

プログラミング学習支援を目的としたオンラインジャッジシステム IOJ の提案

森本 祥之

Shoji MORIMOTO

1 はじめに

現代社会において、ソフトウェアの需要が増しており、需要に伴い、広い分野でプログラミングが行われている。若手技術者の育成を目的として、教育現場でもプログラミングを扱う授業が行われている。しかし、プログラミング教育において、学生が基礎的なプログラミング能力を身につけているのかといった疑問が挙げられてる¹⁾。大学を含む教育機関において、教員と学生の人数比を合わせるといった人的リソースをプログラミング教育にかけるとは困難である。そこで、ソフトウェアでプログラムの正しさを判断するオンラインジャッジシステム (OJS) に着目した。OJS はあらかじめ正解となるデータを用意することで、学生の数に関わらず、学生のプログラムが正しいかどうかを判断することが可能である。また、OJS ではプログラミング能力のレベルに合わせた問題が複数用意されており、問題を解いた人数や不正解数などから問題の難易度の推定が可能である。実際に OJS をプログラミング演習に導入している教育機関も存在する。そこで本研究では、プログラミング能力のレベルに応じて適切な問題をサジェストする INTELLIGENT ONLINE JUDGE system (IOJ) を提案する。

2 オンラインジャッジシステム

2.1 オンラインジャッジシステムとは

ONLINE JUDGE SYSTEM(OJS) はユーザから提出されたプログラムをコンパイル及び実行することにより、提出されたプログラムが正しい挙動をするか否かをユーザへフィードバックするシステムである． Fig.1 に OJS の 1 つである AIZU ONLINE JUDGE system の評価結果画面を示す． ソースコードの評価結果には正しい出力が得られた場合 (Accepted), 出力が正しくない場合 (Wrong Error), コンパイル時エラー (Compile Error), 実行時エラー (Runtime Error), 定められた処理時間を超えた場合 (Time Limit), 定められたメモリ消費量が多い場合 (Memory Limit Error), 出力の形式が間違っている場合 (Presentation Error) などがある． 一部の OJS では、提出されたソースコードや評価結果などの情報が記録されており、誰でも閲覧することが可能である．

2.2 OJS の活用

OJS はプログラミング学習のための問題集形式で，問題の入力データと出力データの組を用意することで，無償でシステムの利用が可能である．実際に大学の教育機関に導入しており，学習効果の向上と教育コスト削減に繋がったと報告されている²⁾．一方で，学習者の問題選択によっ


[PAGETOP](#)


[PROBLEM](#)


[RANKING](#)


[STATUS](#)


[CONTEST](#)


[SOURCE](#)


[Login](#)


[Register/Signup](#)

Recent Activity

Author	Problem	Status	Lang	Time
nl250053	ALDS1_3_C	✓ AC	C	2018-07-17 19:15
pnz99	ITP1_4_A	✓ AC	C++	2018-07-17 19:14
HappyTree3966	ALDS1_1_A	✗ WA	Python	2018-07-17 19:12
HappyTree3966	ALDS1_1_A	✗ RE	Python	2018-07-17 19:12
njl18064	ITP1_10_B	✓ AC	C	2018-07-17 19:11
HappyTree3966	ALDS1_1_A	✗ RE	Python	2018-07-17 19:11
njl8117	ITP1_5_C	✓ AC	C	2018-07-17 19:11
HappyTree3966	ALDS1_1_A	✗ RE	Python	2018-07-17 19:11
njl18055	ITP1_2_D	✓ AC	C	2018-07-17 19:09
HappyTree3966	ALDS1_1_A	✗ RE	Python	2018-07-17 19:09
HappyTree3966	ALDS1_1_A	✗ RE	Python	2018-07-17 19:08
HappyTree3966	ALDS1_1_A	✗ RE	Python	2018-07-17 19:08
HappyTree3966	ALDS1_1_A	✗ RE	Python	2018-07-17 19:08
HappyTree3966	ALDS1_1_A	✗ RE	Python	2018-07-17 19:08
macha2	ITP2_1_A	✓ AC	C++	2018-07-17 19:08
njl18064	ITP1_10_B	✗ WA	C	2018-07-17 19:07

Announcements and Bug reports

“ Welcome to AOJ!

