

プログラミング学習のための ソースコード読解問題類題生成システムの実現

吉田拓己,立岩佑一郎,山本大介,高橋直久

名古屋工業大学大学院 工学研究科 情報工学専攻

1. はじめに

本稿では C 言語のソースコードを読んでそれに対する問題を解く形式の読解問題の類題生成システムを提案する。読解形式の問題は主に文法やアルゴリズムの理解の確認に用いられる。文法を学ぶにはその文法が実際に使われているプログラムの問題（例：for 文の中身が何回実行されるか）、アルゴリズムを学ぶにはプログラム中の任意の変数の値を問うなどを問う問題（例：ソートの途中結果）を出題し動作を理解しているか確認する必要がある。

また、読解形式の問題は記述形式の問題よりも比較的容易なため、繰り返し学習に適している。しかし全く同じ読解問題を繰り返し解いてもあまり効果的ではなく、それを解決するために多数の問題と正解例を作成するのはとても手間がかかる。これらの問題を解決するため、以下の要件を満たす類題生成システムを実現することを目標としている。

【要件 1】 プログラムの内容を深く理解しているか確認するため、変数の値の変化やプログラムの動作の追跡を伴うような読解問題を生成する。

【要件 2】 演習効果を高めるため、類似しているが少し異なるような多数なプログラムの読解問題を生成する。

【要件 3】 課題取得、答案提出、答案評価の QA サイクルからなる繰り返し学習を可能にするため、答案の自動評価を実現する。

この要件を満たすようタグにより出題可能な読解問題類題生成システム⁽¹⁾を実現した。プログラムの書き換えたい箇所や出題したい設問の種類などの情報を含んだ出題方針をタグとしてプログラムのソースコードに記述することで問題プログラムの書き換えや、その正解例を自動で生成することで答案の自動評価を可能にした。しかしこのシステムには以下の問題がある。

【問題 1】 どのタグをどの個所に付けられるか把握しないといけない。

タグにはプログラムの書き換えを行うものや、設問を出題するものがあるがプログラムのどの個所に書くことができるのか把握していないと誤った箇所に付けてしまう可能性がある。誤った箇所にタグ付けを行ってしまうと類題が正しく作成されない。

【問題 2】 問題として不適切な類題が生成される可能性があるが、それを取り除くことができない

正しく類題が作成されたとしても問題として出題するには不適切なものが存在する場合がある。

例えば for 文の繰り返し回数が 5 から 10 の間で書き変わるプログラムに対し、繰り返し回数が 7 の時のある変数の値を問う設問がある場合について考える。この場合、繰り返し回数が 7 から 10 の時は適切な問題となるが、繰り返し回数が 5 か 6 の時は不適切な問題となる。問題の生成は受講者が問題を要求した時点のためこのシステムでは不適切な問題を取り除くことができない。以上の問題から問題追加者に余計な手間をかけてしまうことになり負担軽減の目的を達成することができない。

2. 提案システムの特徴

これらの問題を解決するために本稿ではソースコード読解問題類題生成システムを提案する。

【特徴 1】 類題設定ファイル作成機能

類題設定ファイルとは、ソースコードの中のどの個所を書き換え、どの設問を出題するかといった類題を作成するのに必要な情報（**出題方針**）を書き込んだファイルである。システムがソースコードを解析し、出題方針として設定可能な箇所を提示する。問題追加者はシステムによって提示された箇所を選択し内容を入力することで、その内容に応じたタグ付けを行った類題設定ファイルをシステムが自動で作成する。これにより問題 1 を解決する。

【特徴 2】 類題生成機能

特徴 1 によって作成された類題設定ファイルの内容を解析し類題を生成する。類題は設問、ソースコード、正解例によって構成されており、類題の幅は設問の種類と変数の初期値の組み合わせによって広げられる。類題を用いることで、読解問題の誤答者に正解などの解説を見せた後でも次に解くときには別の類題にすることができる。これにより正解が分かっている場合に問題を見なくても解答できてしまう問題を解決することができる。生成された類題の書き換え内容、正解例、出力結果の一覧を確認できるようにする。このとき正しい正解例が作成されていないなど、問題として不適切なものが生成されていたらそれを削除することができる。特徴 2 により問題 2 を解決する。

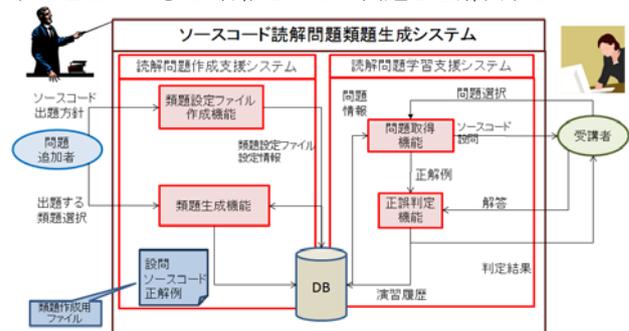


図 1: 提案システム全体図

Implementation of a System for Generating Similar Problems of Reading From a Program For Programming Learning

Takumi Yoshida, Yuichiro Tateiwa,
Daisuke Yamamoto, Naohisa Takahashi
Nagoya Institute of Technology

