

プログラミング教育に利用する 再帰プログラム学習支援システム

税 所 幹 幸

1. はじめに

コンピュータのプログラミング教育では、問題を解くための手順(アルゴリズム)を考えた後に、プログラム言語を利用してその手順をプログラムに表現し、コンピュータで実行してその手順が正しいかどうかを確認している。その手順をプログラムに表現する場合、手続きや関数が直接的あるいは間接的に自分自身を呼び出す技法⁽¹⁾ (再帰プログラム)を利用すると、複雑なアルゴリズムが簡潔な表現となるので、プログラミング教育においても再帰プログラムを学習することが重要となる。

しかしながら、再帰プログラムは手続きまたは関数が再帰的に呼び出される構造をとっているので、学習者は、直接的に2個以上の自分自身の手続きまたは関数を呼び出す再帰手続きまたは再帰関数の動き、あるいは間接的に呼び出す再帰手続きまたは再帰関数の動きなどが再帰プログラムの記述だけでは把握しにくく、再帰プログラムの動きを理解することに苦勞している。

筆者らは、再帰を学習する学生に対して教師が授業での説明に利用するために、再帰アルゴリズム(クイックソート、ハノイの塔)の動きを可視化する学習支援システムを設計・開発している。このシステムでは、再帰関数を木構造の再帰木で表現しており、この再帰木を動的に生成してオブジェクト(オブジェクトとは、クイックソートではデータであり、ハノイの塔では円盤と柱である)

がどのように動いていくかを示すアニメーションで再帰アルゴリズムの動きを可視化している。そこで、本システムがプログラミング教育において再帰プログラムの動きの理解に支援可能かどうかを実験してみた。

本稿では、工業高等専門学校を2グループ（講義のみとシステム利用）に分けて再帰プログラム（Pascal 言語を利用）について授業を実施し、本システムが再帰プログラムの実行順序の理解に支援可能であるかどうかを客観的なデータとなる確認試験と主観的なデータであるアンケートで評価した。

2. 学習支援システムの概要

2. 1 再帰手続きまたは再帰関数を表現する再帰木

本システムでは再帰プログラムの動きを可視化する方法に再帰木を利用する。再帰木はn分木の木構造であり、nは再帰手続きまたは再帰関数を構成する手続きまたは関数の数である。

例を利用して再帰木を説明する。図1において再帰手続きpが関数aと2個の自分自身の手続きpから構成されるとする。図1の再帰手続きのプログラムコード例を再帰木に表現した例が図2である。図2のように再帰手続きpが木構造の節点となり、その再帰手続きを構成する3個の関数や手続きをその葉の位置に表示する。また、それらの手続きと関数の実引数の値も併せて表示する。再帰手続きまたは再帰関数が呼び出されるごとに同様な再帰木を木構造に展開する。

```

procedure p( )
begin
  ...
  a( );
  p( );
  p( );
end;

```

図1. 再帰手続き

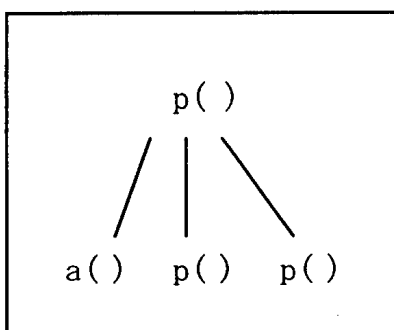


図2. 再帰木

2. 2 本システムの設計

本システムの設計方針を説明する。

- (1) 本システムは教師が授業で再帰プログラムの動きを説明する補助教材として利用する。
- (2) 再帰手続きや再帰関数を木構造の再帰木で表現し、この再帰木を動的に生成してオブジェクトの動きを表示する。
- (3) 再帰木とプログラムコードとの対応を学習者に提示するために、(2)の再帰木の色の変化に同期して、対応するプログラムコードも同じ色で変化させる。
- (4) プログラムの記述には、多くの大学・高専で教育されている Pascal 言語を採用する。
- (5) 2つの再帰プログラムの可視化を実現する。

本システムでは、分割統治法の例として (a) クイックソート、抽象的な表現の代表的な例として (b) ハノイの塔の再帰プログラムを組み込む。

(a) クイックソート

クイックソートでは、再帰木とプログラムコードの表示のほかに再帰手続きや再帰関数の実行前と実行後のデータを表示し、データが並んでいく過程も表示する。また、クイックソートでは、データの分割処理が実行されるので、その過程を説明する画面を別に準備する。

