

重度視覚障害者による Java プログラミングの可能性

筑波技術短期大学 障害者高等教育センター

長岡英司

要旨：重度の視覚障害を持つ学生に Java 言語を点字教材で学習させ、プログラミングを行わせた。プログラミング環境には、スクリーンリーダの音声読み上げに加えて、点字ディスプレイ出力を導入した。その結果、重度の視覚障害者もテキストベースのプログラミング環境でならインターネット関連の初歩的なプログラムを作成できること、その過程で点字の活用が有効であること、などが明らかになった。

キーワード：重度視覚障害、Java 言語、プログラミング、点字、テキストエディタ

1. はじめに

視覚障害者によるコンピュータプログラミングがわが国で本格的に行われるようになったのは、今から約30年前である。1972年に、社会福祉法人日本ライトハウスが、諸外国の状況を参考に、職業訓練として視覚障害プログラマーの養成を始めた [1]。当時はまだパソコンはなく、バッチ処理方式の中・大型コンピュータの下で、COBOL や FORTRAN などの言語を用いてプログラミングが行われていた。

その後、80年代になるとパソコンが登場し、アクセス支援のソフトウェアや周辺機器が開発・整備されて、視覚障害者のプログラミング環境は次第に整った。80年代から90年代にかけて標準的な基本ソフトであったDOSの下で、視覚障害者によるプログラミングは、主にC言語を用いて、盛んに行われた。それによって、情報処理分野での重度視覚障害者の就業が進んだほか、視覚障害者自身によって有用な支援ソフトが多く開発された。

しかし、90年代半ばから普及した基本ソフト Windows の下では、視覚障害者がプログラミングを行うことが容易でなくなった。その主因は GUI (Graphical User Interface) である。つまり、プログラミング環境が GUI 化し、また、開発するソフトウェアも GUI 方式でなくてはなくなり、視覚を必要とする度合いが増した結果である [2]。

一方、Windows の登場で一時期後退を余儀なくされた視覚障害者のパソコン利用は、スクリーンリーダの開発と改良によって再び盛んになった。そして、インターネットの普及が、パソコンの利用をさらに促進した。DOS 時代には視覚障害者のパソコン利用の主たる目的は、墨字 (点字ではない一般の文字) を独力で書くことであった。しかし、近年は、インターネットにアクセスするための端末としての利用が中心となっている。イン

ターネットは、情報へのアクセスの可能性を飛躍的に拡大するという意味で、情報障害者といわれてきた視覚障害者にとってとりわけ重要である。情報化社会に参加するためのライフラインともいえる。それゆえ、職業や学業をはじめ日常生活のさまざまな場面で、視覚障害者がより快適かつ確実にインターネットを利用できるようにする必要がある。そのための有用な支援ソフトやアクセスツールを開発するには、DOS の時代にそうであったように、視覚障害者自身がプログラミングに関われることが重要である。

このような背景から、視覚障害者が Java 言語を用いてプログラミングを行うことの可能性を検討した。

2. Java 言語の利用 [3]

Java は、インターネット環境の下で広く用いられているプログラミング言語である。

(1) Java の歴史

Java は、米国の Sun Microsystems 社が 1991 年に開発を始めたプログラミング言語である。家電品などをネットワークに接続して制御するためのプログラムを開発する言語 Oak をベースに、技術者の James Gosling によって設計された。そして 1994 年ごろから、インターネットのワールドワイドウェブ (WWW) との関連で活用されるようになり、その後急速に普及した。

