

# 教育学科で J A V A プログラミングを教える

## Teaching Java Programming at the Department of Education

山口 栄一\*

玉川大学文学部

織田 真生\*\*

前玉川大学大学院生

町田市玉川学園6-1-1

TEL 0427-39-8018

E-mail:yama@lit.tamagawa.ac.jp

Abstract:Java 10 Lessons was developed to teach Java applet programming to students at the department of Education where most students are not motivated to learn programming itself, but are interested in Internet for their own use. At the beginning of the Java programming course, they were shown what they could make at the completion of the 10 lessons, i.e., a simple interactive CAI system with animation and audio. They managed to learn Java programming in a piecemeal manner.

Key words: introduction to Java programming, improving curriculum, teaching programming

### 1. はじめに

社会の高度情報化に対応するだけでなく、学生たちの希望に応じて、教員養成に関わる筆者らの学科でも、コンピュータ教育のカリキュラムを用意してきた。しかしながら昨今では、いわゆる「総合学習」を中心とする新しい学力観のもとで、さらに充実した内容が求められるようになってきている。それは、特にインターネットの利用を含んでいることはいままでの玉川大学文学部教育学科でも、インターネットの教育利用に対応する内容を含んだカリキュラムを組んできたが、それはWebの検索とホームページづくりにとどまるものであった。しかし、インターネットが小学校、中学校に導入されはじめている。それはとりもなおさず小・中学校の子どもたちとプログラミング的には同様のレベルであるために、大学ではより高度な内容を学生に与える必要もみえはじめてきた。そのために筆者らは、HyperCardを使用したマルチメディア教材づくりを通して行っていたプログラミング教育を、ネットワークにおけるマルチメディア教材の開発に拡大し、ホームページづくりにおける、より高度のプログラミング能力を高めることを意図して、昨年度よりJavaプログラミングを教える実験的な授業を開始したのである。本稿では、私たちがこのような経緯で開発した、Javaプログラミング教育カリキュラム「Java 10 Lessons」の考え方と、その内容、および、実験的な授業での学生の反応を紹介する。

### 2. Java 10 Lessons 開発の考え方

筆者らがJavaをプログラミング教育の対象としたのは、いくつかの理由がある。第一に、それは、インターネットへ対応しながら、HyperCardと同様に、部分的な発想でプログラミングが可能であるからである。第二に、JavaはWindowsやMacintoshなどの上で稼働するOSに依存しない言語であり、将来性が期待されるためである。本学は

Macintoshを使用しているが、学生たちが就職する多くの学校では、Windowsマシンを利用しているはずである。教材の作成にあたっては、プラットフォームに独立することは魅力的である。しかも、開発ソフトを無料で入手できるのは、何よりもありがたいことである。第三に、C言語のように難しくない、というメッセージも魅力的である。教育学科の学生が必要にかられてそうした言語を学ぶのは悪いことではないが、プログラミングのために多くの授業時間を費やすことはできないからである。できるなら、プログラム作成が簡単で、高度なことができるものが望ましいのである。

開発の考え方は、これまでのマルチメディア教材開発のカリキュラムと同じである。将来の情報教育を担う教育学科の学生であれば、生徒たちが使用するコンピュータ教材の仕組みをある程度理解しておくことは必要であり、さらに、担当する生徒たちのために教材を自作できることは望ましい。しかしながら、それにはかなりの努力が求められるので、そのレベルに到達することは多くの学生にとって困難と思われる。そのために、筆者はこれまでのマルチメディア教材づくりの授業の中で、教材となるさまざまな部品のプログラムを用意し、それをカスタマイズすることによって、それぞれの作品を制作するカリキュラムを、HyperCardを使用して開発してきた。はじめから作り上げることができなくとも、部品を理解し、それをカスタマイズできる程度のことのできるならば、それはかなりの力となりうると思われるからである。筆者はこれをブラックボックスのグレーボックス化とよび、学生たちが部品を理解し、それを利用できるようにする方向を授業としてめざしてきた。HyperCardとその言語であるHyperTalkは、その目的にきわめて適したものであった。

