーケティング・サイエンスで用いられる分析手法の中から特にデータサイエンス分野で用いられる手法に絞って分析手法の説明と実習を行った。具体的な手法としては、階層的クラスタ分析やロジスティック回帰分析といった分類に関連する手法、アソシエーション分析のような大規模データの分析にかかわる手法である。

3.情報科学入門理解度と関心調査

3.1 調査実施概要

情報科学入門で目標としているプログラミングの基本的な考え方の理解が達成されているか、また、達成に不安のある点、講義の中でわかりにくかった点を確認し、かつ今後の学修に関連しての希望と、情報系各分野に関する興味関心の調査を同科目履修者に対して実施した。実施の概要は以下の通りである。

- 調査日時：2022年1月20日～27日
- 調査対象：本学科の情報科学入門3クラス
- 回答者数：111名
- 調査方法：Microsoft forms を用いたオンライン調査
- 質問項目
 1. 情報科学入門で学んだ各分野に関する理解度。全くできていない～よく理解できているまで5段階評価。
 2. 情報系科目に関する希望。専門的な技術を身につけるものが多い、日常生活で役立つ知識が欲しいなど。全く当てはまらない～よく当てはまるまで5段階評価。
 3. 情報系科目各分野に関する興味関心。Python プログラミングや Java プログラミング、通信ネットワークやデータサイエンスなど。全く興味はない～非常に興味があるまで5段階評価。

3.2 結果と考察

理解度に関する回答は図5の通りとなった。上から授業前半の内容、下に行くにつれて後半の内容となっており、前半のフローチャート作成や Python で文字を作成するプログラムの作成などはかなり理解度が高いが、後半、Python のパッケージの読み込みや配列作成などは、理解できていると回答した割合が50%を下回る結果となった。後半部分はデータサイエンスに関連する部分でもあるため、経営統計II以降と連携した教育が求められていることが確認できる。

情報科学系の授業に期待する事柄に関する回答結果は図6の通りであった。何よりも日常生活に役立つことと基本的な技術が中心であり、続いて多様な実践事例、最新の知識が欲しいという回答であったが、一方でプログラムを自分で作製する機会や専門的な技術と言った、スキルを身につけることに関する期待が低い結果となってしまった点は課題と言える。

情報科学系各科目に関する興味関心の強さを尋ねた結果が図7である。最近の状況を反映してか、Python プログラミングやデータサイエンスに対する興味は非常に高い。また、アプリケーション作成などへの興味が高いことも確認できる。こういった分野への興味関心を維持しつつ、基盤となる数学や統計学、ネットワークやデータベースといった分野に関心を広げていく教育が求められるだろう。