(2) Java の特徴

Java には以下のような特徴がある。これらによって Java は高く評価され、広く用いられている。

a. インタプリタ型

コンパイルによって生成される中間コード (バイトコード) が解釈・実行されるインタプリタ形式の言語であるが、C や C++ と同等の高いパフォーマンスを有する。

b. オブジェクト指向

C++ などと同様、オブジェクトベースの効率的なプログラム開発が可能である。

c. 簡便性

C++ 言語と類似の構文であるが、ポインタ機能がないなど、複雑さが排除されている。また、自動ガーベジ・コレクタ機能があるため、記憶領域の管理が不要である。

d. 分散型

プログラムは、ローカルでもネットワーク経由でも実行可能である。また、TCP/IP 用の内蔵ライブラリが用意されている。

e. 安定性

コーディングにおいて厳格な型指定が必要な上、コンパイル時と実行時に厳しいエラーチェックが行われ、バグ原因の早期の検出が可能である。

f. セキュリティ

コンパイル時と実行時にコードやファイルアクセスの厳しいチェックが行われ、安全が確保される。

g. プラットフォーム非依存

作成したプログラムコードは、複数種類のプラットフォームでそのまま動作する。再コンパイルも必要ない。

h. 移植性

他の言語と異なり、処理系に依存しない。

i. マルチスレッド

言語レベルでマルチスレッド機能をサポートしている。

(3) Java の選定理由

Java は、前述のように、C や C++ などの利点を備え、同時にそれらの難点のいくつかが解消された、使いやすい言語である。実際 Java では、従来の言語でのプログラミング経験が生かせるうえ、記憶領域の管理に関する煩雑さなどから開放される。また、機種や基本ソフト、処理系に依存しないことは重要である。開発したプログラムは、Windows をはじめとするほとんどの基本ソフトや、ネットワーク端末など Java をサポートしている機器で、そのまま動作する。さらに、インターネットへの対応が容易なことも大きな魅力である。用意された機能を利用して、WWW の下でのさまざまな処理を実現することができる。そして、視覚障害者にとっての最大の利点は、従来型のコマンドライン方式で開発を行えることである。

Java のプログラミングには、Java Developer's Kit (JDK) という開発環境を利用する。JDK は Java の開発元の Sun Microsystems 社が無償で配布している。配布されるコン

パイラとインタプリタは、いずれもコマンドラインのキー入力で起動する。また、ライブラリも、コマンドラインからアクセスできる。さらに、入出力機能が充実しており、処理結果を音声や点字に変換しやすいなど、アプリケーションプログラムへのアクセスが容易である。

現在の標準的な基本ソフトは Windows である。この GUI 方式の基本ソフトのもとで、視覚障害者には確実に利用できる開発環境がない。そのような状況にあって、Java が持つ、一般的には一時代前の古い方式の開発環境は、視覚障害者にとってきわめて貴重である。言語としての優秀性ととともに、この開発環境を持つことが、Java を利用対象に選定した理由である。

3. プログラミングの方法

非視覚的な方法での Java プログラミングについて概説する。

(1) 概要

Windows の操作やインターネットへのアクセスは、既存のスクリーンリーダーの読み上げ機能や点字変換機能を利用して行う。

JDK の操作は、主に音声出力を利用して行う。Windows には、システムとのやり取りをコマンド入力によって行う機能がある。98系までは「DOS 窓」、2000 以後では「コマンドプロンプト」と呼ばれる機能である。これらのウインドウ内の表示を読み上げさせる方法がそれぞれにあり、その下で、Java のコンパイラやインタプリタを操作する。

コーディングは、テキストベースで行い、ソースコードの入力や編集・修正には、点字表示機能付テキストエディタを用いる。

コンパイラからのメッセージやアプリケーションプログラムからの出力は、スクリーンリーダーの読み上げ機能やテキストエディタの点字表示機能を活用して読み取る。

(2) コーディング

ソースコードの入力と編集・修正には、そのために開発した、点字表示機能付テキストエディタ EB を用いる。EB にはテキストデータを情報処理用点字でリアルタイムに表示する機能がある。また、既存のスクリーンリーダーの音声読み上げ機能との併用が可能である。音声による読み上げと情報処理用点字によるディスプレイ表示を併用することにより、ソースコードの迅速で正確な入力や編集・修正が可能になる。

(3) コンパイルとデバッグ

Windows95 及び 98 系が備えている「DOS 窓」の読み上げには、専用ソフトウェアの VDM100F を用いる。これは、PC-Talker などのスクリーンリーダーの下で動作する。

一方、Windows2000 と XP の「コマンドプロンプト」は、PC-Talker などの下で動作する「コマンドプロンプトビューア」が読み上げる。また、XP の「コマンドプロンプト」については、スクリーンリーダー JAWS の読み上げ機能も有効である。

