

プログラム空欄補充問題を用いたオンライン学習システムの運用と評価

Operation and Evaluation for Online Learning System using
Fill-in-the-blanks-type Questions of Source Programs野上 裕二[†] 鈴木孝幸[†] 納富 一宏[†]
Yuji Nogami Takayuki Suzuki Kazuhiro Notomi

1. はじめに

大学の学部等で行われているプログラミング教育では、プログラミング言語を中心とした講義とノート PC などのコンピュータ機器を活用した実習とを交互に行う形式が一般的である。入門者を対象とした場合、アルゴリズムの理解やプログラミング言語の文法の理解が必要不可欠である。またさらに、中級以上のプログラミング学習者においては、ソースプログラムのコーディング方法や具体的なコーディング・スタイルのバリエーションを学ぶことが実践的なプログラミングをマスターする上では特に重要となる。多くのサンプルプログラムを丁寧に読解したり、コメントを含めたプログラムの記述の仕方を真似たりすることが初期の学習の基本となると考えられる。筆者らは、コメント付きのソースプログラムから学習用の空欄補充問題を自動生成することに関して、具体的な検討とオンライン学習システムの開発を継続的に進めている。

本稿では、筆者らがこれまで行ってきた、初学者を対象としたプログラミング学習における空欄補充問題の自動生成手法を示すと共に、現在開発している利用者が個別に学習できるオンラインシステムについて紹介する。また、本システムを運用について述べ、利用者による評価結果について考察する。

2. 空欄補充問題を用いたオンライン学習システム

2.1 自動生成手法

任意のソースコードから自動採点可能なプログラミング空欄補充問題を自動生成するためには、ソースコードに含まれる、①関数定義、②変数宣言、③制御構造などをキーとして出題対象を決める。そして、そのキーを基にソースコードからブランクとフィラーを作成する。ここで、ブランクは出題時に設問となるデータ、フィラーは解答となるデータとなる。これらをペアにして、データベースに保存する。一連の流れを示したものを図 1 に示す。

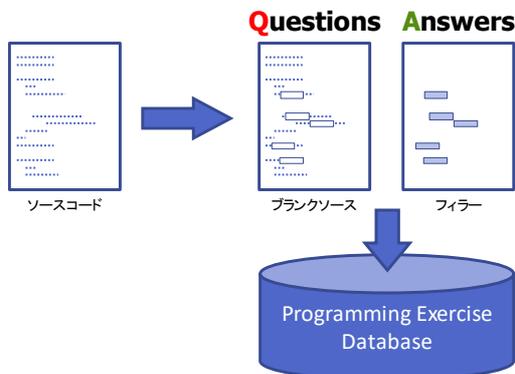


図 1 問題自動生成の流れ

2.2 実装システム

システムはブラウザ上で動作するオンラインシステムであり、ユーザは問題となるソースコードを選択後、表示される問題文中の解答欄に妥当と思われる内容（プログラムの一部となる文字列）を問題形式に沿って記入もしくは選択肢から選択し、回答の送信ボタンをクリックすることで回答がサーバへ送信される。システムの解答画面を図 2 に示す。また、問題は生成する度に更新される。例を図 3 に示す。

Dig1 [BETA] ver.0.2.8

C言語の演習問題を実施するシステムです。

操作概要:

1. 下記に表示されている選択肢から演習問題を選んでください。
2. 問題が表示されるので、空欄部分 (a), (b), (c) ... に正しい答えを入力または選択してください。
3. 空欄部分にマウスカーソルを合わせる (ホバーする) とヒントが表示されます。
4. 問題内容はランダムに変化します。
5. 「回答の送信」をクリックすると自動的に採点され、記録が保存されます。
6. 「履歴一覧」をクリックするとこれまでの記録が表示されます。
7. 何度でもチャレンジすることができます。
8. 終了する場合はこのタブを閉じてください。

是非、アンケートにご協力ください: [アンケートに回答する](#)

```

src004.c  この問題をskip
001: #include <stdio.h>
002:
003: int main(void) {
004:     int num;
005:
006:     do {
007:         (a) ("0 から 9 の数を入力してください--> ");
008:         (b) ("%d", &num);
009:     } (c) (num >= 0 && num <= 9);
010:
011:     (d) ("終わります. \n");
012:
013:     return 0;
014: }
回答の送信  チャレンジ  履歴一覧
Copyright(C) 2018 by NOTOMI & SUZUKI Lab., Kanagawa Institute of Technology
  
```

図 2 解答画面の外観

```

src004.c  src004.c
1: #include <stdio.h>
2:
3: int main(void) {
4:     int num;
5:
6:     (a) {
7:         printf("0 から 9 の
8:         scanf("%d", &num);
9:     } (b) (num >= 0 &&
0:
1:     printf("終わります. \n
2:
3:     (c) 0;
4: }
回答の送信  チャレンジ

1: #include <stdio.h>
2:
3: (a) main(void) {
4:     int num;
5:
6:     do {
7:         (b) ("0 から
8:         (c) ("%d", &
9:     } while (num >= 0 && n
0:
1:     (d) ("終わります.
2:
3:     (e) 0;
4: }
回答の送信  チャレンジ
  