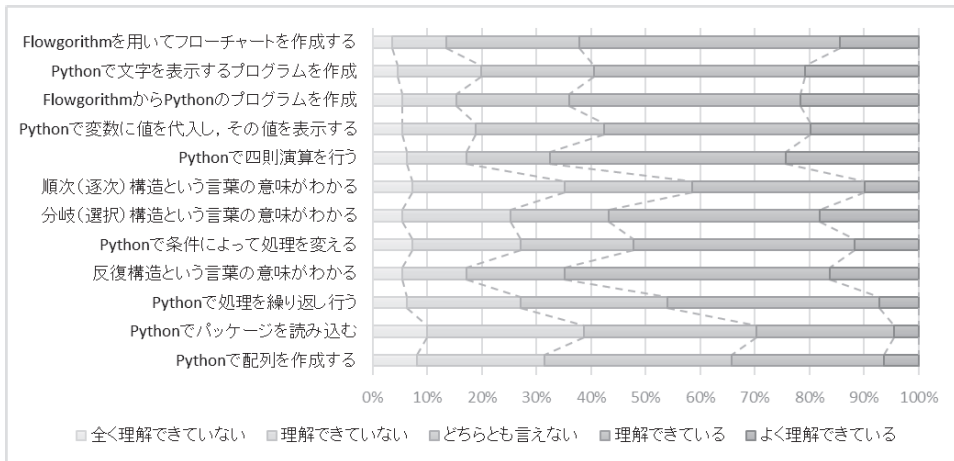


図 5 情報科学入門各分野の理解度

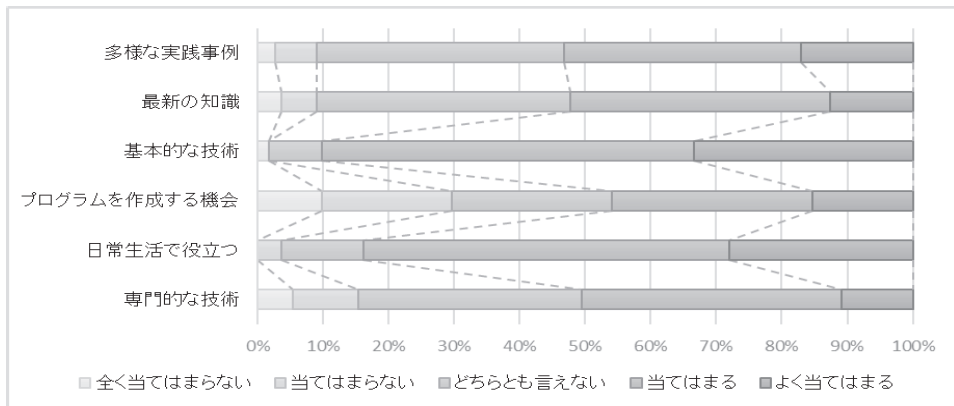


図 6 情報科学系の授業に期待する事柄

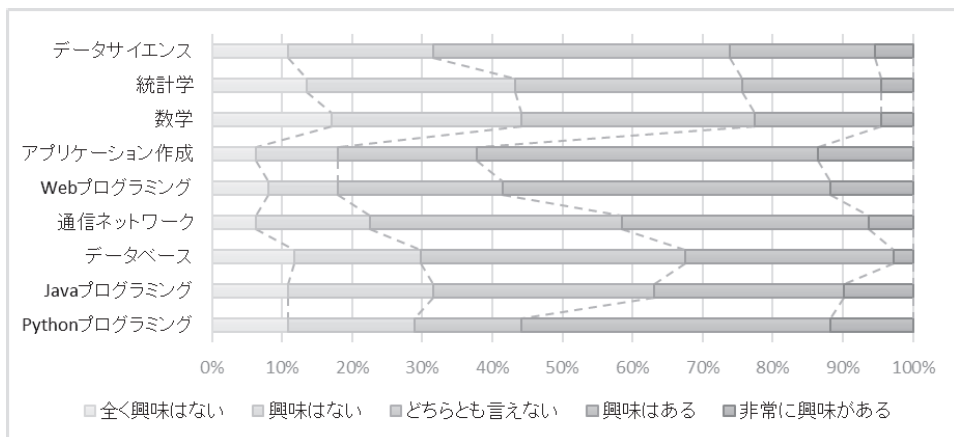


図 7 情報科学系の授業に期待する事柄

4. まとめと今後の課題

情報, AI・データサイエンス教育の現状を踏まえ, 文科系学生に対してプログラミングやデータサイエンスの能力を身につけて貰い, 各自が専門とするべき分野にその知識・スキルを活かすための一貫した教育プログラムを作成し, それを実施した事例について紹介した. 学生の興味関心は高く, この分野の知識の重要性自体は理解しているようだ. ただ, 現状では, アンケート結果にも現れているように, 専門的な知識や自分でプログラミングを行うスキルを身につけることに対して積極的な回答は得られておらず, 飽くまでも知識を習得すべき独立した一分野という認識が強いようだ. 今後は, プログラミングやデータサイエンスのスキルを身につけることが, その分野を専門としない学生にとっても, 専門を補強する形で有用であるということを伝えていく教育が求められるものと思われる,

参考文献

- 井出広康(2021). 「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」『情報教育シンポジウム』2021年8月, 246-253.
- 隈本覚・池田欽一・齋藤朗宏・平山克己(2014). 「Scratch を用いたプログラミング教育の実践」『北九州市立大学商経論集』49(3・4), 23-27.
- 齋藤朗宏・池田欽一・平山克己・隈本覚(2015). 「Scratch を用いたプログラミング導入授業の現状と課題」『北九州市立大学商経論集』50(1・2・3・4), 37-44.
- 佐藤万寿美(2012). 「高等学校全体の教科「情報」の状況について」『JUICE Journal』2021年度 No.1, 2-6.
- 佐野洋(1998). 「大学の文系学部におけるプログラミング教育の試み」『コンピュータと教育』50(6), 41-48.
- 澤田大祐(2008). 「高等学校における情報科の現状と課題」『調査と情報』604, 1-10.
- 重田桂子・植原啓介・村井純(2015). 「高校教科「情報」に関するアンケート調査と分析」『情報教育シンポジウム』2015年8月, 31-38.
- Davenport, T. H., & Patil, D. J. (2012). 「Data scientist.」 『Harvard business review』90(5), 70-76.
- 間辺広樹・長島和平・並木美太郎・長慎也・兼宗進(2019). 「C の学習経験を持つ高校生への Python の授業導入事例」『情報教育シンポジウム』2019年8月, 256-262.