(b) ハノイの塔

ハノイの塔では、円盤の数が4枚で、再帰木および円盤の動きの表示とともにプログラムコードも併せて表示する。

2. 3 学習支援システムの実現

2. 2で述べた設計方針に基づき、2つの再帰アルゴリズムの動きを可視化するシステムを実現した。その使用方法と各再帰プログラムの概要を説明する。

2. 3. 1 システムの利用方法

本システムの利用方法は以下のとおりである。

- ① 学習する再帰プログラムを選択し，起動する。
- ② 再帰プログラムは再帰木を生成して表示するので，教師は再帰プログラムの動きをテキストと画面を利用して説明する。
- ③ その説明後，マウスかキーボードで指示をして再帰プログラムが終了するまで②に戻り，繰返す。

2. 3. 2 各再帰プログラムの概要

(a) クイックソート

実現したクイックソートプログラムには，その実行画面（図3）のほかにピボット値を決めてデータを入れ替える操作を実行する分割処理関数の説明画面を準備している。クイックソートのプログラムを起動すると，上記の2つの画面を必要に応じて選択できる。

図3がクイックソートの実行中の表示画面である。再帰木の色の表示は，黒色が実行前，青色が実行開始，赤色が実行終了を意味する。再帰手続き（quicksort）は分割処理関数（partition）と2個の手続き（quicksort）から成る。下記に再帰プログラムの動きを可視化する過程を詳細に説明する。

- ① 手続き quicksort が呼び出されると，partition 関数と2つの手続き quicksort の再帰木を黒色で展開表示する。
- ② partition 関数が実行を開始し（再帰木および対応するプログラムコードを青色で表示する），終了すると（再帰木および対応するプログラムコードを赤色で表示する），分割するデータ（配列構造）のピボット位置のインデックスを表示すると同時に，2つの手続き quicksort に振り分けたデータを表示する（これらの入力データを黒色で表示する）。
- ③ 第一番目の手続き quicksort が実行を開始し（再帰木，対応するプログラムコードおよび入力データを青色で表示する），さらに再帰手続きを呼び出す場合は，①に戻る。終了した場合には④に進む（第一番目の手続き quicksort の再帰木，対応するプログラムコードおよび出力データを赤色で表示する）。
- ④ 第二番目の手続き quicksort が実行を開始し（再帰木，対応するプログ

ラムコードおよび入力データを青色で表示する), さらに再帰手続きを呼び出す場合は, ①に戻る。終了した場合には呼び出された手続きに戻る (第二番目の手続き quicksort の再帰木, 対応するプログラムコードおよび出力データを赤色で表示する)。