この考え方にもとづくHyperTalkのプログラミングの学習「HyperTalk 20 Lessons」は、学生たちの興味を喚起してきた。それは学生にとって、何を行っているのかが明確に理解できると思われるからである。部品を作り上げる命令は、部品とともに理解される。「学習者は自分の活動を自らコントロールしながら学ぶときに、よりよく学べる」ことが明らかであるとすれば、コースの設計では、できるだけ彼らが目的をもって活動できる場を与えることが必要で

\*Eiichi Yamaguchi

Department of Education, Tamagawa University

\*\*Masao Oda,

graduated at the Med. course of Tamagawa University in March, 1997.

ある．そのためには、コースを学べば何ができるようになるのかを理解し、その過程で自分なりのものを作ることのできる機会が用意されるべきである．つまり、目的と同時にその部品を提示し、それをカスタマイズして使えるようにすることは、プログラミングへの動機と目的を与える意味でも、意義があると考えているのである．<sup>(注1)</sup>

この「Java 10 Lessons」開発もこれと同じ考え方で構成されている．しかしながら、C言語と比べて「Javaはやさしい」とは言われるけれども、それはプロのレベルの話であって、プログラミングの初心者にとって、Java言語はHTMLやHyperTalkと比べると、はるかに難しいものである．彼らがそれを「自分のものになった」という感覚をつかむのは、とても困難であるように思える．なぜなら、そうした感覚は、自分が作りたいものが作れたことによって生まれるものであり、そこまでいかなくとも、ある程度のものできたときに、感じるができるものだからである．しかし、逆に言えば、ある程度のもを自分で作れたという手応えが得られたならば、それなりの達成感と自信も得られるのではないかと思う．したがって、「これだけのことが差し当たりできればよいのではないか」という感覚で目標を設定している．

しかしながら、HyperTalk プログラミングと異なり、それに至る手続きがかなり複雑なために、目標となる課題は絞られなければならない、レッスンの構成も、最終目標を設定し、その下位目標を達成していく、という積み重ねたやり方を用いざるを得なかった．そのために「Java 10 Lessons」の開発では、まず第一に、学習者が作る最終的な目標となるソフトウェア Java Applet を設定している．ただし、下位目標となるアプレットでも、ホームページに載せられるくらいの、意味のあるものであることに心がけたことは、このレッスンの特徴である．このレッスンを終了するときに学習者が作成できるようになるのは、「動画・音声を使った選択・記述式のCAI型教材」のアプレットである．これは教育関係者だけでなく、プログラミングに興味ある初心者にはおもしろいものとなるはずである．

図1は、最終目標となるLesson10のアプレットを組み込んだHTMLの画面である．Lesson10では、「I am standing on two legs. Who am I?」という問いの画面があり、ユーザーにその回答の入力を求める．そして、ユーザーが「penguin」と入力すると、図1の正解の画面を組み込んだページが出てくるようになっている．このアプレットでは、図のペンギンが画面を飛び跳ねながら、横に移動して歩いていく．

学習者が、自分が始めようとしている練習によってどのようなものができ上がるのかを明確に知るためには、完成した作品の例を見ることが大切である．「Java10 Lessons」では、それぞれのレッスンごとに、何が作成できるのかを見ることができるようになっている．学習者は各レッスンの始めに、完成したアプレットの例を見たり、操作したりすることができる．

レッスンでは、できるだけ専門概念を使用することを避け、わかりやすい説明を心がけている．プログラミング専門用語を多用した説明は、初心者のプログラミング学習においては、つまずきの原因になっているからである．もち



