

大学の一般教育としてのプログラミング教育における到達目標の検討

加藤 利康[†] 勝間田 仁[‡] 中村 一博[‡]日本工業大学 共通教育学群[†], 先進工学部情報メディア工学科[‡]

1. はじめに

2020 年度より小学校においてプログラミング教育が始まり, プログラミング能力の高い人材の育成が社会的に求められている. 初等教育では英国, ハンガリー, ロシアにおいてプログラミング科目が必修である. 英国では CAS (Computing At School) がプログラミングを含む教科「Computing」の学習の到達目標として Pathway を作成している[1]. この Pathway は, コンピュータの技術を 6 つのカテゴリー, 8 段階のレベルに分けており, プログラミングの技術に関する内容が含まれている.

初等中等教育でプログラミングを学んでくる学生に対して, 大学の一般教育としてのプログラミング教育をどのように位置づけるのかは重要な課題である. 大学入学時の学生のプログラミングに関する知識やスキル的前提条件は, 今後 10 年で大きく変化すると言われている[2]. 本研究の目的は, 大学の一般教育としてのプログラミング教育の到達目標を検討することである. その到達目標として CAS が作成した Pathway が応用できないかと考えた.

本研究では, まず, 日本国内の現在のプログラミング教育を取り巻く情勢から学習項目を抽出した. つぎに, 抽出した学習項目と CAS の Pathway を対比してプログラミング教育における到達目標を作成した. そして, 到達目標の検討として, 到達目標ごとに問題を作成し, 大学生に解答してもらって評価した.

2. プログラミング教育における到達目標

プログラミング教育を取り巻く情勢として, 本研究では, つぎの 4 種類の文献から学習項目を抽出した.

- 高等学校の教科書
文部科学省検定図書「情報の科学」

- 大学の参考書
日本工業大学のプログラミング系の授業で使用しているテキスト
- 資格試験
基本情報技術者試験, 日商プログラミング検定
- 日本国外
文部科学省が発行する「諸外国におけるプログラミング教育に関する調査研究」※ [3]
※学習項目については未掲載であった.

抽出した学習項目は, 表 1 である. ◎はほぼ全ての文献で採用, ○は半数以上採用, △は一部採用である. この結果から, 今後の大学の一般教育としては, 表 1 で挙げられた学習項目を全て含む到達目標を作成する必要がある.

表 1 の学習項目と CAS の Pathway を対比して, プログラミング教育における到達目標を作成した結果は, 表 2 である. 8 レベル, 32 の到達目標となった. このレベルは学習項目を系統的な内容と置き換えたときの段階のことである.

表 1 抽出した学習項目

	フロー チャート	値と リテラル	変数 と型	順次 分岐 反復	配 列	関 数	並べ 替え
高校	◎	◎	◎	◎	○	△	△
大学	◎	◎	◎	◎	◎	◎	△
資格	◎	◎	◎	◎	◎	◎	◎

3. プログラミング教育における到達目標の評価

到達目標のレベル分けが, 現在の大学生に即しているかを検討するため, 到達目標ごとに問題を作成して評価を行う. 1 つの到達目標について, 1 問となるように著者ら 3 名で合計 32 問を作成した. 問題形式は, 自由な解答を得たかったため, 全て手書きによる記述式とした.

学生については, 現在 C 言語を学んでいる 1 年生 12 名 (情報工学系 10 名, システム工学系 2 名) と, これから C 言語を学ぶ電気工学系 2 名の合計 14 名である. プログラミングの既有知識

A Study of Learning Attainments Targets in Programming Education as General Education in University

[†]Toshiyasu Kato, [‡]Masashi Katsumata, Kazuhiro Nakamura
Nippon Institute of Technology

