

# 工科大学での初年次プログラミング教育における TA/SA 評価のためのルーブリックの開発

大橋裕太郎<sup>†1</sup> 勝間田仁<sup>†1</sup> 中村一博<sup>†1</sup>  
石原次郎<sup>†1</sup> 橋浦弘明<sup>†1</sup> 松浦隆文<sup>†1</sup> 山地秀美<sup>†1</sup>

**概要**：大学でプログラミングを教える際、学生の進度や理解度に対応するため、ティーチング・アシスタント (TA) やスチューデント・アシスタント (SA) が授業支援を行うことが広く行われているものの、TA/SA に対する評価基準が曖昧であるという問題がある。本論文では、TA/SA の評価基準について検討するため、筆者らが所属する情報工学科で実施している1年生向けのプログラミング科目において、TA/SA がどのように授業支援を行っていたかを調査した。グループインタビューを行い、「事例—コードマトリクス」と呼ばれる質的分析の手法を援用して分析を行った。その結果、20種類の事例が抽出され、「問題の特定と分析」「教え方・伝え方の工夫」「TA/SA 間の連携・情報共有」「振り返り」の4つのカテゴリーが生成された。この結果を踏まえ TA/SA の活動を評価するためのルーブリックを作成した。

**キーワード**：プログラミング教育、初年次教育、情報教育、TA/SA、ルーブリック

## Developing a Rubric for Assessing Teaching Assistants and Student Assistants in the First Year Computer Programming Education in a Technical University

YUTARO OHASHI<sup>†1</sup> MASASHI KATSUMATA<sup>†1</sup>  
KAZUHIRO NAKAMURA<sup>†1</sup> JIRO ISHIHARA<sup>†1</sup>  
HIROAKI HASHIURA<sup>†1</sup> TAKAFUMI MATSUURA<sup>†1</sup> HIDEMI YAMACHI<sup>†1</sup>

### 1. はじめに

#### 1.1 プログラミング教育における TA/SA 制度の問題点

大学での実験や演習を伴う授業科目では、学生一人ひとりの進度や理解度に対応するため、ティーチングアシスタント (以下、TA) やスチューデントアシスタント (以下、SA) が授業支援を行う方法が広く取り入れられている (以下、両者を指す場合は TA/SA と表記)。情報学、情報科学・工学などを専攻する情報系学部・学科も同様である。例えばプログラミングを講義形式の科目で教える場合、学生の進度や理解度にばらつきがあるため教員だけでは手が足りないことが多く、TA/SA による個別対応が必要になる。プログラミング教育において、TA/SA 制度を導入することで教育効果が認められると複数の研究が指摘している [1] [2] [3]。

しかし、履修学生ではなく TA/SA の活動をどのように評価するかについては先行研究が少ない。文部科学省によれば、現状では TA に対する明確な評価基準はなく、TA 制度の評価システムの確立が望まれている [4]。授業運営側は履修学生への教育と評価を第一に行うため、TA/SA の授業支援活動の評価が疎かになってしまうと考えられる。TA/SA 制度は、履修学生だけでなく TA/SA 自身にとっても学習の

契機となりうる。苅谷は、TA の経験が研修的な意味をもちうるかどうかは、制度的な支えや組織的な訓練の場がなければ教授の力量や裁量に依存し、運用次第では TA を「大学における皿洗い」におとしめる危険性をもっているとしている [5]。履修学生に対する教育の質だけでなく、TA/SA の学習の質を高めるためにも、TA/SA の活動内容をどのように評価するかを考えていく必要がある。

#### 1.2 本学科特有の課題

本学科では、プログラミングは2年次以降の学習の基盤となり、大学における学習スタイルを確立するためにも重要な科目群として位置づけられている。そのため、学生それぞれの進捗状況や理解度へのきめ細かな対応を可能にする TA/SA の存在は欠くことができない。しかしこれまで、TA/SA をどのように育成・評価するかについて、教員間で十分に議論と意識共有が行えていなかった。これまでは、教員が TA/SA の活動状況や報告状況とレポートなどを「総合的に判断」(すなわち属人的な価値基準をもって評価)していたが、この評価方法は公平性と厳密性の観点から見直す必要があった。また、TA/SA がどのように授業支援を行っているかはレポートからある程度理解できたが、詳細な活動のプロセスや TA/SA 間の相互作用を十分に理解していたわけではなかった。

<sup>†1</sup> 日本工業大学  
Nippon Institute of Technology

## 2. 研究の目的

本研究では、TA/SA の評価について考える足がかりとして、プログラミング科目の中で TA/SA がどのように授業支援を行っているかを明らかにする。具体的には、次の 2 点を明らかにすることを研究の目的とした。