[特徴 3] 正誤判定機能

特徴 2 によって生成された正解例と受講者の入力を比較することで答案の自動評価を行う。

3. 提案システムの概要

本システムは問題追加者向けの読解問題作成支援システムと受講者向けの読解問題学習支援システムの2つにより構成されている。問題追加者は読解問題作成支援システムを用いて類題設定ファイルを作成し、それにより生成された類題から受講者に問題を出すものを選ぶ。受講者が問題を選択するとシステムがその問題の類題をランダムに取得し問題を作成する。受講者が入力した解答は正誤判定機能で正しいか判定され、その結果を受講者に表示する。また正解率などの演習履歴を DB に格納し、問題取得機能で参照することで受講者の達成度に応じた類題を出すことも可能となっている。

4. 提案システムの実現法

4.1 類題設定ファイル作成機能の実現法

静的解析機能、設定箇所提示機能、出題内容取得機能、タグ付け機能を提供することで類題設定ファイル作成機能を実現する。

[STEP. 1] 汎用的なプログラム静的解析ツールを用い、プログラムに含まれる関数定義情報、関数参照情報、変数定義情報、制御構造情報を取得する。取得した情報にはそれぞれ名前、出現行といった情報が含まれるのでそれぞれに ID を割り振る。出題箇所と内容はこの ID で管理する。

[STEP. 2] この解析結果を用いてタグを付けることが可能な箇所を抜き出し、問題追加者に提示する。例えば print が含まれる箇所は出力を問う問題が出題され、変数が定義されている箇所は初期値の書き換えが可能であることを表示する。

[STEP. 3] 問題追加者は表示された内容から出題したい箇所を選択し、どのような設問、書き換えを行いたいといった問題内容に関わるものや、類題生成数、類題設定ファイルの名前といった問題情報を入力する。

[STEP. 4] システムはその内容に合うタグ付けをソースコードに行い類題設定ファイルを生成する。

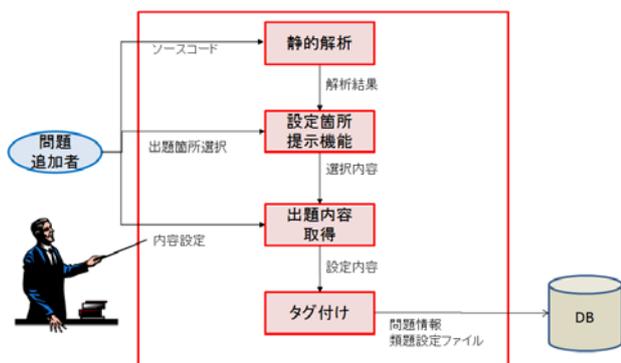


図 2: 類題設定ファイル作成機能

4.2 類題生成機能の実現法

タグ解析機能、プログラムの書き換え機能、設問作成機能、表示用ソースコード作成機能、正解例作成機能、類題確認機能を提供することで類題生成機能を実現する。

[STEP. 1] 類題設定ファイルのタグを解析し、プログラムの書き換える箇所と内容、設問の出題箇所と内容を取得

する。これをもとにプログラムの書き換えを行う。

[STEP. 2] 類題設定ファイルの問題情報をもとに各種類の設問の出題数を取得する。出題数が 1 以上の種類の設問に対し、類題を作る数だけタグ解析機能、設問作成機能、表示用プログラム作成機能、正解例作成機能を用いて類題を生成する。

[STEP. 3] 生成された類題は全て別のファイルとして登録され、最大数が類題生成数の ID で管理される。この ID で設問、表示用ソースコード、正解例を管理する。

[STEP. 4] 生成された類題の中から実際に受講者に表示する問題を選択する。

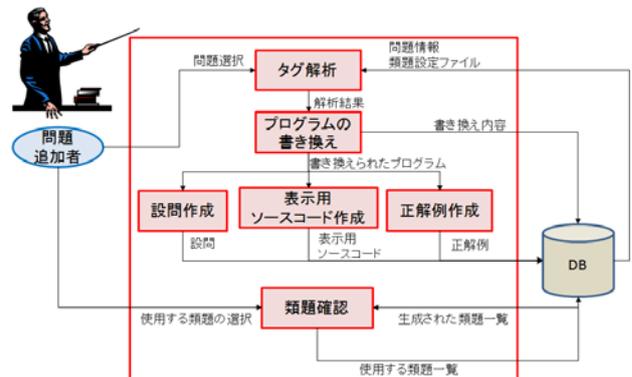


図 3: 類題生成機能

4.3 正誤判定機能の実現方法

正解が一意に決まるものを読解問題として扱っているため、文字列の完全一致で正誤判定ができる。

例として出力結果を問う問題の正誤判定をあげる。出力を問う問題は printf で出力される内容を問うもののため、

```
n=5 のときの printf("出力:%d", n);
```

の出力結果は、「出力:5」以外にはなりえない。このことから正誤判定をするのに必要とされる文字列の意味を解釈せずに文字列一致のみで正誤判定を行うことができる。

[STEP. 1] 問題取得機能で得た正解例ファイル中の文字列を取得する。

[STEP. 2] 受講者が入力した文字列を取得する。

[STEP. 3] 取得したそれぞれの文字列を比較できるように変換する。

[STEP. 4] それぞれの文字列を比較し、判定結果を受講者に返す。

5. まとめと今後の課題

本研究では読解問題の類題を少ない手間でお題するために、問題作成の支援を行うシステムを提案した。しかし本システムはインターフェースが直感的でないため、使い方の説明をしないと問題を作成できない可能性がある。

参考文献

(1) 吉田拓己, 立岩佑一郎, 山本大介, 高橋直久, “タグにより出題方針指定可能な読解問題類題生成システムの実現”, 平成 22 年度電気関係学会東海支部連合大会, 教育支援 F3-4 (2010).