図1 最終目標となるアプレット

ろん「継承」、「イベント」などの専門用語も結果としては用いるとしても、「継承」をテーマとして直接取り上げることはなく、プログラムの説明の中で、日常現象による表現、特に、説明にはアナロジーを使用した説明の中で、簡単に言及されるにとどまっている．例えば、インスタンス化するなどは、「ギャングのボスは、道具を貸してくれるが、足がつくといけないから名前は変えて使わなければならない」など、やや乱暴な言い方も含まれている．また、「ライブラリからインポートする」は「東急ハンズのような店からもってくる」とか、「継承する」は「改造して使う」、「型宣言する」は「部品に名前を付けて使えるようにする」と表現し、基本的には、最終レッスンまでこうした表現を使い続けてきた．

プログラムはプリント教材として提示し、解説はソースリストと1対1に近い形で細かく行うことで、できるだけ学生の学習の負担を軽くするようにしている．つまり、サンプルを打ち込めば、そのレッスンの目標となるアプレットを作り上げることができる．学生にはそれを踏まえた簡単な課題が与えられるが、それは、それを利用して自分なりの作品を作るものであって、カスタマイズの初歩にほかならない．

### 3．授業とJava 10 レッソンの構成

Java 10 Lessons を実験的に使用した環境、対象となった学生、その内容については次の通りである．

#### (1) 授業と利用環境

[ 授業科目名・単位 ] コンピュータ教育2・2単位(12回)  
[ 対象 ] 文学部教育学科2, 3年生  
[ 授業形態 ] 一斉及び個別指導

[ 機器 ] Macintosh Power PC 6100/66

[ 利用環境 ] コンピュータ演習室

[ システム ] ファイルサーバ1台, 学生用の PC 40台がイーサネットのLANでつなわれ, インターネットに接続している。

実験授業を行った「コンピュータ教育2」は, 教育学科の2, 3年生を対象としており, 「コンピュータ教育1」に続く, 半期科目である。ほとんどの学生は, 前期科目の1をとってからこのコースを選択する。従来は, 1, 2ともに, 「HyperTalk 20 Lessons」を使って, 教材作成の実習を行っていたが, この実験授業のために内容を変更した。学生もJavaがどういうものかを知って興味を示し, 内容の変更を了解した。

## (2) Java 10 Lessons の内容

「Java 10 Lessons」の最終目標および下位目標を達成するための各Lessonの構成は次の通りである。なお, 使用したのは, JDK1.02のMacintosh版である。

### Lesson 1 : アプレットをもらおう

[ 目標 ] あらかじめ用意されたソースリストをコンパイルしてクラスファイルを作り, それをHTMLから呼び出して使用する。また, コンパイルしたお絵かきアプレットを操作する。

[ 解説 ] ここでは, Webのページにアプレットを組み込むための手続きを学習する。すなわち, HTMLやJavaプログラムを書くために使用するエディタやワープロの使い方とファイルの拡張子の書き方, コンパイルの概念とその作業の方法, ホームページにJavaアプレットを組み込むときに必要なHTMLのタグ(命令)の書き方を学習する。MacintoshでのコンパイルはWindowsと異なり, アイコンの操作でできるのは便利である。解説の中で, アプレットもまたWebページを構成する部品と考え, こうして他人の作ったものを利用する, という感覚で考えていくことを説明している。

### Lesson 2 : アプレットに絵を出そう

[ 目標 ] テキストと絵をアプレットで表示する。

[ 解説 ] プログラミングの第一歩として, テキストや1枚の絵を表示するアプレットを作る。そのために必要な, ファイルの読み込み, 表示のための準備を学習する。また, Javaプログラミングに共通して必要な, 既存プログラムを使用する方法, 各種の宣言, Javaプログラムの基本型を学ぶ。もっとも, はじめてのレッスンとしては内容が多いので, 説明よりもどのようになるのかを予想し, 自分のテキストと自分の選んだ絵を表示できるアプレットができる, という達成感を重視している。