[Link to β version of AOJ 2.0](#)

Recent messages on the web board

- 1053 2018-07-17 11:06 bsRnJK
- ITP1_1_A 2018-07-15 18:46 bonobono
- 1629 2018-07-14 18:00 admin1
- 0166 2018-07-17 18:39 faced
- ITP1_4_D 2018-06-26 21:19 SHAHRIER_1704010
- 0005 2018-06-26 20:13 aklyamaNAO
- ITP1_1_A 2018-06-25 15:36 squeal001
- ITP2_3_C 2018-06-24 11:26 ba4u4
- 2090 2018-06-21 11:27 bsRnJK
- 1167 2018-06-19 11:27 bsRnJK

Fig.1 AIZU ONLINE JUDGE

てプログラミング能力の向上に影響を与える可能性が報告されている³⁾。したがって、学習者のレベルに応じて適切な問題をサジェストする INTELLIGENT ONLINE JUDGE system(IOJ) を提案する。

3 インテリジェントオンラインジャッジシステム

3.1 システムの実働における想定環境

INTELLIGENT ONLINE JUDGE system(IOJ) は学習者のレベルに応じて適切な問題をサジェストするオンラインジャッジシステムである。IOJ の実働は大学のプログラミング授業を想定しており、予習、演習、復習を包括的に支援することで、学生のプログラミング学習の効率化を目標とする。IOJ は授業時間内と授業時間外の 2 つの場面での利用を想定しており、学生と教員のサポートを行う。学生は授業時間外に授業の予習として IOJ を使用し、複数の問題を解く場合には IOJ が学習者に対して問題のサジェストを行う。学生が予習で解いた問題の進捗状況については、教員が確認出来るよう解いた問題の番号と回答結果をシステムが自動的に収集する。教員は予習段階の学生の進捗状況を把握することにより、授業で実施する演習問題を適切に選択することが可能である。

3.2 システムの流れ

Fig.2 に IOJ のシステムの流れを示す．はじめに，IOJ で使用する問題はあらかじめ教員が複数の問題を用意し，それぞれの問題に対して教員がタグ付けを行う．例えば，ある問題に対するプログラムの中に for 文や if 文が使用されているならば，その問題に for と if のタグ付けを行う．他には，構造物やポインタなどの道具に関してもタグ付けを行う．タグ付けを行い，学習者が問題を選択する際は，

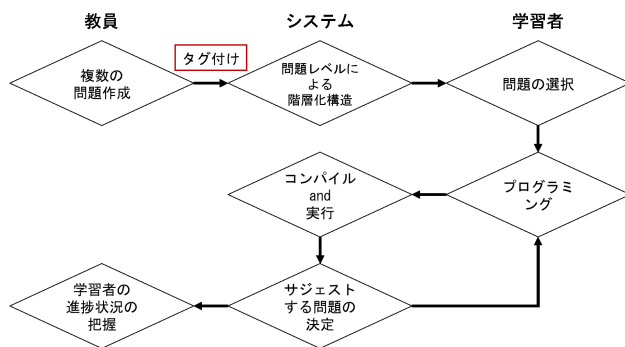


Fig.2 システムの流れにおけるフローチャート

for 文や if 文などの形式的な問題が構造体やポインタなどの道具的な問題かを分類しておき、選択できるようにする。また、複数用意した問題をタグごとに分類したあと、その中で初級問題から上級問題までを分け、階層的に問題ごとのひも付けを行う。Fig.3に階層構造の一例を示す。次に、学習者は複数ある問題の中から1つの問題を選択し、プログラムを書く。完成したプログラムをシステムに提出し、正解であれば、階層構造の1つ上の問題をサジェストし、不正解であれば1つ下の問題をサジェストする。そして、再び問題に挑戦し、正解であれば1つ上に進み、不正解であれば1つ下に下がるということを繰り返して、最終的には階層の頂点にある問題を解くことを目的とする。また、学習者が不正解のループに入っている場合には、システムが教員に対して自動的にアラートを出すことで、教員は演習時間内にサポートすることが可能である。

