

K-009

ソースコードのコメント文に着目した学習者の気づき抽出 Extraction of Understanding Points from Comments in Student Source Codes

大野 公裕[†]
Masahiro Ono

原田 史子[‡]
Fumiko Harada

島川 博光[‡]
Hiromitsu Shimakawa

1. はじめに

大学で行われるプログラミング教育では、学生がプログラムを作成する過程で理解につまずいた場合、Teaching Assistant(以降 TA と表記する)と教員からなる複数の指導者がマン・ツー・マンで指導を行う。指導者は、学生と問答することで理解につまずいた箇所を特定し、その箇所に対して指導を行う。このさい、教員による指導を受ける学生と TA による指導を受ける学生とでは、指導者によって指導方法が異なるため、均質な指導が保証されない。

本論文では、均質な指導を保証するため、学生の理解につまずいた箇所に応じた指導者の指導方法を統一する手法を提案する。

2. 均質指導実現のための教員の負担

2.1 演習科目における指導者

大学でのプログラミング教育では、学生にプログラミング課題を与え、学生にソースコードを作成させることで、プログラミングの実践力を養う演習科目が設けられる。このような演習科目では、プログラミング課題を解く過程で理解につまずいた学生が、理解につまずいた内容に関する質問を指導者に投げかける。この場合、指導者は質問を投げかけた学生に対しマン・ツー・マンで指導を行うため、教員のほかに、教員を補助する複数の TA が指導者として配備される。教員や TA は、プログラミング課題を解く過程で理解につまずいた学生に対し、理解につまずいている箇所を学生と問答を交わすことで特定し、その内容について指導を行う。

2.2 与える指導と気づかせる指導

演習科目における指導者は、理解につまずいた学生に対し、理解につまずいている箇所(これを「つまずき箇所」と定義する)を特定し、その内容について指導を行う。このとき学生に対して行われる指導には、つまずき箇所に対する具体的な解決策を与え、なぜその方法で解決できるのかを示すことで理解を促す指導と、つまずき箇所やつまずき箇所の解決策を与えることなく、学生にそれらを気づかせようとする指導の二つの指導方法が存在する。本論文では、前者を「与える指導」と呼び、後者を「気づかせる指導」と呼ぶこととする。与える指導は、指導者が学生に具体的な解決策を与えるため、学生の課題を解くために必要な思考錯誤の過程を省くことができる。気づかせる指導は、学生に具体的な解決策を与えないため、学生に解決策を考えさせる思考錯誤を促すことができる。学生が課題作成に思考錯誤することは、その過程から学生の知識定着を図ることや学生が学習上の気づきを得る機会を与えるなど、教育上のさまざまな利点が存在する。そのため、プログラミング指導におい

ては理解につまずいた学生に対し、気づかせる指導を行い、学生に思考錯誤を促すことが重要である。

2.3 指導の均質化とマニュアル作成

気づかせる指導を行うには、与える指導を行うよりも、指導者に指導スキルが必要とされる。そのため、指導スキルの豊富な教員は気づかせる指導を学生に行うが、教員に比べ指導スキルの劣る TA は、理解につまずいた学生に対し気づかせる指導ではなく与える指導を行うことが多い。これにより、教員は気づかせる指導を行い、TA は与える指導を行うため、個々の学生に対し行われる指導の質を保証することは難しくなる。

多くの場合、指導の質を保証する方法として、指導手順のマニュアル化が行われる。このマニュアルに従った指導を行うことで、個々の学生に対する指導の質を保証することができる。理解につまずいた学生に対する指導手順をマニュアル化するには、あらかじめ学生のつまずき箇所となりえるポイントを想定する必要がある。しかし、学生のつまずき箇所となりえるポイントをすべて想定し、それらのつまずき箇所に対応するすべての指導手順をマニュアル化することは、マニュアル作成者におおきな負担がかかる。

3. コメント文を活用した指導指標の提示

3.1 指導手順指標の提示による指導の均質化

本研究では、過去に学生が提出したソースコードのコメント文を計算機によって解析し、学生のつまずき箇所に応じた適切な指導手順を TA に示すことで、指導手順のマニュアルを教員が人力で作成することなく指導の均質化を試みる。本手法の全体像を図 1 に示す。本論文では、あるプログラミング課題で学生が理解につまずいた場合、そのつまずき箇所を TA が学生との問答により特定したのち、そのつまずき箇所に対応する指導手順を、

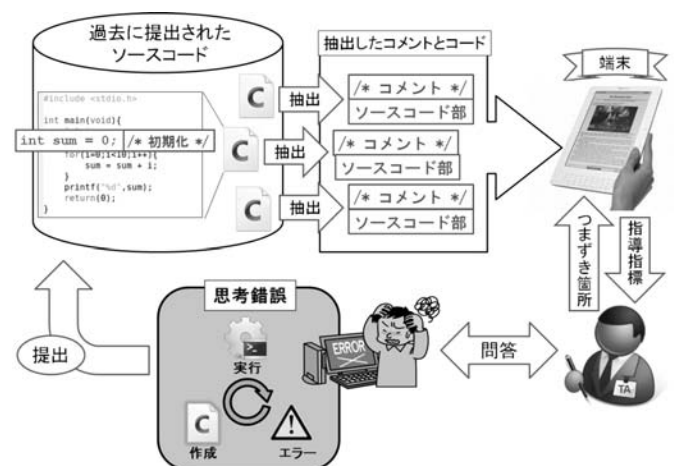


図 1: 手法の全体像

[†]立命館大学大学院 理工学研究科

[‡]立命館大学 情報理工学部

過去に同じプログラミング課題に取り組み、提出された学生のソースコードのコメント文から取捨選択し、そのコメント文を気づかせる指導手順の指標として表示する手法を提案する。