コンパイラからのエラーメッセージはこれらの音声読み上げを利用して読みとることができる。出力されたメッセージを読み返すには、VDM100F や「コマンドプロンプトビューア」のレビュー機能、あるいは Jaws の Jaws カーソル機能を用いる。しかし、エラーの件数が多い場合など、大量の出力に対しては、音声読み上げでは対応しにくい。また、出力内容によっては、音声だけでは細部まで正確に理解できない場合がある。そのような場合は、出力を点字に変換して読み取る。ただし、スクリーンリーダーの点字出力は情報処理用点字に完全には対応していないため、エラーメッセージをファイルに書き出して保存し、それをテキストエディタ EB の点字表示機能を使って読む。

そのためには、JDK のコンパイラ javac.exe の起動時に出力チャンネルの切り替えを指示する必要がある。これは、javac.exe からのエラーメッセージは、通常、標準エラー出力に書き出され、そのままではリダイレクトでファイルに書き出すことができないからである。切り替えを指示する方法は、JDK のバージョンによって以下のように異なる。

1.2 より前のバージョン：

javac -J-Djavac.pipe.output=true ソースファイル名 > 書き出しファイル名

バージョン 1.2：

javac -Xstdout ソースファイル名 > 書き出しファイル名

バージョン 1.3：

oldjavac -Xstdout ソースファイル名 > 書き出しファイル名

バージョン 1.3 環境では古い 1.2 バージョンのコンパイラを起動しなければならない。

(4) 実行

アプリケーションプログラムは、コマンドラインでイ

ンタプリタ java.exe を起動して実行する。コマンドラインは音声読み上げがサポートしており、この操作には全く問題はない。

一方、アプリケーションプログラムからの出力については、以下の通りである。

- ・標準出力には、前述の音声読み上げや点字変換で対応することができる。

- ・処理結果をファイル出力させ、実行終了後に、その出力ファイルを音声や点字に変換する方法もある。

また、アプリケーションプログラムからの入力要求や問い合わせ、そしてそれらに対する応答などは、原則として標準入出力を用いて行う。

(5) アプレットの作成と実行

Java アプレットは、独立型のアプリケーションプログラムと同様の方法で作成する。その動作確認は、テスト用のプログラムや AppletViewer を使って行うが、現状ではアプレットからの出力を音声や点字に変換することはできない。

4. 視覚障害学生によるプログラミング

視覚障害者による Java 言語の学習とプログラミングを試行した。被験者は、重度の視覚障害を持つ学生 2 名である。Java の学習には、市販図書 [4] をベースにして作成した点字教材を用いた。

(1) プログラミング環境

2 名の被験者が使用したプログラミング環境は以下の通りである。

コンピュータ：デスクトップ型パソコン（日本語 109 キーボード、液晶モニタ、外付けスピーカ）

点字ディスプレイ端末：BN-46D

基本ソフト：WindowsXP

スクリーンリーダー：PC-Talker および「コマンドプロンプトビューア」

Java 処理系：JDK1.3.1

テキストエディタ：EB（Edit in Braille）

(2) 事例 1（2002.12～2003.2）

a. 被験者

事例 1 の被験者は、筑波技術短期大学（以下「本学」）情報処理科 3 年生の A（21 歳）である。A は幼児期からの全盲で、盲学校入学時（6 歳）から点字を使用している。盲学校高等部のときにパソコンを使い始め、本学入学後に Windows を本格的に学習した。普段か

らスクリーンリーダの読み上げ機能を用いて Windows を操作しており、本実験で利用したスクリーンリーダにも精通していた。システムの設定、各種アプリケーションの利用、インターネットでの情報の収集や発信などに十分に習熟している。プログラミングについては、本学の授業で C 言語の基礎を学習した。

b. プログラミング実験

Java の学習は、点字教材のサンプルプログラムを参照する方法で、基礎的な部分から段階的に進めた。情報処理用点字の使用経験があったため、サンプルプログラムの読み取りには全く問題はなく、理解も確実にできた。また、JDK と EB の使用法は、実際にそれらを操作することによって習得し、短時間でプログラミング環境を理解した。

