

# アルゴリズムアニメーションを用いた 「アルゴリズムとデータ構造」授業の試み

武藤武士

奈良工業高等専門学校 情報工学科\*

mutoh@info.nara-k.ac.jp

**Abstract** 情報系の科目として「アルゴリズムとデータ構造」は全てのプログラミングの基礎を与える重要な科目である。一般に、アルゴリズムは疑似プログラムなどの静的な形で提示される事が多いが、学習者はこの動きを頭や図的な表現として動かしてみることで、理解を深める必要がある。本報告では整列アルゴリズムの動作をアニメーションで提示するアプリケーションを Tcl/Tk を用いて実装し、実際に授業で利用した結果を報告する。アンケートの結果からは、図でのアルゴリズム教示が学習に好影響を与えるという結果は得られたが、アプリケーションを利用しての学習での効果は現れなかった。さらに学生の要望が多かったことから、学生が自分の作ったプログラムの動作を見るための Xlib 利用のライブラリも作成し、今年度の授業で利用する予定である。

キーワード:アルゴリズムアニメーション, アルゴリズムとデータ構造

## 1 はじめに

情報工学科のカリキュラムとして、「アルゴリズムとデータ構造」の授業は全てのプログラミングの基礎となる点から重要である。特に、情報処理技術者試験の近年の問題 [1] をみても、アルゴリズムに関する問題が増えてきていることから、これを理解することは情報処理技術者にとって重要なことである。奈良工業高等専門学校情報工学科では、3 学年に 2 単位の授業が用意されており、筆者がその授業を担当している。一般に、アルゴリズムは疑似言語などのプログラムとして提示されることが多いが、学習者はこの動きを頭や図的な表現として動かして見ることで、理解を深める必要がある。アルゴリズムの動作の理解のために、アルゴリズムアニメーションを用いるなど、その動作を図示する方法がしばしば用いられる [2]。本研究では、アルゴリズムの学習者に (整列) アルゴリズムの動作をアニメーションで示すことで、学習者に動作を示すことを試みた。このアプリケーションは Tcl/Tk で実装されており、UNIX をはじめとするさまざまなオペレーティングシステムで動作する。本報告では、実際に授業で利用し、アンケートを行った結果について報告する。このアンケートから、作成したアルゴリズムの動作を確認できることに対する学習者の興味が高いことがわかったので、Xlib をつかって簡単なアニメーションライブラリを作成した。これは、本年度の授業で利用する予定であるが、ここではその実装について報告する。

ーションライブラリを作成した。これは、本年度の授業で利用する予定であるが、ここではその実装について報告する。

### 1. 講義目的

プログラミング I,II で学んだプログラムの基礎が理解できていることを前提とし、プログラム設計の際に問題となる「アルゴリズムとデータ構造」に関する基礎を学び、修得することを目的とする。

特に実装する言語は問わないが、Pascal もしくは C が使えることが望ましい。

### 2. 内容

- アルゴリズムと計算量

- 基本的データ構造

- 探索アルゴリズム

- 整列アルゴリズム

バブルソート, 選択法, 挿入法シェルソート, クイックソートヒープソート, マージソート

3. 評価方法 テスト 40%, レポート (毎回) 40%, 出席 20%

4. 教科書「アルゴリズムとデータ構造」, 石畑 清, 岩波書店

\*大和郡山市矢田町 22

図 1: シラバスの内容

## 2 計算機ソフトウェア I「アルゴリズムとデータ構造」の概要

奈良高専情報工学科3年では、計算機ソフトウェア I「アルゴリズムとデータ構造」が、アルゴリズムの学習のためのカリキュラムとして用意されている。図1にシラバスの内容を抜粋する。授業の半分は計算量の定義や基本データ構造で費され、残りの2/3が整列アルゴリズムの解説に当てられている。

「アルゴリズムとデータ構造」の授業では、講義でのアルゴリズム解説以外に、情報処理工学科実習室のFreeBSD[3]を用いて、実際にアルゴリズムを実現する実習を行っている。この際、実装はPascalで行い、p2cトランスレータを用いて、C言語に変換した後にコンパイルし、実行している。実習は最初の説明のみを授業で行い、あとは課題を与え放課後に行っている。整列に関する課題の内容は、以下の通りである。

- $O(n^2)$  のアルゴリズムの中からどれか一つ実現する
- シェルソートの実現
- クイックソート (自分で可能な限り高速化したもの)
  - [追加課題] 全然高速化していないものと速度を比較しなさい。
- [追加課題] その他の整列アルゴリズムについて実現してみなさい。

実際に実現したアルゴリズムでデータの数  $n$  を変化させて、実行時間がどうなるかを検討することも課題に含まれている。このレポートは毎週提出の通常のレポートの5回分の加点を行っているため、学生は熱心に取り組んでいる。