### Lesson 3 : 簡単なアニメを作ろう。

[ 目標 ] 複数の絵を切り替えて表示する方法を用いて, 簡単なアニメを作成する。

[ 解説 ] Webページを開いたときに1回だけ動画を表示するアプレットを作る。繰り返し命令と配列を使ってプログラムを整った形にしたり, 古くから行われてきた, for文を使った時間稼ぎの方法を用いて, 切り替え速度を調整するといった改良も行う。絵の表示のタイミングをあわせる方法としては, 次のレッスンからはスレッドを使用するが, ここでは条件によって実行条件を変えるための練習として, for文を組み込んでいる。配列を使用する目的を実感させるために, 絵を読み込むに際して, すべての絵ごとに変数を指定するプログ

ラムを実行し, その作業のあと, 配列を使って変数指定の作業を簡略化する方法を用い, プログラムの改善を行う。

### Lesson 4 : ボタンでアニメ

[ 目標 ] ボタンの制作とボタンによる制御を学ぶ。

[ 解説 ] ボタンをクリックするごとに, 動画が動作・停止を繰り返すアプレットを作る。また, このアプレットばかりでなく, Javaアプレットのプログラミングには不可欠のスレッドを学習する。動作の実行と停止を, 変数の値の切り替えで制御する方法を知る。

HyperCardなどではボタンは道具として用意されており, プログラムもボタンのスクリプトとして書き込めばよかったが, Javaではそれを部品として読み込み, イベント処理としてプログラムするという, これまでとは異なるプログラムの方法に出会うことになる。

### Lesson 5 : 複数のボタンでアニメ

[ 目標 ] 二つの動画を切り替えて制御する方法を学ぶ。

[ 解説 ] lesson 4では, ボタンが一つでイベント処理を行うことを学んだのであるが, ここでは二つのボタンを使って, 2種類の動画の切り替えを行うアプレットを作る。ただし, このレッスンはイベント処理に条件分岐のif文が加わった程度なので, プログラム的には難しいものではない。課題としては, ボタンの数を増やしたり, 切り替える絵を別のものにする。

### Lesson 6 : 音を鳴らそう

[ 目標 ] 音声再生の方法を学び, さらに, 音声と動画を組み合わせるとい課題を行う。

[ 解説 ] ボタンをクリックすると音声再生されるアプレットを作る。そのために必要な音声ファイルの読み込み, 再生のための準備を学習する。このプログラムは, AudioClipを自分の部品として宣言して, playを設定するだけであって, しかもそれは, イメージの取り込みと似たようなプログラムであるために, 学生たちには簡単と思える。そのため, このレッスンではさらに, アニメを動かしながら音を鳴らすという, これまでのプログラムと組み合わせるとい課題を出題している。

### Lesson 7 : 他のページにとぶ

[ 目標 ] 複数のアプレットをリンクする(呼び出す)方法を学ぶ。

[ 解説 ] ボタンをクリックすると別なアドレスのWebページにリンクするアプレットを作る。ページをリンクさせるのに必要な準備とリンクの方法を学習する。ここでは, URLを指定して, 呼び出す方法をとっている。ページをめくるタイプのCAIの画面は, 類似したプログラミングの繰り返しとなるので, このレッスンから, 実践的な教材を作れるようになる。なお, 彼らはCAIの基本的な要素(情報提示, 反応喚起, 評価とフィードバック)と, それがプログラムの中でどう構成されるかをHyperCardやHTMLでの実習の中で学んでいるために, ページをリンクすることの意味はすでに理解している。

### Lesson 8 : 自作のクラスを使う

[ 目標 ] 自作の部品(スーパークラス)を作る。

[ 解説 ] 一つのクラスを利用(継承)して, 複数のボタンを使った多肢選択式問題のクラス(アプレット)を

作る．そのために，メソッドを利用して構造化されたクラスを作る．ただし，解説では，彼らにクラスの継承といったプログラミングの概念を教えることではなく，部品として共通の特徴を持っている場合には，一つずつ作るのではなく，それを自分の共通の部品として何度も使用することができるのだ，という点を強調している．概念的な説明より，手続きと目的とを同時に提示する方が，初心者にはわかりやすい（そうでなければわからない）と思われるからである．

