

## プログラムリストの点字・普通文字相互自動変換

5E-1

長岡 英司

(筑波技術短期大学)

## 1. 背景と目的

わが国には独自の情報処理用点字体系がある。これは、(1)一般に用いられている点字体系が多義的かつ文脈依存的でプログラムリスト等の記述には適さないこと、(2)諸外国の情報処理用点字体系では仮名文字に対応できないうえ、国内体系との整合性の問題もあること、から定められた。ところが、全盲者やそれに近い低視力者のプログラミング場面では、プログラムリスト等の点字による入出力はほとんど行われておらず、専ら音声出力が用いられている。こうした状況は、点字入出力装置が高価なうえに、既存の装置やソフトウェアでは同点字体系での入出力が十分にできないこと、などによる。だが、一般に、音声出力のみでプログラミングを行うことには無理があり、とくにプログラミングの初学者の場合は、確実な点字入出力の併用が望ましい。

このような背景から、重度視覚障害者に対するプログラミング教育の環境の改善を目的に、プログラムリスト等を点字と普通文字との間で相互に適切に変換するソフトウェアの開発を行った。

## 2. わが国の情報処理用点字

## 2.1. 概要

日本点字委員会は、1981年に、情報処理用点字を定めた(1985年に一部改正)。この体系は、JIS X 0201の図形キャラクタ(158種)等に対する点字符号の割当(符号系)と、その点字符号の配列方法(表記規則)とからなる。付加的な符号も含め、

- ・多義性や文脈依存性はない。
- ・一般点字との符号の共通性が十分保たれている。
- ・個々の符号の触読性は概ね良好である。

Automatic Conversion of Program List between  
Braille Code and General Character Code

Hideji Nagaoka

Tsukuba College of Technology

## 2.2. 図形キャラクタ用点字符号

- (a) 点字符号の長さは1マスまたは2マスである。
- (b) ローマ小文字、ローマ大文字、数字、片仮名(長音符号を含む)は、一般点字と等しい。
- (c) ローマ文字系の記号33種(間隔を含む)のうち、17種は一般点字と等しく、他は異なるかまたは対応するものがない。
- (d) 片仮名系の記号、濁点、半濁点、及び片仮名小文字は、一般点字と異なる。

## 2.3. モードとモードフラグ

同一形状の点字符号が割り当てられている図形キャラクタを識別するために、モード切り替え方式が用いられている。モードは、図形キャラクタの部分集合(ローマ小文字、ローマ大文字、数字、片仮名)に対応して4とおりに切り替わり、点字符号の対応先は、モードによって確定する(各部分集合内では点字符号の重複はない)。モードの設定/切り替えは、4種類のモードフラグ(いずれも1マスの点字符号)で明示的に行うほか、暗黙の設定/切り替えもある。暗黙の設定/切り替えでは、必ず初期設定状態(ローマ小文字かローマ大文字のどちらかに予め宣言しておく)になる。

## 2.4. 使用上の問題点

現体系の不備や不明確さなどから、次のような問題が生じている。

- ・モードフラグの使い方がまちまちで、欠落や過記載が多く見られる。
- ・点字1行のマス数が少ないために生じる行の折返しによって、読みにくくなる場合が多い。
- ・行の折返し箇所空白がある場合に、その有無や個数を読み取れない。
- ・2バイトコードの箇所で書式が乱れる。

## 3. 相互変換ソフトウェアの開発

## 3.1. 情報処理用点字体系の明確化と補強

(a) 体系の論理的再構築：

(a.1) ローマ文字系の記号は初期設定状態に対応するキャラクタ集合に属するものとした。

(a.2) 現モードと次点字キャラクタ（1マスの点字）とによって、その点字キャラクタの位置におけるモードが確定するものとした。

(a.3) 点字キャラクタ（64種）によるモードの遷移を、モードフラグの出現頻度等を勘案して定めた。

これらによって、すべての場合に対するモードフラグの要否とその種類が決まる。

(b) 行の折返し：

(b.1) 二つの図形キャラクタ間での行折返しの可否を、キャラクタの組合せごとに定めた。

(b.2) 折返し後の書き出し位置を定めた。

(b.3) 行折返し箇所にある空白の表記法を定めた。

(c) 2バイトコードは、専用の2マス符号、または前置符付きの16進表現で表記するものとした。

### 3.2. 点字の内部コード

点字入出力機器のいずれもが対応している「北米点字コンピュータコード」を用いることとした。

### 3.3. 普通文字から点字への変換

#### 3.3.1. 変換処理

次の表を用いて変換処理を行う。

(a) キャラクタ属性表：変換元の1バイトコードを、モードとの関係や結合特性で13とおりに分類。

(b) コード変換表：各図形キャラクタに対する点字符号（1バイトまたは2バイト）。

(c) モード遷移表：変換元の次キャラクタの属性による変換先のモードの遷移。

(d) モードフラグ表：変換先の現モードと変換元の次キャラクタの属性との組合せに対するモードフラグの要否とその種類。

(e) 行折返し可否表：変換元キャラクタ間の結合強度（属性ごと）。

#### 3.3.2. 変換ソフト“HTOCB”の機能

点字出力のページ・行長を設定できるほか、初期設定状態、行番号の付加、2バイトコードの変換形式、行折返し箇所の空白の表現などを指定できる。

### 3.4. 点字から普通文字への変換

#### 3.4.1. 変換処理

次の表を用いて各点字符号を変換する。

(a) 点字キャラクタの属性表：点字キャラクタを用途等によって15とおりに分類。

(b) モード遷移表：変換元の次点字キャラクタの属性による変換元のモードの遷移。

(c) 点字符号変換表：点字符号とモードとの組に対する図形キャラクタ。

#### 3.4.2. 変換ソフト“CBTOH”の機能

初期設定状態の指定、行番号やページ番号行の読み飛ばしの指定などが可能なほか、16進形式の表記を2バイトコードに変換することができる。

### 3.5. 変換精度の評価

(a) C言語で記述された“HTOCB”のソースリスト（2バイトコードを含む28562バイト）を点字変換して（29559バイト）触読検査した結果、誤りのない完全な点字リストであることが確認できた。

(b) 同点字リストを逆変換して元のリストとバイト単位で比較した結果、完全に一致した。

### 4. 活用効果

筑波技術短期大学情報処理学科の点字使用学生を対象に本ソフトウェアを用いた。

#### 4.1. 点字変換

5人の学生にC言語のプログラムリストを点字変換して提示し、伝達精度を調べたところ、全員が、記述内容やレイアウトを正確に読み取れた。また、授業における活用で、行の折返しが適切で読みやすい、モードフラグに誤りがなく正確に読み取れる、レイアウトが良くわかる、などの評価を得た。

#### 4.2. 相互変換

点字エディタで作成・編集したプログラムリストを普通文字に変換して言語処理系に入力し、その出力を点字変換するという方法を、プログラミング実習に導入した。その結果、音声出力のみを用いた場合と比べて、作業の能率や精度で向上が見られた。

<参考文献> 「日本の点字」第9・10・13号、日本点字委員会

・本研究は平成5・6年度文部省科学研究費一般研究Cの補助によるものである（課題 0568-0322）。