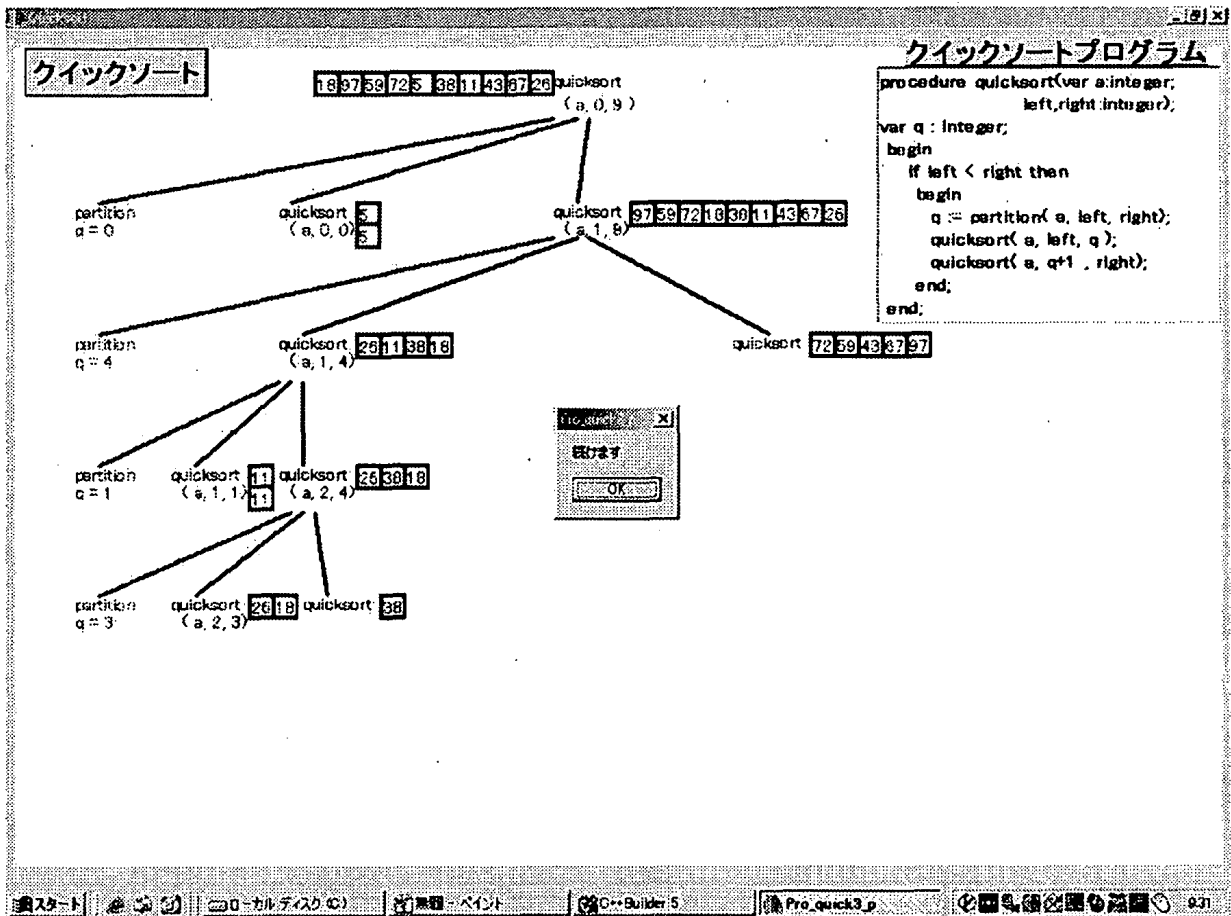


図 3. クイックソートの実行画面

(b) ハノイの塔

図4は円盤数が4枚のハノイの塔の再帰プログラムの実行画面である。図の画面では、再帰木の動きに合わせて円盤の移動を表示し、プログラムコードも表示する。円盤の表示内容は、円盤と柱、円盤の番号および移動元とその先の柱名である。再帰プログラムの動きを表現する再帰木とプログラムコードとの色の変化は、クイックソートでの説明と同様に、手続きの実行開始（青色表示）と終了の時点（赤色表示）でおこなっている。