Lesson 9：自作のクラスを使う（ユーザーの入力）

[ 目標 ] テキストの入力とその評価の方法を学ぶ．

[ 解説 ] 一つのクラスを利用（継承）して，ボタンだけでなく，フィールドに答えを書き込む形式の問題を提示するアプレットを作る．これまではユーザーとのインタフェースはボタンによる選択に限られていたが，フィールドに入力する方法を学ぶことによって，ユーザーの反応をとる方法が広がり，CAIの簡単なプログラムを作成するための道具が揃うことになる．また，これによって，HyperCardで作ったものをJavaに移植することができるようになる．すでに用意されている絵を利用して，ページを増やしていくことが課題である．

Lesson 10：簡単なCAI教材を作る

[ 目標 ] 画面の乱れを抑え，表示の工夫をして各自の作品を仕上げる．

[ 解説 ] 「目的的に学ぶ」という，「Java 10 Lessons」の基本的な考えの通り，Lesson 1からLesson 9までの学習は，このLesson 10の最終目標であるJavaホームページを作るための学習である．このレッスンでは，Lesson 9までの技術を統合して一つのインタラクティブなWebページを完成させる．新しい部品としてMediaTrackerを使い，画像の取り込みを行う．学生たちのほとんどは，基本的にこれまで学習したプログラムを使うことになる．ただし，教材の表現が彼ら自身の創作であるかぎり，「自分のものになる」感覚を得ることができると期待している．

## 4．Java 10 Lessons を使用した授業の状況とその結果

### (1) 授業の実際

このレッスンは1997年の10月から12月まで，後期の12コマのうち10コマ（90分/コマ）を使用し，玉川大学の教育学科の学生（2，3年生：20名）に対して行った．はじめの2回の授業は，前期の復習をかねて，HyperTalkの練習を行った．

学生たちはすでに前期のHyperTalkの授業を受けているが，その能力にはばらつきがある．それは，彼らのプログラミングそのものの能力差もあるが，作成しようとした教材の内容にも関係している．教育学科の学生は，プログラミングそのものよりも，作品としての見栄えにこだわるために，プログラミングそのものの学習が遅れる場合が少なくないためである．

授業の形態は，はじめにプログラムとその解説をプリントにして配布し，講師（織田と筆者，主に織田）がそれをはじめに説明し，学生たちはそのプリントで示されたプログラミングにもとづく課題を実習しながら学ぶ，という形をとった．

各レッスンの内容上の問題や，学習過程での問題を把握するために，各時間ごとに学生たちは電子メールで次のよ

うな項目について，回答するように求めた．

Javaプログラミングについてわかったことと，そのきっかけ

Javaプログラミングについてわかりかけたことと，そのきっかけ

Javaプログラミングについてわからないこと

Javaプログラミングについて質問してみたいこと

Javaプログラミングについての感想

コースの最後では，仕上げとして学生たちは各自の作品を制作した．

### (2) 各レッスンの状況と学生の反応

Javaプログラミングにおける各概念や技術についての学生の学習の過程については，各レッスンに対する彼らの回答と観察によって，次のようにまとめることができる．

Lesson 1

このレッスンの目標であるJavaアプレット作成手順（コンパイル，HTMLへの組み込み）については，全員が授業の中で理解していた．しかし，コンパイルできないというトラブルがあったが，これはコンピュータの準備の不十分さによるものであった．ここでは，プログラム作成からWebページへの組み込みまでの手順を学ぶ前に，学生はあらかじめ用意されたJavaアプレットで遊んだ．用意されたアプレットは実際に自分で絵が描けるものであるため，学生は，HTMLだけで作られたホームページとの違いに気づき，Javaの可能性を理解できたようである．そして，例として見たものの他に，どのようなことがJavaによってできるのか興味を持つ者も多かった．また，このレッスンではまだプログラムには触れてはいないが，コンパイル前のプログラムを見たことが，Javaの雰囲気を感じさせている．