## 3 アルゴリズムアニメーションツール

### 3.1 実装

アルゴリズムアニメーションツール(以下, sorter) [5] は, Tcl/Tk で実装されたアルゴリズム可視化アプリケーションであり, Tcl version 7.6 jp 以上, Tk version 4.2 jp 以上 [4] が動作するさまざまなオペレーティングシステム上で動作可能である。sorter の動作の様子を図2に示す。

左端のメニューからアルゴリズムを選択し、実行メニューで動作を行わせる。アプリケーションの実行時

間時間が時間欄に表示される。これは、実際の計算時間ではないが、アルゴリズムの動作時間の目安を与える。データの初期化やファイルからのデータの読み込みなどはデータ操作メニューから行うことができる。スピードレバーにより動作速度を調整することが可能である。

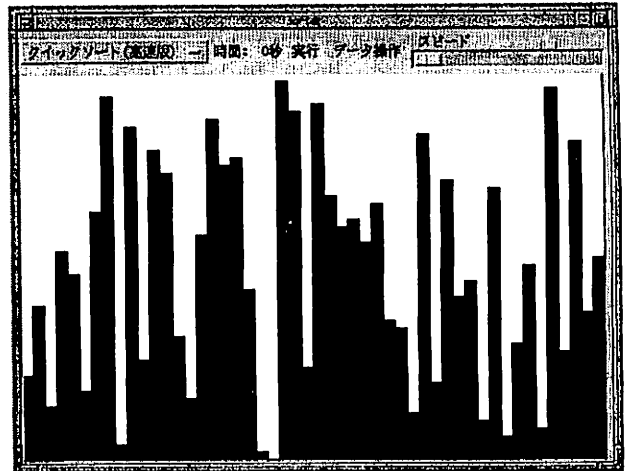


図 2: sorter の動作の様子

データは、長方形の高さで要素の大きさを表し、要素の値を検索している時にハイライト表示される。さらに、データが交換される場合には、色を変化して交換される2要素を表示するようになっている。

実装したアルゴリズムを表1に示す。アルゴリズム名の後に付けた○は、授業で解説したアルゴリズムである。基本的に授業で習うアルゴリズムの中で、可視化に問題の無いものは実現している。

表 1: sorter で実装したアルゴリズム

アルゴリズム	授業
馬鹿ソート	○
バブルソート	○
ダブルバブルソート	
選択ソート	○
シェーカーソート	
挿入ソート	○
シェルソート	○
ヒープソート	○
その場マージソート (遅い)	
マージソート	○
自然マージソート	○
クイックソート	○
クイックソート (高速版)	○
LOAS	

ヒープソートは、本来2分木として表現した方が動作の理解がしやすいが、ここでは1次元の配列として表現できるアルゴリズムがその対象となっている。ヒープソートも実現しているが、直観的にその動作が分かりやすい図示とはなっていない。

マージソートは、本来別の記憶域を必要とするが、この別の記憶域でのデータの扱いを画面の上から表現することで、疑似的に別記憶域を表現した。

### 3.2 アンケート結果

実習は探索アルゴリズムの実現と整列アルゴリズムの実現に分かれており、後者の実習中に `sorter` を用いて、実際のアルゴリズムの動作を確認してもらった。授業の最後に WWW をつかって、アンケート調査を行い、学生 40 名中 38 名の回答を得た。アンケート項目と結果を表 2 に示す<sup>1</sup>。評価は各項目 1～5 の 5 段階評価で行い、点数が多い程肯定的な評価となるように設定した。

表 2: アンケート項目と結果

アンケート項目	評価
整列の授業はわかりやすかったか?	3.87
図をつかった説明はどうだったか?	4.03
授業だけでアルゴリズムの動作が理解できたか?	3.32
Sorter の操作性はどうでしたか?	3.84
Sorter をつかって、アルゴリズムの動作が理解できましたか?	3.63
このようなプログラムをつかって授業を進めた方が良くと思いますか?	3.95
自分の整列でこのようなアニメーションを見たいか?	4.32

アンケート結果から、図をつかった説明によって、アルゴリズムの理解が深まっていると考えることができる。`sorter` の操作性に関しては、特に問題がなかったようである。`sorter` を利用したことによる効果は、アンケートから見出すことはできないが、授業に興味を持たせるためにはこのようなツールの利用も有効であることはわかった。学生は特に自分の実装したアルゴリズムの動作を、実際に見れるような環境に対して興味強いことがわかった。

<sup>1</sup><http://herb.info.nara-k.ac.jp/answer.cgi>

## 4 アニメーションライブラリ

アンケートの結果から、自分で作成した整列アルゴリズムの動作を図示するようなツールに学生は強い興味をしめしていることが分かった。そこで、簡単なアニメーションライブラリを Xlib を用いて実装し、本年度の授業で利用する予定である。

このライブラリは、`sorter` での整列アルゴリズムの可視化と同じように、ハイライト表示と交換の表示が行えるようになっている。プログラムの検索部分や要素の交換部分に所定の関数を呼び出すことで、これらの図示が行われる。このような指示関数を表 3 に示す。

