N-026

データ構造に関する理解度把握のための動きある図式を用いたテスト手法
A Test Method to Grasp Understanding on Data Structure Using Active Diagrams

稲葉 大祐 † 田口 浩 ‡ 原田 史子 † 島川 博光 †
Daisuke Inaba Hiroshi Taguchi Fumiko Harada Hiromitsu Shimakawa

1. はじめに

多くの大学において，データ構造やアルゴリズムを説明するさい，教員は図式を用いる．図式を用いてアルゴリズムが実際に動作する様子を可視化することは，学習者の理解度向上のための有効な手段である [1].

これまでにも，データ構造やアルゴリズムの学習支援のための研究は，多く行われている．文献 [2] では，プログラムが実行される手順を，ソースコードと図式を対応して表示する手法が述べられている．学習者に，ソースコードとアルゴリズムとの対応付けを理解させることは可能であるが，学習者がそのアルゴリズムを理解できているかどうかを認識することは困難である．

そこで本論文では，データ構造やアルゴリズムの理解度を把握することを目的としたテスト手法を提案する．本手法で対象とするアルゴリズムは，リストやスタックなどのデータ構造独自のアルゴリズムである．これらのアルゴリズムについて，図式を用いてテストを行い，理解度を把握する．

2. C 言語プログラミング教育

2.1 プログラミングにおける学習内容の分類

学習者は，プログラミングを習得するために多くを理解しなければならない．その内容は，以下の 3 つに分類される．

- プログラミング言語に依存する概念
- リストやスタックなどのデータ構造独自のアルゴリズム
- サーチやソートなどのデータ構造に依存しないアルゴリズム

ソースコードを作成するさい，学習者は上記の内容を同時に複数考える必要がある．

2.2 プログラミング教育の現状

C 言語プログラミング教育では一般的に，学習者は C 言語の文法を学習した後，C 言語でデータ構造やアルゴリズムを実現する方法を学習する．演習において，学習者は，データ構造とアルゴリズムを理解しながらソースコードを作成し，教員は，学習者のソースコードを採点する．採点のさい，教員がソースコードから判断できる情報には限界がある．たとえば，演習において，ある問題ではソースコードが正確に書けているが，同じ分野の問題で出題形式を変えた場合，急にソースコードが書けなくなるということが挙げられる．この場合，ソースコードが書けない原因がソースコードの書き方とデータ構造とアルゴリズムの仕組みのどちらにあるか特定する必要がある．原因が分からない状態では，教員は的確な指導を行えない．

3. データ構造に関する理解度把握

3.1 動きある図式を用いたテスト

本論文では，データ構造独自のアルゴリズムの理解度を把握するためのテスト手法を提案する．その理解度を把握することにより，ソースコードの書き方とデータ構造独自のアルゴリズムのどちらが理解できていないのかを明確にできる．本手法では，C 言語でデータ構造やアルゴリズムを実現する方法を学んでいる学習者を対象とし，プログラミング言語に依存しない箇所の理解度を判断するためのテストを行う．

本手法の概要を図 1 に示す．テストにおいて出題される問題は,「リストに要素を追加する過程を示せ」といった，図式を操作することによって解答する問題である．テストを実施するさい，まず学習者は，図式を任意に操作できる答案作成ツール上で与えられた図式を操作して問題を解く．本論文では，答案作成ツール上で操作される図式を，動きある図式と呼ぶ．学習者が答案を解き終えると，動きある図式を操作した過程が答案として教員に提出される．そして，教員は提出された答案を採点ツールを用いて採点し，学習者にその結果を返却する．採点結果は，答案の正誤結果と不正解の場合は誤っている点の記述で構成される．このように，答案作成に動きある図式を用いることで，プログラミング言語に依存することなくデータ構造独自のアルゴリズムの理解度を判断できる．したがって，データ構造独自のアルゴリズムにおける学習者の理解度をもとに，教員が学習者に的確な指導を行うことが可能になる．また，学習者は自身の理解度を認識することにより，今後の学習に役立てることが可能である．