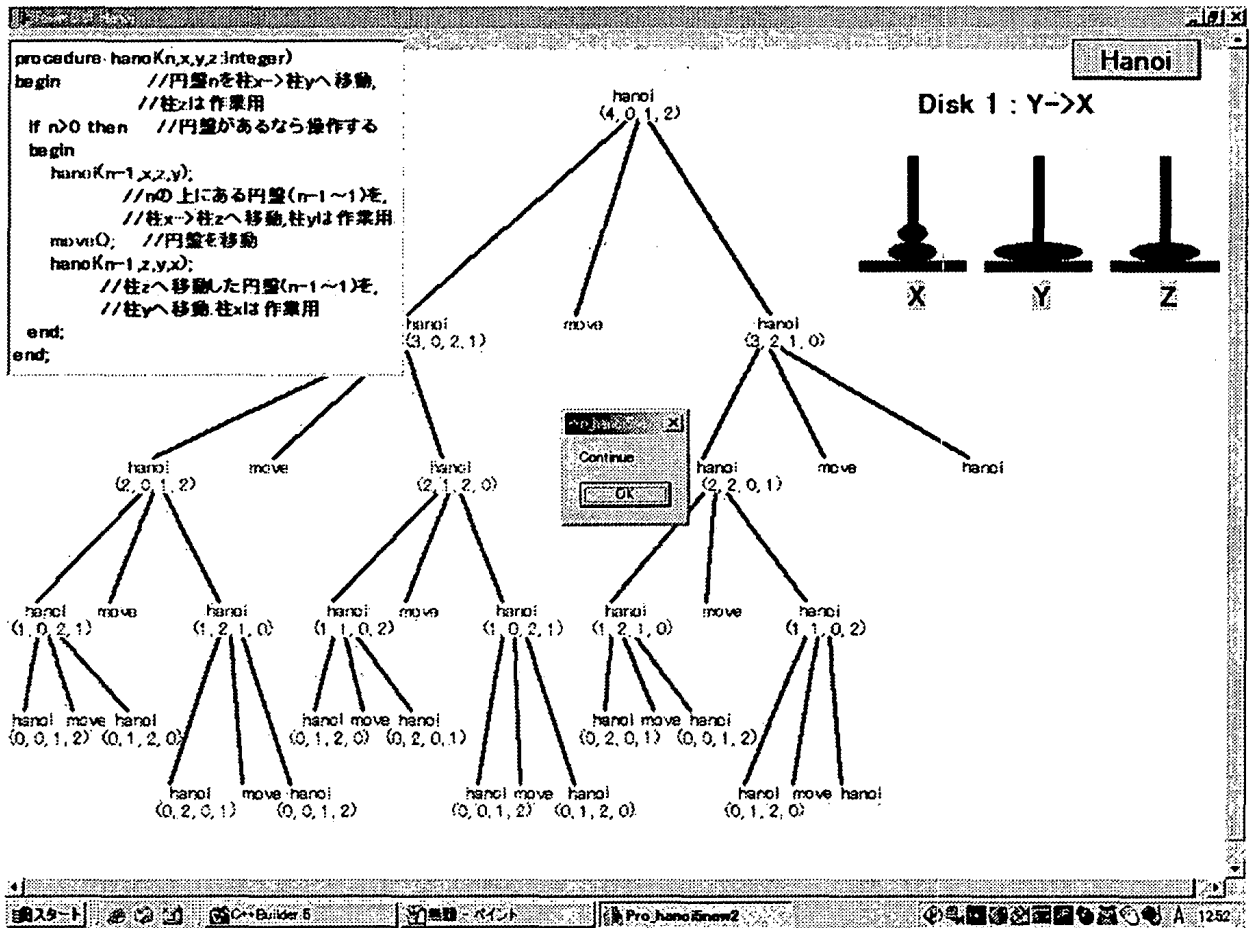


図4. ハノイの塔の実行画面

3. システムの評価

3. 1 被験者

熊本電波工業高等専門学校・情報工学科の3年生（37名）に対して「プログラミング通論」の授業科目で本システムの評価を実施した。本システムはクイックソートとハノイの塔の再帰プログラムを提供できるが、今回は、上記の授業科目で取りあげている関数と手続きにおける再帰呼び出しの項目で、再帰呼び出しの一例としてハノイの塔の再帰プログラムを利用した。

3. 2 評価実験方法

システムの学習支援可能性を評価するために、講義のみ（19名）とシステム利用（18名）の2つのグループに振り分け、上記の再帰呼び出しの一例としてハノイの塔の再帰プログラムで実行順序を説明後、各グループの学生に対して、システムの客観的な評価として実行順序を確認する試験とシステムの主観的な評価としてアンケートを実施した。講義のみのグループでは説明用資料⁽²⁾（この資料には再帰木の表現を含んでいなかった）を利用した授業形式で、またシステム利用のグループでは上記資料と本システムを利用した形式で実施した。

3. 3 評価結果と考察

(1) 確認試験

客観的に評価するための確認試験問題はハノイの塔の再帰プログラムを利用し、そのプログラムの実行順序にしたがって手続き名およびその引数と出力結果を記述させた。表1に各グループの試験の平均点（10点満点）を示す。

表1の各グループの平均点に対してt検定をおこなった結果、講義のみのグループとシステム利用グループとの間には有意水準5%で有意差が認められた（ $t(35)=2.084$, $p<.05$ ）。この結果より、教師が授業で本システムを再帰プログラムの実行順序の説明に利用することは、学生が再帰プログラムの実行順序の理解を高めるために有効であるといえる。

表1. 各グループの確認試験の平均点 (10点満点)

	講 義	システム
ハノイの塔	2.7	4.9

(2) アンケート

アンケートの回答項目は、「できた」、「少しできた」、「あまりできなかった」、「全くできなかった」の4つからなる。この各項目にそれぞれ4点から1点を配点し、点数化した。表2が授業で取りあげた再帰プログラムに対する講義のみとシステム利用のグループのアンケート結果 (平均点) である。