Lesson 2

このレッスンからJavaプログラミングがはじまる．学生たちは，画像ファイル名，表示色，文字フォントの変更といった，サンプルプログラムの部分的な変更はできるようになった．しかし，一つの命令の中の一部である引数やパラメータの意味は理解しているが，予想通り，プログラム全体の流れや，命令同士のつながりである読み込みから表示までといった手続きについては，あまり理解していないようであった．

Lesson 3

このレッスンの目標である「静止画を切り替えてアニメーションを作る」原理については，多くの者が理解できたようである．また，それにともなって，多くの学生は，配列の利用法についても理解した．for文についての疑問を持つ者はいなかった．型宣言については，Lesson2で，Font型やImage型の型宣言を学習したときには質問が出なかったが，このレッスンでのint（変数名）という型宣言について，何人かの学生が疑問を感じていた．また，色やフォントの指定や，クラス名とファイル名の関係など，以前に学習した概念や技術のいくつかを，ここではじめて納得した学生も少なくなかった．

Lesson 4 - 5

Lesson 4からスレッドを導入している．スレッドを使用するためには，準備と実行というJavaに共通のプログラミング手順の理解が必要となる．彼らにとって，スレッドの役割のうち，速度調整という役割は比較的理解しやすかったようである．また，スレッド使用時に必要な準備と実行

の手順も理解できたものが多い。しかし、スレッドを導入した目的は、動画を継続して動作させるためであったが、そこまで気づく学生はわずかであった。

Lesson 5 の目標とその下位目標については、このレッスンを終えた時点で理解できたものが多かった。Lesson 4 から学んでいたスレッドについても、Lesson 5 の時点で理解する者が多かった。また、ここで型宣言の必要性という、コンピュータ言語の専門的な部分に興味を持つ者がいた。また、Javaが、ホームページ以外でも利用できるかどうかに関心を始める者もあった。それぞれの関心の違いが現れはじめたと言える。

#### Lesson 6

このレッスンでは、「音声の再生は画像の表示と手順が同じ（似ている）ことがわかった」という比較による理解の報告が多かったが、その後、手順の違いに気がつく者が多かった。手順の異同に目をむけながら、結果としてほとんどの者が再生手順を問題なく理解した。ここで興味深かったことは、レッスンの目標となる技術（音声を動画に組み合わせる技術）の学生の理解度が、以前のレッスンに比べて大きく異なっていたことである。

このレッスンの主題は「音声」であったが、課題として、アニメと組み合わせるプログラムを作成した。この課題の達成には学生のあいだでかなりのばらつきがあったが、できた学生が援助することで、それぞれが課題を達成していた。学生の中には、プログラムの効率的な方法として、動画再生のプログラムをもっと簡単にする方法を質問するものがいた。

#### Lesson 7-9

Lesson 7では、目立ったつまずきはなかった。パス指定の問題でつまずいた者もすぐに解決できた。HyperTalkとHTMLのみの経験者にとって、Javaプログラミングにおける最も大きな壁は「クラス」の概念とその利用であると考えられる。Lesson 8ではそのクラスを学習者自らが作成し、それを部品として継承する技術を学んだ。スーパークラスとサブクラスの関係、クラス継承の概念については、学生の多くが、考え方としては理解していた。しかし、クラス継承の意義や必要性について疑問を持つ者もあった。また、クラス継承時に起きる問題についてのつまずきもみられた。