3.2 答案作成

答案は，学習者が問題を解くさいに動きある図式に対して行った操作の過程である．本論文では，答案に含ま

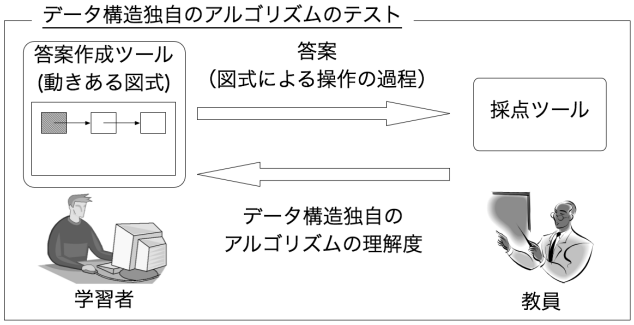

図 1: 提案手法の概要

† 立命館大学 情報理工学部
‡ 立命館大学大学院 理工学研究科

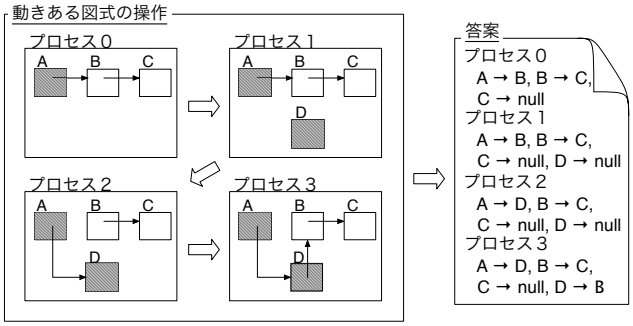

図 2: 単方向リストに要素を追加

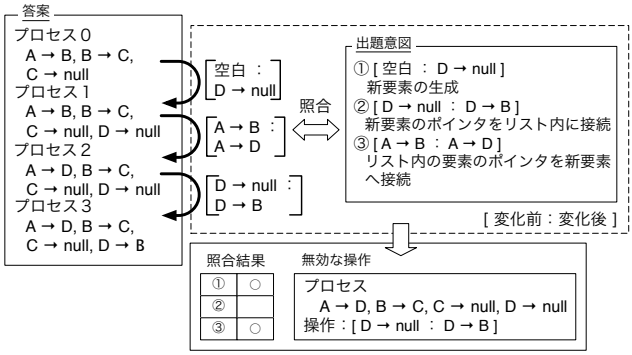

図 3: 答案の照合

れる操作のひとつひとつをプロセスと定義する．答案の作成は，答案作成ツール上でマウスを使い，動きある図式を操作することにより行われる．単方向リストに要素を追加する問題の例を図 2 に示す．答案において，各要素は名前付けを行うことで識別され，次の要素へのポインタは矢印を用いて表現している．たとえば，プロセス 0 では，要素 A，B，C があり，それぞれの要素が持つポインタが指す要素は，B，C，null である．教員に提出される答案では，この状態は,「A → B，B → C，C → null」と表現される．このように本手法では，動きある図式の操作を要素の名前付けにより，文字で表現する．

3.3 教員の役割

問題を作成するさい，教員は解答および出題意図も一緒に作成する．解答は，学習者と同様，答案作成ツールを用いて作成され，出題意図は解答をもとに作成される．出題意図は，解答に含まれる各操作に対してその意味を付加したものである．プロセス $(n+1)$ における操作の意味は，プロセス n からプロセス $(n+1)$ への変化分に相当する．たとえば，図 2 においてプロセス 0 からプロセス 1 へ変化したとき，要素 D が増えている．これより，プロセス 1 での操作の意味は「新要素の生成」であるといえる．

また，採点するさい，教員は採点ツールを用いて学習者の答案を採点する．採点ツールは，出題意図と学習者の答案を照合するものである．

3.4 出題意図の充足と無効な操作

出題意図に記載されている操作が，問題を解くうえで必要な操作である．採点ツールが，出題意図と学習者の答案の各プロセスにおける操作を照合する．リストを用いた例を図 3 に示す．図 3 における答案は，以下の条件下で作成されたものとする．

条件 リストにおける先頭の要素と生成された新要素，およびそれらの要素からポインタでつながれている要素のみが参照可能であり，それ以外は，参照不可能である．

ここで参照不可能とは，その要素を特定したり，その要素に対してポインタを設定したりすることができない状態である．図 3 において，プロセス n からプロセス $(n+1)$ への変化分すなわち，プロセス $(n+1)$ での操作が [変化前 ： 変化後] と表現されている．たとえば，プロセス 0 からプロセス 1 では，D → null が新たに増えているので，[空白：D → null] と表現している．学習者の答案に含まれる操作と出題意図に含まれる操作を照合し，図 3 の照合結果のように，対応する操作が存在すれば，その操作を学習者は理解できていると見なす．

採点のさい，動きある図式上では操作可能であっても，出題上の条件を満たさない操作を無効な操作と呼ぶ．プロセス $(n+1)$ における操作が無効な操作である場合，プロセス n に対して行った操作が無効な操作であるため，無効な操作としてプロセス $(n+1)$ の操作とプロセス n が得られる．図 3 において，プロセス 3 では，要素 D のポインタを要素 B に設定している．プロセス 2 では，リストにおける先頭の要素である要素 A と新たに生成された要素 D のどちらからも要素 B はポインタでつながれていない．したがって，要素 B は参照不可能であり，プロセス 3 における操作は無効な操作であるため，図 3 の無効な操作のように，プロセス 3 の操作とプロセス 2 は無効な操作となる．

3.5 結果の開示

上記の方法により採点ツールは，問題を解くうえで必要な操作をすべて行っており，かつ無効な操作を行っていない場合を正解とする．採点後，教員が正誤結果，不正解の場合はできていない操作の意味，無効な操作がある場合は無効な操作を学習者に返却する．

4. おわりに

本論文では，データ構造独自のアルゴリズムの理解度把握を目的とした，動きある図式を用いたテスト手法を提案した．今後，図式を用いたテスト手法の有効性を検証するために，実装，評価を行う予定である．

参考文献

[1] T.Chen and T.Sobh, A Tool for Data Structure Visualization and User-defined Algorithm Animation, *Proc. ASEE/IEEE Frontiers in Education Conference, Vol. 1*,T1D 2-7, (2001)

[2] Y.Miyadera, N.Huang and S.Yokoyama, A Programming Language Education System Based on Program Animation, *Proc. Education Uses of Information and Communication Technologies, IFIP 16th World Computer Congress*, pp 258-261, (2000)