表2 プログラミング教育の到達目標

レベル	No.	到達目標 ()内は正答率 n=14
1 (100)	1	フローチャートを読める(100)
	2	リテラルを理解する(100)
	3	順次アルゴリズムを設計できる(100)
2 (91)	4	データに型があることを理解する(79)
	5	分岐アルゴリズムを設計できる(100)
	6	繰返しアルゴリズムを設計できる(100)
	7	セマンティックエラーを対処できる(86)
	8	繰返しと分岐を組み合わせたアルゴリズムを表現できる(79)
3 (81)	9	変数の宣言および初期化ができる(79)
	10	プログラムの出力を予測できる(86)
	11	代入演算子を理解する(79)
	12	if-else 文を理解する(86)
	13	後判定の繰返し文を使ったプログラミングができる(79)
4 (76)	14	関係演算子を理解する(79)
	15	状況に応じて if 文と else 文を使い分けられるようにする(79)
	16	関係演算子を用いてループの途中で終了できることを理解する(71)
5 (55)	17	同じ問題に対して異なるアルゴリズムがあることを理解する(29)
	18	適切なデータの型を選択できる(57)
	19	様々な演算子を理解する(算術, 論理)(79)
	20	反復とは, ループなどのプロセスの繰返しであることを理解する(57)
6 (51)	21	関数の概念を理解する(64)
	22	一次元配列を理解する(50)
	23	ビット演算子を理解する(14)
	24	ネストされた分岐処理を理解する(79)
	25	シンタックスエラーを対処できる(50)
7 (55)	26	同様の問題に対して異なるアルゴリズムを書ける(64)
	27	変数にスコープがあることを理解する(57)
	28	前判定と後判定の繰返しを使い分けられる(43)
8 (20)	29	再帰を利用した問題の解決(0)
	30	処理を関数化して利用する(21)
	31	二次元配列を理解する(29)
	32	while と for の違いを理解する(29)

は問わず, 教員らとコミュニケーションが取れる学生を対象として, 問題解答をお願いした.

解答結果の正答率は, 表2の()内である. レベルが高くなるにつれて正答率が下がっている. また正答率がとくに低いのは以下の3つである.

- No. 23 「ビット演算子を理解する(14)」
- No. 29 「再帰を利用した問題の解決(0)」
- No. 30 「処理を関数化して利用する(21)」

4. 考察

レベルが高くなるにつれて正答率が下がっているため, レベル分けの到達目標は適正であると考えられる. 到達目標間では, 到達目標ごとの相関係数を算出したところ, No. 4 「データに型があることを理解する(79)」と相関係数が1.0になったものが, No. 11, 14, 19, 24である. 変数に対する理解が不十分であれば演算子に対する理解も不十分になると考えられ, 相関を考慮して系統的に分類する必要がある. また, 負の相関はなかった.

誤答を確認すると, 問題文を理解していない可能性があった. 例えば, 計算結果が30になる問題で, 「30以上なら Yes, 29以下なら No で答えよ. 」と聞かれて, 30と答えている. 単なるミスなのか, プログラミングの知識が不足しているのかは, さらに調査する必要がある.

5. まとめ

英国のCASが作成した「Computing」教科のためのPathwayに基づいて大学のプログラミング教育における到達目標を検討した. 高等学校の教科書や資格試験を調査して到達目標を作成し, その評価を14名の大学生に到達目標ごとに作成した問題に解答してもらうことによって行った結果, レベル帯は適正であった. 高い相関のある到達目標があったため, 今後は到達目標の構造化を行っていく.

参考文献

- [1] CAS Computing Progression Pathways KS1 (Y1) to KS3 (Y9) by topic
<https://community.computingatschool.org.uk/resources/1692/single> (Accessed 2020-1-2)
- [2] 布施 泉, 大学の一般教育としてのプログラミング教育, システム制御情報学会, システム/制御/情報, Vol. 62, No. 7, pp. 266-271, 2018
- [3] 文部科学省, 情報教育指導力向上支援事業(諸外国におけるプログラミング教育に関する調査研究)
http://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1408119.htm (Accessed 2020-1-2)