```

図 3 問題の変更

3. 実験

本システムが妥当なものであるかを判断するために利用者による評価実験を行った。実験協力者は本学情報工学科 C 言語基礎ユニット受講者 83 名、当該科目の授業中にシステムを利用して問題を解いてもらった。また、システムの利用後にアンケート調査を行った。アンケートの集計結果を図 4-図 7 に示す。

今回の「演習クイズ提示システム」は、何回繰り返してみましたか？

83 件の回答

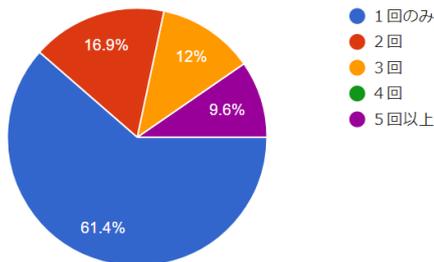


図 4 システムで問題を解いた回数

空欄に入力する項目は分かりやすかったですか？

83 件の回答

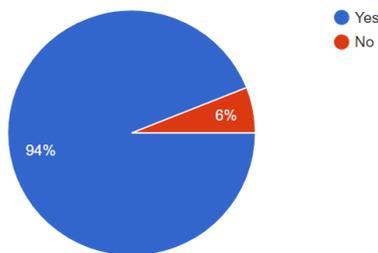


図 5 空欄入力の分かりやすさ

空欄に入力しないとならない項目は分かりやすかったですか？

83 件の回答

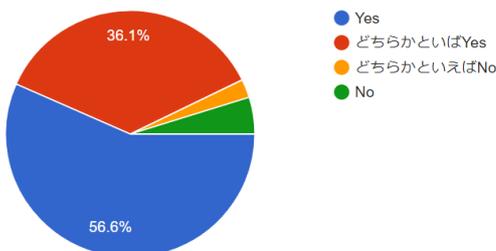


図 6 入力しないとならない項目の分かりやすさ

この形態で演習の一部として、今後使ってみてみたいと思いませんか？

83 件の回答

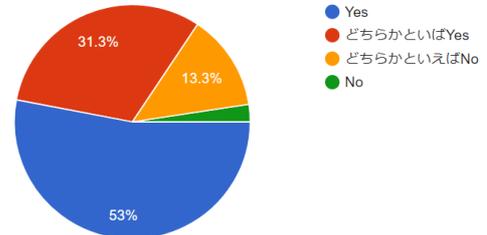


図 7 システムを今後も使ってみてみたいか

アンケートから読み取れることについての所見を記す。図 4 から、2 回以上の挑戦が 4 割弱であり、繰り返し挑戦しやすい仕組みの土台は出来上がっていると考える。繰り返し挑戦をしやすくなるよう、繰り返し挑戦する目的を難易度の追加や仕組みの導入をしたい考えられる。

図 5 から、9 割以上の肯定的な意見が得られている。そのためこの方式での出題は受け入れられており、機能的に問題はないと考えられる。

図 6 から、おおむね肯定的な意見が得られた。インターフェースの調整を行い、さらにユーザにわかりやすくする。

図 7 から、8 割以上の肯定的な意見がえられた。

また、自由記述欄には、予習復習に活用したい、空欄があっても確認無しで送信されるのは操作ミスが誘発される、解答方式単一化という意見があった。

4. まとめ

アンケート結果から、ユーザはシステムの利用に肯定的であることが分かった。今後は、アンケートの自由記述欄に寄せられた意見を参考に機能改善を行う。

本研究では、プログラム空欄補充問題を用いたオンライン学習システムの運用と評価について述べた。現在、授業において運用中であり、データの収集を行っている。今後、分析を行う予定である。

参考文献

- [1] 有安浩平, 他, “学習者に合わせた C 言語演習穴埋め問題の自動生成,” 第 7 回日本データベース学会年次大会 DEIM Forum 2009 D9-5, 2009.
- [2] 野上裕二, 納富一宏, “プログラミング学習支援における問題自動生成に関する基礎的検討,” 情報処理学会 第 16 回情報科学技術フォーラム(FIT2017)講演論文集, 第 3 分冊, K-022, pp.465-466 (Sept. 2017)
- [3] 野上裕二, 納富一宏, 鈴木孝幸, 西村広光, 示野浩士, “プログラム空欄補充問題を用いたオンライン学習システムの開発と評価”, 電子情報通信学会技術研究報告, vol. 117, no. 335, pp.39-43(2017)