- (1) TA/SA はどのような問題に直面し、どのようにそれを解決していたか？
  - (2) TA/SA は授業支援の際にどんな工夫をしていたか？
- さらに、これを踏まえ TA/SA を評価するための評価基準としてルーブリックを作成する。ルーブリックとは、「ある課題について、できるようになってもらいたい特定の事柄を配置するための道具」と定義される[6]。

## 3. 「プログラミング技術・演習 I」の実践

### 3.1 授業概要

著者らが所属する情報工学科では、初年次の累進性の必修科目として「プログラミング技術・演習 I (春学期)・同 II (秋学期)」を設置している。使用言語は C 言語、使用するソフトウェアは Visual Studio である。学生は基本的に自分のノートパソコンを持参して授業・課題に取り組む。再履修者を含めて毎年 200 名以上の学生がいることから、複数の教員と、TA/SA によるチームティーチングによって授業を進めている。2015 年春学期は 7 名の教員（専任 6 名、非常勤 1 名）と 35 名（学部生 32 名、院生 2 名、科目履修生 1 名）の TA/SA が本科目を担当した。

ほとんどの TA/SA は本授業を受講し成績が上位だった学生で、高校時代から数えて数年のプログラミング経験のある者もあり、高いプログラミング能力と知識を持つ学生も多い。TA/SA による授業支援は 2013 年度から始まり、本年度で 3 年目となった。なお TA/SA は、本科目で授業支援に従事し、「授業支援ボランティア」という科目を同時に履修することで単位が認められる。なお、「授業支援ボランティア」は自由科目であり、卒業単位には含まれない。前年度に先輩 TA/SA からの指導を受けて履修する学生や教職を目指す学生も多く、総じて意欲的である。

## 4. 研究方法

### 4.1 調査方法

日本の TA/SA 制度をめぐる問題は、彼ら自身がこの制度の問題点を語る場や仕組みがないことにあるとの指摘がある[7]。先行研究の中には、選択式・自由回答式のアンケートや統計的な調査をもとに TA/SA の実態について調査・分析を行っている事例が見られる。しかし、TA/SA が彼らの活動について彼らの視点から自発的に語ったり記述したりしたもの、さらにそれを質的に分析したものは見つからなかった。TA/SA は時には教員よりも学生に近い存在であり、教育活動に直接的に関わることから、TA/SA が気づいたことや学んだことを彼ら自身から聞く必要があると考えた。

そこで本研究では、TA/SA に対して、活動する 5 つのクラスごとにグループインタビューを行い、彼らが授業支援をどのように行ったかについて調査することとした。できるだけ多くの TA/SA から話を聞くために授業期間内にインタビューの時間を設けた。最終的に、今回 TA/SA 登録をした 35 名中、すべてのクラス (5 クラス) の 31 名にインタビューを行った。インタビューを行う際、筆者ら (本科目の教員) のうち少なくとも 2 名がインタビュアーとして参加した。インタビューの平均時間は 61 分となった。インタビューでは、おおまかな質問内容をあらかじめ決めつつ、話の流れに従いながら柔軟に会話を進めるインタビュー方法である、半構造化インタビューを採用した。

### 4.2 分析方法

今回の調査では、TA/SA が授業支援をどのように行っていたかについて仮説を得るために、仮説検証型の質問紙調査や量的分析方法ではなく、仮説探索型のアプローチによる質的データ分析法を採用することとした。具体的な分析方法として、佐藤 (2011) が提案する「事例-コードマトリクス」という方法を援用した[8]。この手法は、事例の個別性や具体性に十分配慮しつつ、規則性を見出す際に有効とされている[8]。分析は以下の手続きをとった。

- 学生の許可を得てグループインタビューの発話内容を記録した。
- 記録された発話内容をすべて文字起こしした。
- 発話の内容を精読した上で、発言を要約し、事例集を作成した。なお、特有の言い回しが観察された場合は、それを「」で示した。
- 次に、事例集に対して焦点コーディングと呼ばれる、抽象化したコードを付すコーディングを行った。このプロセスを 3 回繰り返し全体の整合性を調整した。
- MECE (Mutually Exclusive, Collectively Exhaustive) の原則に従いながらそれぞれの事例の漏れ・重複が出ないように事例を分類し、それぞれに合うカテゴリー名を作成した。なお、この一連の作業は著者のうちすべてのインタビューに参加した著者 (大橋) が行った。
- コードの妥当性と整合性を共著者が確認した。

## 5. 結果

### 5.1 TA/SA の活動の特徴

分析の結果、20 種類の事例が抽出され、(1) 問題の特定と分析、(2) 教え方・伝え方の工夫、(3) TA/SA 間の連携・情報共有、(4) 振り返り、の 4 つのカテゴリーが生成された。事例の一覧と、各カテゴリーとの対応を表 1 に示す。事例の種類はカテゴリーごとに、(1) 問題の特定と分析 が 5 種類、(2) 教え方・伝え方の工夫 が 8 種類、(3) TA/SA 間の連携・情報共有 が 3 種類、(4) 振り返りが 4 種類であった。1 から 5 のすべてのグループで抽出された事例が 7 種類あった (表 1 の網掛け部分)。

表1 事例-コードマトリクス 各事例と5つのグループとの対応  
 Table 1 Case - code matrix: correspondence of cases to the five groups (1 to 5).