Lesson 8, 9では、インタラクティブなマルチメディア教材の基本的なプログラミングを学び、Lesson 10でさらにそれを応用するのだが、プログラムが複雑となっているために、かなりわかりにくそうであった。それは、サンプルプログラムのスーパークラスにデータが入っていたりして、サブクラスのメソッドと区別がつきにくかったなど、十分に整理されたものとなっていなかったためであると考えられる。

### (3) 全体としての考察

Lesson 10 では、学生たちはそれぞれの作品を作ったために、時間外の実習が必要であったが、それぞれが作品を作りあげた。ただ、容易ではなかったようで、ほとんどはサンプルプログラムを一部変更したものである。もっとも、それは予想していたことであり、「Java 10 Lessons」の目標が、サンプルをカスタマイズできればよいのであるから、それを不十分とする理由はない。

また、Javaが簡単ではないので、何人かの脱落者を予想し、実験にもある程度のためらいがあったが、コースを通

して脱落者もなく終了できたことは、喜ばしいことであった。はじめの反応にあるように、Javaに取り組んだ当初は難しいと感じつつも、Javaが社会的に注目されていることや、HyperCardでの教材をインターネットで実現するための道具として有効であるとはじめに説明したことが影響したのかもしれない。しかし、「難しい」と言いながらも、彼らは最後まで熱心に課題に取り組んだ。

## 5. Java 10 Lessons の展望と改良

この授業プログラムの目的は、インターネット上で利用できるマルチメディア教材を作ってみる、ということであった。学生たちがHyperCardで作ってきた、スタンドアロンで動く教材をインターネット上で実現できる可能性は、彼らにとって魅力的であったと思われる。また、全体的な見通しの中でそれぞれの部分を有意義に作り上げていくという展望のもとで、それぞれのプログラミングが理解されていることは、配列の理解を容易にしたり、try-catchの意味を容易に理解させることになったとも思われるのである。

このレッスンは、教育に携わる人々を対象として試験的に開発したものである。しかし、筆者らが思うに、このような作品は、見た目にわかりやすく、多くの初心者を取り組みやすいものである。また、インターネットでは情報発信型のページが多いので、マルチメディア教材のようなものは、一般の人たちにもその用途がわかりやすいものではないかと思う。したがって、文学部の他の学科で行われるJavaの授業や、これからJavaに取り組みたいと思っている中・高生にも利用できると考えている。また、私たちがこれを開発する関係で、Javaを学びはじめた工学部の学生たちの行動を観察していると、彼らもまた、抽象的な概念の理解にとまどっていた。それを考えると、彼らもはじめからグラフや複雑な計算に入らず、遊び的な要素を含んだマルチメディアを利用した作品の制作から入るのもいいのではないかと思う。

ところで、JDKの進化と今回の実験的な結果から、現在はずでいくつかの改良をほどこしている。一つは、Macintoshではまだ現時点ではJDK1.02しかないが、すでにWindowsでは、JDK1.1.6が出ており、その仕様も異なるからである。まもなくMacintosh用にも1.1がリリースされるであろう。ただし、「Java 10 Lessons」のレベルでは、イベント処理の違いだけなので、その対応は容易であった。もう一つはLesson 8以降の自作部品プログラムの改良であり、より部品らしく、簡明にしたことである。また、はじめのレッスンで、より達成感をもたせるために、ちょっとした応用でももしろいプログラムができることを示すサンプルを用意している。

今日の学習理論は、全体的な見通しの中での学習の意義を強調し、主体的な活動の中での学習の有効性を強調する。しかし、初心者にとっては、何を、どこからはじめるべきか、その手がかりとなるものが必要であり、それをどう与えるかが重要である。この「Java 10 Lessons」は、学生に何が得られるかを示し、それを達成する筋道を示している。このように、目的的に学べることは、特に初心者にとっては、意義のある学習となると筆者らは考えている。

#### 注

- (1) 山口栄一：目的的にプログラミングを学ぶ-HyperTalk 20Lessons の作成。日本教育工学会第11回大会講演集, pp. 377-8, 1995。