その結果、ファイル処理プログラムや、インターネットにアクセスして基本的なデータをやり取りする初歩的なプログラムなどを、独力で作成できるようになった。

(3) 事例 2 (2003.12 ~ 2004.2)

a. 被験者

事例 2 の被験者は、本学情報処理科 2 年生の B (20 歳) である。B は、3 歳のときから右目の視力 0、左目の視力 0.01 の障害状況にあり、小学 1 年 (6 歳) から点字を使っている。パソコンは、中学 1 年 (12 歳) の時に練習を始めた。現在は Windows の操作に習熟し、スクリーンリーダの読み上げ機能を使って、アプリケーションソフトやインターネットを日常的に利用している。プログラミングについては、本学の授業で C と ruby を学習し、簡単なプログラムを作成した経験がある。

b. プログラミング実験

Java の学習は、A と同じ点字教材を使い、サンプルプログラムを理解してから類似のプログラムを作成するという方法で進めた。また、Java のプログラミング環境で用いたスクリーンリーダが、ふだん B が使用しているものとは異なる種類であったため、その使用法の学習が必要であった。JDK と EB については、実際の操作を繰り返すことで学習した。

その結果、ファイルの処理やインターネットへのアクセスを行う初歩的なプログラムを、独力で作成できるようになった。

(4) 考察

被験者 2 名は、パソコンの使用や Windows の操作に十分に慣れているうえ、わずかながらプログラミング経験

があったため、短時間で Java 言語の基礎を理解し、実際のプログラミングを行えるようになった。

テキストベースのプログラミングでは、プログラミング環境とのインタフェースとしてテキストエディタが重要な役割を果たす。そのテキストエディタに点字表示機能を付加したことにより、音声読み上げだけの場合の非能率や不確かさが解消できた。音声読み上げと点字ディスプレイ出力を併用することにより、プログラムの正確なコーディングや、メッセージ類の迅速で確実な読み取りなどが可能になった。

本実験により、重度の視覚障害者も、テキストベースのプログラミング環境においてなら、点字を活用して比較的能率よく、インターネット関連の初歩的なプログラムを作成できることが明らかになった。

5. 今後の課題

Java で作成したアプリケーションプログラムからの出力のうち、標準出力については、音声や点字で対応できる。しかし、GUI 方式の入出力への非視覚的な対応は、現在まだ困難である。スクリーンリーダの改良や触覚図形提示装置の導入などによってこの問題の解決を図る必要がある。また、この問題に対する取り組みが進んでいる英語版の開発環境を導入してその活用を検討することも、一つの方法であろう。

一方、コーディングやデバッグの過程に点字を導入し、能率が向上したが、視覚によるプログラミングとの能率の格差はまだ極めて大きい。コンパイラ、デバッグ、エディタ、コーディング支援ツールなどを点字・音声機能の下で統合した、視覚障害者用プログラミング支援システムの開発が、今後必要である。

さらに、インターネット上に流通する様々な形態の情報に視覚障害者が対応できるようにすることも重要な課題である。そのために、マルチメディアへのアクセスを支援する方法を、早急に確立する必要がある。

6. おわりに

適切な点訳による点字教材を用意すれば、重度の視覚障害者も Java 言語を習得できる。また、テキストベースのプログラミング環境でなら、インターネット関連の初歩的なプログラムの作成が可能である。その際、点字の導入が、能率や精度の向上をもたらす。こうした事実を基盤として、今後さらなる可能性を追求することが重要である。

参考文献

- [1] 長岡英司・黒川哲宇 他：重度視覚障害を持つ情報処理技術者－わが国における養成の歴史と職務の現状．筑波技術短期大学テクノレポート4：221-224, 1997
- [2] 長岡英司：重度視覚障害者のソフトウェア開発技能の職業的有用性．職業リハビリテーション第16巻：43-51，日本職業リハビリテーション学会，2003
- [3] Sun Microsystems, Inc.：Java プログラミング講座，株式会社アスキー，2000
- [4] 北山洋幸：Java によるはじめてのインターネットプログラミング，株式会社技術評論社，2002 視覚障害者による Java プログラミング

Possibility of Java Programming by the Blind

NAGAOKA Hideji

Research Center on Higher Education for the Hearing and Visually Impaired, Tsukuba College of Technology

Abstract : The spread of GUI (Graphical User Interface) has made it difficult for the blind to develop computer programs. An experiment was performed to examine the usefulness of Java language as one of their programming tools. In the experiment, blind students were required to write tiny programs in Java. Learning materials were prepared in Braille in order that they would be able to obtain the skills of Java programming. A text editor was introduced to assist them in their programming tasks. The editor had the function of outputting the currently focused portion of text onto a braille display terminal. The result showed that Java was quite useful for the blind and that the realtime braille display function was helpful for their programming situation.

Key Words : Blindness, Java Language, Programming, Braille, Text Editor