カテゴリー	事例	1	2	3	4	5
(1) 問題の 特定と分析	(1-1) 学生の傾向を把握する	○	○	○	○	○
	(1-2) 学生をつまづきや理解度を確認する	○	○	○	○	○
	(1-3) 授業の進め方を検討する	○	○	○	○	○
	(1-4) 正答や採点・評価基準を検討する	○	○	○		○
	(1-5) 技術的問題を特定する	○	○			
(2) 教え方・ 伝え方の 工夫	(2-1) より分かりやすい言葉遣いや教え方に修正する	○	○	○	○	○
	(2-2) 積極的に、満遍なく、優しく声をかける	○	○	○	○	○
	(2-3) 文法とアルゴリズムを分けて教える	○	○		○	○
	(2-4) 教える範囲に注意する、ヒントを教える、考えさせる	○	○	○		
	(2-5) 相手によって教え方を変える、調整する	○			○	○
	(2-6) よい教え方をまねる		○		○	○
	(2-7) 紙に書く、図を描く	○			○	
	(2-8) プログラミング以外のリテラシーも教える	○		○		
(3) TA/SA 間の関係・ 情報共有	(3-1) 学生の状況、対処法や教え方を共有する	○	○	○	○	○
	(3-2) 仮の担当を決めつつ協力しながら対応する	○			○	○
	(3-3) 採点基準をすり合わせる		○	○		
(4) 振り返り	(4-1) 人と話す、伝える、教える方法について振り返る	○	○	○	○	○
	(4-2) 学生とともに学ぶ		○	○	○	○
	(4-3) 知識の転化、教職や将来の仕事に役立つ			○		○
	(4-4) 「自分が分かってないと教えられない」			○	○	

(1) 「問題の特定と分析」

学生にプログラミングを教える前に問題を特定する行動が含まれるカテゴリーである。ここには、事前に学生の傾向を把握し(事例(1-1)), 学生をつまづきや理解度を確認すること(1-2)が含まれる。プログラミングの内容とは直接関係ないものの、授業を受けるために解決が必要な技術的な問題(ネットワークにつながらない、ソフトウェアがうまくインストールされないなど)を特定することも含まれる。また、授業をより良いものとするため、授業の進め方(1-3)や採点基準の修正・改善(1-4)にまで踏み込んだ内容の発言も見られた。

(2) 「教え方・伝え方の工夫」

TA/SAは試行錯誤を繰り返しながら、自分たちなりの教え方を確立していたことが伺える。すべてのグループが、より分かりやすい言葉遣いや教え方に修正する(2-1)、積極的に、満遍なく、優しく声をかける(2-2)といった工夫を図っていた。

(3) 「TA/SA間の連携・情報共有」

すべてのグループが学生の状況や学生への対処法を何らかの形で情報共有していたことが分かった(3-1)。しかし、学生が情報共有を積極的に行う機会はプレミーティングとポストミーティングを除いて少なかった。

(4) 「振り返り」

すべてのグループで、TA/SAの活動が、彼らの話し方、伝え方、教え方について考えるきっかけになったという発言が見られた(4-1)。また、プログラミングについて学び直すことにつながった((4-2), (4-4))という発言が見られた。中でも、教職課程を履修する学生や、就職活動を行っていたり意識したりしていた学生は、この活動が教職課程や将来の仕事に役立つと感じていた(4-3)。しかし、それ以外の学生からの振り返りに関する発言は少なかった。

● ルーブリックの策定

この調査結果を踏まえ、TA/SAが自己評価を行うためのルーブリックの案を作成した(表2)。作成に際しては、ステーブンス・レビ(2014)が提案する、ルーブリック作成の4段階(振り返り、リストの作成、グループ化と見出し付け、表の作成)を参考にした[6]。また、大橋・山地(2015)による、サービスマーケティングを取り入れた授業科目の中で学生を評価するために用いるルーブリックも検討材料とした[9]。今回は、表1の4つのカテゴリーを踏まえながら、「問題の特定と分析」「情報の共有と活動の記録」「教え方・伝え方の工夫」「振り返り(最終報告書)」の4つの評価領域と、それぞれ5問、計20問の質問項目を設け、5段階評価とした。

表2 TA/SA 評価のためのルーブリックの案  
Table 2 An example of a rubric for assessing performance of TA and SA.