3.2 つまづき箇所に記述されるコメント文

プログラミング初心者が作成した多数のソースコードを著者らが調べたところ、ソースコード中に記述されるコメント文は、プログラムの理解を助ける目的で記述されるコメント文と、プログラム作成過程で学生が何らかの気づきを得た場合に、その気づきの内容が記述されたコメント文が存在する。前者のコメント文は、プログラム作成者の主観でプログラム内容を理解しにくい場合に記述されるという特徴があり、後者のコメント文は、教員がプログラミング課題を通して学生に理解してほしいと考える課題の要点に学生が気づいたことを示しているという特徴がある。これらの特徴は、学生が理解につまずく原因となりえることから、いずれのコメント文も学生が理解につまずく原因となりえる処理をあらわすソースコード部分に記述されると考えられる。また、その理解につまずく原因となりえる処理をあらわすソースコード部分の理解を助けるものがコメント文であるため、コメント文にはソースコード部分の処理を連想できる内容が記述される。そのため、本手法では、過去に学生が作成した課題の解答となるソースコードから、コメント文とそのコメント文が指し示すソースコード部分を抽出する。抽出したコメント文は、TA に提示する気づかせる指導の指標として用い、コメント文の指し示すソースコード部分は、TA が学生をつまづき箇所を指定する場合の候補として用いる。すなわち、システムは抽出したソースコード部分を学生をつまづき箇所の候補となりえる箇所として TA に表示するため、TA は学生と問答により学生をつまづき箇所を特定したのち、システムに表示されたつまづき箇所に該当するソースコード部分を指定することで、そのつまづき箇所のコメント文を気づかせる指導の指標として閲覧することができる。

3.3 コメント文の内容に応じた指導指標の選別

気づかせる指導を行うには、つまづき箇所を解決する具体的な解決策を示さないことが重要である。しかし、学生がソースコードに記述したコメント文には、そのコメント文が指し示すソースコード部分の内容を詳しく記述したものも含まれる。このようなソースコードの内容を詳しく説明するコメント文は、つまづき箇所を解決する具体的な方法を記述したものであり、気づかせる指導ではなく与える指導の指標となるコメント文である。そこで本手法では、学生がソースコードに記述したコメント文と与える指導の指標であるか、気づかせる指導の指標であるかを判断し、それらを分類したのち TA に提示する。

与える指導の指標と気づかせる指導の指標を判断する基準の一例として

- コメント文の長さ
- コメントにソースコード一部が含まれているか

の 2 点があげられる。

たとえば、図 2 において「初期化」というコメント文は、他のコメント文「0 を代入」や「sum に 0 を代入」と

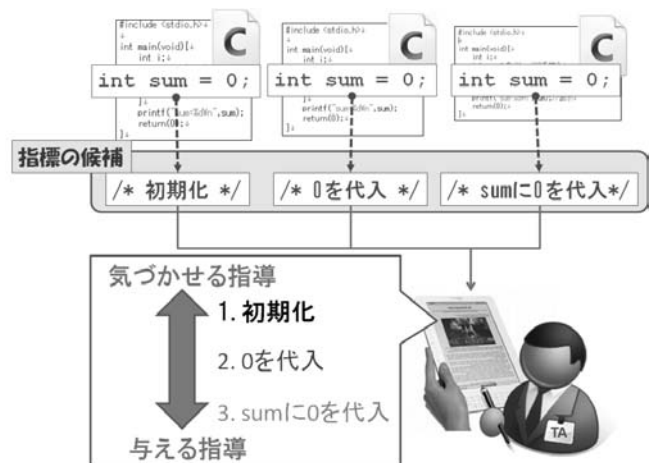


図 2: 指標の選別

比較し、コメント文の長さが短く、コメント文中にソースコードの一部が含まれていないため、気づかせる指導の指標となるコメント文である。これに対し、「sum に 0 を代入」というコメント文は、「sum」や「0」といったソースコード中に出現する変数名やその具体的な値を含み、

```
int sum = 0;
```

のソースコードを詳しく説明するコメント文であるため、与える指導の指標となるコメント文である。

このようにして、本手法ではコメント文の長さやコメント文中のソースコードの一部の有無から、与える指導の指標であるか、気づかせる指導の指標であるかを判断する。

4. 関連研究

複数の指導者が指導を行う場合に生じる指導の質の異なりを均質化する研究として、指導者間で指導すべき項目を共有する研究 [1] が挙げられる。この研究は、教員が教えるべき項目を形式化することで、TA が指導すべき項目を教え漏れることを防ぐことが可能である。しかし、それらの項目について指導するさいの具体的な指導手順は TA の裁量に任されるため、TA に気づかせる指導を期待することは難しい。本手法は、学生をつまづき箇所に対する具体的な指導手順を示すため、TA に気づかせる指導を期待できる。また、学生の過去に提出したソースコードを解析するため、教員の負担も少ない。

5. おわりに

本論文では、指導者の指導方法に依存しない均質な指導を実現するため、指導者の指導方法を統一化する手法を提案した。今後は過去に提出された学生のソースコードに本手法を適用し、手法の有用性を検証する。

参考文献

- [1] 大野公裕: "講義資料からの意味ネットワーク作成による学習事項の形式化", 電子情報通信学会/情報処理学会 情報科学技術レターズ, Vol.9, No.3, pp.115-118, Sep., 2010.