「再帰の概念の理解」では、講義のみとシステム利用のグループともに平均点は同じ値となり、2つのグループ間には差が認められなかった。主観的には2つのグループの学生は再帰の概念をおおむね理解している結果となった。コメントでは、「再帰プログラムの実行順序がわかりづらい」と書いている学生が講義のみのグループに3名、システム利用グループに1名いた。

次に、「ハノイの塔の実行順序の把握」では、システム利用のグループの平均点は、講義のみのグループに対して、率にして約44%高かった (表2)。また、2つのグループ間における t 検定の結果は、有意水準5%で有意差が認められた ($t(30) = 2.382, p < .05$)。この結果、システムを利用して再帰プログラムの実行順序を学習することは有効であると学生は実感していることがわかった。学生は、「ハノイの塔はわかるが、プログラムが把握しにくい」(2名)、「再帰プログラムでは自分自身を繰り返す関数が含まれるので、再帰関数が2個含まれるとその動きがわからなくなる」(2名)、「実行順序が把握しづらい」とコメントを記述しており、再帰プログラムの実行順序を理解することに苦勞している。システム利用者の中にも実行順序の理解が難しいと答えている学生もいた。

表2. 講義のみとシステム利用グループのアンケート結果（4点満点）

アンケート項目	講義	システム利用
再帰の概念の理解	2.6	2.6
ハノイの塔の実行順序の把握	1.8	2.6

システムを利用したグループに対して本システムに関するアンケートを実施した結果を表3にまとめた。

「再帰木の理解」については、平均点は2.8と高い値ではなかったが、学生のコメントでは、「プログラムよりはわかりやすい」、「とても見やすく再帰がどのようにおこなわれているかがよくわかった」、「図で見るとわかりやすい」との意見があった。一方、「再帰木の意味がつかめなかった」との意見があり、限られた授業時間内に再帰木の説明が不十分だったのではないかと反省している。今後システムを利用する場合にこの意見を生かしていきたい。

次に、「再帰木等の色の変化によるプログラムの実行状況の表示」のわかりやすさは、平均点が3.4であり、学生のコメントは、「色の変化は今どのような実行状況になっているのかがよくわかった」（5名）、「実行状況がとてもわかりやすかった」（2名）であった。アンケートの平均点とこれらのコメントから、再帰木等の色の変化でプログラムの実行状況を示すことは学生にとって有効であると考えられる。

「再帰プログラムの把握に再帰木による説明」の有効性は、平均点が3.5であり、学生のコメントは、「プログラムの実行順序がわかりやすくて良かった」（3名）、「プログラムを視覚的に把握できるので有効だ」（2名）であった。これらのコメントから、再帰プログラムの説明に有効であることが確認できた。

表3. 再帰木に関するアンケート結果（システム利用グループ）

アンケート項目	平均点（4点満点）
「再帰木の理解」	2.8
「再帰木等の色の変化によるプログラムの実行状況の表示」のわかりやすさ	3.4
「再帰プログラムの再帰木による説明」の有効性	3.5

上記(1)の確認試験の客観的なデータ結果と(2)のアンケートの主観的なデータ結果から、再帰木およびオブジェクトの動きを利用して再帰アルゴリズムの動きを表示する本システムは、教師がプログラミングの授業で補助教材として利用可能であり、再帰プログラムの実行順序の理解に対して支援が可能であることが確認できた。

4. まとめ

筆者らは、授業で利用できる、再帰木を動的に生成してオブジェクトの動きで再帰プログラムの動きを可視化する学習支援システムを設計・開発している。本システムを工業高等専門学校でのプログラミングの授業に利用した後、客観的な評価となる確認試験と主観的な評価となるアンケート結果により、教師が授業で再帰プログラムの実行順序の説明に本システムを補助的な教材として利用できることが確認できた。

今後の課題は、再帰プログラムの作成支援をおこなうシステムを構築することである。

謝辞

本研究を進めるにあたり、日ごろからご指導いただいている熊本大学・中村良三教授に感謝致します。また、本システムの評価にご協力いただいた熊本電波工業高等専門学校・孫寧平助教授に感謝致します。

参考文献

- (1) Wirth, N.: “Algorithms + Data Structures = Programs”, Prentice-Hall, Inc. (1976), 邦訳 : 片山卓也訳: “アルゴリズム+データ構造=プログラム”, 日本コンピュータ協会(1993)
- (2) 伊藤正安, 金谷範一: “明解 PASCAL”, サイエンス社