領域	評価基準 (1項目に対して5点)	点数
A. 問題の特定と分析 (25点)	A1 学生が何に困っているか, どの箇所でつまづいているか理解しているか	
	A2 学生の問題の理由・原因が分析できているか	
	A3 対処方法が分からない, あるいは個人では対処できない場合, 他の TA/SA や担当教員に助言を求めたり, 情報収集を行ったりするなどして対応しているか	
	A4 プログラミング以外の技術的な問題がある場合, 問題を特定し対処できているか	
	A5 何らかの理由で問題を改善できない場合, 次善策を講じているか	
B. 情報の共有と活動の記録 (25点)	B1 質問とそれに対する対応策や気づいたことなどを整理し構造化できているか	
	B2 授業終了後に TA/SA や担当教員と情報共有を行っているか	
	B3 チームで教える場合, 役割分担などを話し合っているか	
	B4 毎回の活動記録を, 質問された人数, 質問内容, 回答, 反省点, 同様の問題に対する今後の対応策などの項目ごとに詳細に記述しているか	
	B5 採点の際, 間違いに対して分かりやすいコメントができているか, 学生の間違いの傾向とそれに対するコメントを記録できているか	
C. 教え方・伝え方の工夫 (25点)	C1 質問をしない学生に対しても積極的に, 満遍なく話しかけているか	
	C2 図解や例え, 分かりやすい言葉を用いるなどして教えることができているか	
	C3 相手の理解度や進度を理解し, 相手に合わせた教え方ができているか	
	C4 文法とアルゴリズムを区別しながら教えられているか	
	C5 答えをすべて教えるのではなく, 学生に考えさせるような工夫ができているか	
D. 振り返り (最終報告書) (25点)	D1 参加動機, 活動内容, 工夫した点, 学んだことなどを詳細に記述しているか	
	D2 第三者が読んで理解できるよう適切な言葉遣いで記述されているか	
	D3 多かった質問とそれに対する対応策が構造化されて記述されているか	
	D4 次学期の TA/SA が引継げるように反省点が記述されているか	
	D5 規定の分量・枚数を十分満たしているか	

## 6. おわりに

プログラミング教育を支援する TA/SA の評価基準について検討するため, 初年次プログラミング科目において, TA/SA がどのように授業支援を行っていたかを質的に調査・分析した. 今回の調査を通して, これまで知ることのできなかつた細かなエピソードを踏まえた TA/SA の活動内容を明らかにし, TA/SA の活動実態に合わせた評価基準を作成することができた. 教員・TA/SA 間で分かっている, あるいは共有できていると思っていながらできていなかったことを明らかにし, 分かりやすい形でまとめられたことは教員にとっても有意義であった. 今後, 授業の中でルーブリックを継続的に見直し, 評価を行いながら, TA/SA の評価や授業運営に役立てていきたい.

## 謝辞

本論文作成にあたってご協力いただいた皆様, 快く調査に応じてくれた TA/SA の皆様に, 謹んで感謝の意を表する.

## 参考文献

- [1] 松永公廣, 横山宏, 佐野蘭美. 認定 SA を導入したプログラミング基礎の授業実践と評価, 経営情報研究, 2011, Vol.19, No.1, pp.57-71.
- [2] 森田彦. 学生教育補助員を活用した演習教育 「プログラミング」の場合, 社会情報, 2005, Vol.14, No.2, pp.151-166.
- [3] 森田彦. SA を活用した授業運営: プログラミング演習の場合, 社会情報, 2009, Vol.18, No.2, pp.117-129.
- [4] “ティーチング・アシスタント (TA) について”. [http://www.mext.go.jp/b\\_menu/shingi/chukyo/chukyo4/003/gijiroku/07011713/001/002.htm](http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo4/003/gijiroku/07011713/001/002.htm), (参照 2016-09-01)
- [5] 荻谷剛彦. アメリカの大学・ニッポンの大学 TA・シラバス・授業評価, 玉川大学出版部, 1992.
- [6] ダネル・スティーブンス, アントニア・レビ. 大学教員のためのルーブリック評価入門, 玉川大学出版部, 2014.
- [7] 上野哲. 日本の大学院におけるティーチング・アシスタントの現状と課題, 日本教育学会大会研究発表要項, 2009, Vol.68, p.338.
- [8] 佐藤郁哉. 質的データ分析法 原理・方法・実践, 新曜社, 2011.
- [9] 大橋裕太郎, 山地秀美. サービスラーニングの手法を取り入れた大学での情報教育 「情報ボランティア」の質的分析, 情報処理学会論文誌「教育とコンピュータ」, 掲載決定.