4 予備実験に向けた課題設定

IOJを構築するにあたり、2点の課題が存在する。1つ目の課題は用意した問題の難易度が分からないという点である。教員が複数の問題を用意する際に、問題が初級レベルか上級レベルかの判定を行い、階層的に並べる必要がある。そこで、タグの量と模範解答のコード行数のそれぞれに重み付けを行い、問題レベルの推定を行う。タグの量と模範解答のコード行数に対する重みの配分は、先行研究において推定された難易度に一致するよう設定する。なお、先行研究では、項目応答理論と系列パターンマイニングを用いて問題レベルの推定を行っている⁴⁾。したがって、重み付けにより推定した問題の難易度により、問題を階層的に並べることとする。また、初級問題は上級問題に必要な技術を包括していることとする。2つ目の課題は学習者に対してシステムが問題のサジェストを行う際に、学習者は次に解きたいと思う問題の傾向が不明確であるという点である。学習者は問題が不正解だった場合に再度同じ問題を解きたいのか、難易度の下がった問題を解きたいのかを明確にする必要がある。また、次に解く問題によって、テストを受けた時に学習効果がどのように変化するかについても検証する必要がある。したがって、問題が不正解だった場合に、同じ問題を再度解くか次の問題に移るかを選択してもらい、学習者が次に求める問題の傾向および正答率

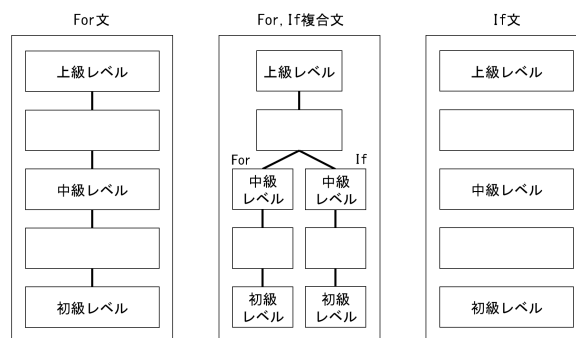


Fig.3 サジェストするための階層構造

を調査する。

5 まとめ

学習者が問題を選択する際に、適切な問題を選択できず、プログラミング能力の向上に影響を与えているという課題があった。そこで、プログラミング演習にOJSを導入し、進捗状況の把握と適切な問題のサジェストを行うシステムIOJの提案を行った。今後は問題の初級レベルと上級レベルの区別を行う予備実験や学習者が求める問題の傾向を掴むための予備実験を行う予定である。また、予備実験の結果を踏まえて、適切な問題のサジェストを行うシステムを構築する。そして、同志社大学のプログラミング授業での導入を目指し、問題のサジェストを行うことで、学習効率の向上が見られることを検証する予定である。現段階で想定している学生に提供する課題は、教員に作問してもらい、提供するという方法である。しかし、教員による作問は教員にとって大きな負荷となる。したがって、今後は学生同士の作問を考えており、学生の作問を利用した学習による学習効果の分析も行おうと考えている。

参考文献

- 1) 高桑稔, 西口大亮, 北英彦, プログラミング能力向上を目的としたプログラムテストの学習環境に関する研究, CIEC, pp.27-30, 2013.
- 2) 古谷勇樹, 長尾和彦, 峯脇さやか, OJSを用いたプログラミング学習支援環境の構築と評価, 信学技報, Vol.115, No.492, pp.45-50, Mar 2016.
- 3) 田口浩, 糸賀祐弥, 毛利公一, 個々の学習者の理解状況と学習意欲に合わせたプログラミング教育支援, 情報処理学会論文誌, Vol.48, No.2, pp.958-968, Feb 2007.
- 4) 則行祐作, 中川尊雄, 畑秀明オンラインジャッジの履歴を対象としたプログラムの成長分析, 信学技報, Vol.116, No.277, pp.97-101, Oct 2013.