表 3: アニメーションライブラリで用意した関数

関数名	機能
<code>AnimSetWaitTime(int time)</code>	待ち時間 <code>time</code> ( $\mu$ 秒) の設定
<code>AnimInit(int dataSize)</code>	アニメーションの初期化
<code>AnimInitData(int i,value)</code>	<code>i</code> 番目の初期データ値 <code>value</code> の設定
<code>AnimShowData(int dummy)</code>	初期データの表示
<code>AnimSwapData(int i,j)</code>	データ <code>i,j</code> の交換
<code>AnimHilightData(int i)</code>	データ <code>pos</code> の強調表示
<code>AnimUnhilightData(int i)</code>	データ <code>pos</code> の強調表示を止める

実際に本ライブラリを使用して、馬鹿ソート [6] を実現した例を図 3 に示す。下線部分が、アニメーションを行うために追加した部分である。ここでは、データの交換時のみに色を変えて変更が分かるようにしているが、データを調べるとハイライトしてそれがわかるように変更することも可能である。このプログラムの動作の様子を図 4 に示した。

## 5 おわりに

アルゴリズムの授業はともすれば、疑似プログラムのコードを追うだけのものになりがちであるが、図的に動作を示すだけで、理解度は格段に向上する。黒板での図的な表現が有効であることは明らかであるが、学生の好奇心を刺激するために、`sorter` のようなアルゴリズム可視化ツールを用いて、授業を行っていくことも有効であろう。本年度の授業では、アニメーションライブラリを用いることで、学生が自分の作ったアルゴリズムの動作を確認することで、更に深い理解ができるものと期待される。実際、授業アンケートの結果

```

#include "anim.h"
#define DATA_SIZE      300
int data[DATA_SIZE];
int main() {
    int i,j;
    srand((int)time(NULL));
    AnimInit(DATA_SIZE);
    /* データの初期化 */
    for(i=0;i<DATA_SIZE;i++) {
        data[i]=(int)(DATA_SIZE*rand1());
        AnimInitData(i,data[i]);
    }
    AnimShowData(0);
    /* 実際の馬鹿ソート */
    for(i=0;i<(DATA_SIZE-1);i++)
        for(j=i+1;j<DATA_SIZE;j++) {
            if(data[i]>data[j]) {
                swap(i,j); AnimSwapData(i,j);
            }
        }
}

```

図 3: 馬鹿ソートの実現例

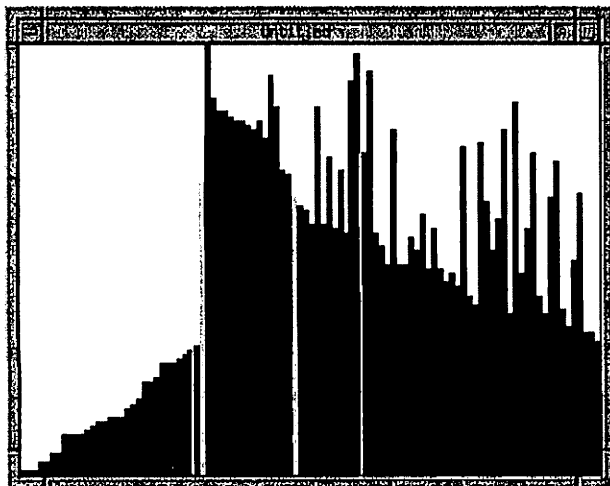


図 4: 馬鹿ソートの動作の様子

<sup>2</sup> から実習を通して、アルゴリズムの理解が深まったという感想も得られているため、このようなアプローチは有効であろう。

## 謝辞

1997 年度奈良高専情報工学科 3 年生には、アプリケーションを利用し、アンケートに答えていただきました。ここに、感謝いたします。

## 参考文献

- 1) 武藤武士, 榊原和彦, 成田絃一, “WWW を利用した選択式問題テスト採点システムの試作”, pp.69-72, 情報処理教育研究集会予稿集 (1996).  
または,  
<http://herb.info.nara-k.ac.jp/InfoExam/>.
- 2) R. セジウイック, “アルゴリズム”, 近代科学社 (1990).
- 3) 武藤武士, 工藤英男, “情報工学科新実習システムの構成とその運用”, pp.198-201, 情報処理教育研究発表会予稿集第 17 号 (1997).  
または,  
<http://herb.info.nara-k.ac.jp/>.
- 4) 宮田重明, 芳賀敏彦, “Tck/Tk プログラミング入門”, オーム社 (1995).
- 5) 武藤武士, “sorter のページ”,  
<http://sentinel.info.nara-k.ac.jp/Sorter/> (1998).
- 6) 石畑清, “アルゴリズムとデータ構造”, p.147, 岩波書店 (1989).

<sup>2</sup><http://herb.info.nara-k.ac.jp/CLASS/SoftI/1998questionary.html>