

プログラムリストの点字出力

情報処理学科 長岡 英司 一般教育等 黒川 哲字

要旨：重度視覚障害者がプログラミングを行う際には、プログラムリスト等が点字で出力されることが望ましい。そこで、JIS X 0201の1バイトキャラクタからなるテキストを情報処理用点字による表記に変換する手順を確立し、それに基づいて二つの点字出力ソフトウェアを開発した。それらのソフトウェアの試用実験を行うとともに実際の教育場面でも活用し、有効性を確認することができた。

キーワード：視覚障害、プログラムリスト、点字出力、情報処理用点字、JIS X 0201

1.点字出力の必要性

近年、重度視覚障害者によるコンピュータの利用が盛んになってきた。これは、聴覚や触覚を用いて非視覚的にコンピュータにアクセスするための技術が開発され普及した結果である。しかし、現在のアクセス手段はまだ満足できるものではない。例えば、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これまで専ら音声出力が使用されてきたが、音声出力のみに依存するプログラミング環境にはいくつかの問題があり、とりわけプログラミングの初学者にとっては深刻である*1。

その改善には、プログラミングの過程に確実な文字を介在させることが不可欠であり、それゆえ点字出力の併用が必要である。

2.情報処理用点字

わが国で一般に用いられている点字の体系（『日本点字表記法』）は、(i)文脈によって表記が変わること（意味依存性）、(ii)同一の記号（列）に対して複数の解釈がありうること（多義性）、などから、コンピュータとのコミュニケーション手段としては適さない。そこで、日本点字委員会は、1981年に『コンピュータ用言語の6点式点字表記』を定めた。これは、コンピュータのプログラムや画面表示などを記述する際に用いる情報処理用点字の体系を規定する表記規則である。同表記規則は、その後1985年と1995年に一部改定されて現在に至っている*2。

2.1 対象字種

情報処理用点字で記述できる文字や記号は、表1に示すJIS X 0201の図形キャラクタ157種と間隔の計158種の1バイトキャラクタである。図形キャラクタは、以下の五つのキャラクタ集合に分類される。

- ・ローマ小文字(英字の小文字)

- ・ローマ大文字(英字の大文字)
- ・特殊記号(英字系の記号)
- ・数字
- ・仮名(カタカナと関連記号)

2.2 点字記号

一つないし複数の点字マス内の点の組み合わせパターンを「点字記号」という。情報処理用点字において各1バイトキャラクタを表す点字記号を表1に示す。その要点は以下のとおりである。

- ローマ小文字、ローマ大文字、数字、片仮名（長音符を含む）は、一般点字と等しい。
- 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。
- 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 状態変更フラグ

情報処理用点字には、同形の点字記号を使い分けるための「状態変更フラグ」があり、その使い方が厳密に定められている。

状態変更フラグは、後に続く点字記号の列が、「どのキャラクタ集合の文字や記号を表すのか」（これを「状態」という）を明示するものであり、小文字フラグ、大文字フラグ、数字フラグ、仮名フラグの4種がある。

- 状態変更フラグが記されない場合の暗黙の状態を「基本状態」という。基本状態は、小文字状態と大文字状態のいずれかに設定することができる。どちらの場合でも、改行やマスあけなどによって状態が暗黙に基本状態に戻る。
- 特殊記号は、基本状態に対応するキャラクタ集合に属することとなっている。
- 仮名に特殊記号が続くときは、通常の状態変更フラグ

表1 JIS X 0201 キャラクタの点字記号と属性

キャラクターコードの 上位=ブル キャラクターコード の下位=ブル	2	3	4	5	6	7	A	B	C	D
0	SP CC	0 NC	@ S3	P U2	 S3	p L2		ー KC	タ KC	ミ KC
1	! S1	1 NC	A U1	Q U2	a L1	q L2	 KC	ア KC	チ KC	ム KC
2	" S4	2 NC	B U1	R U2	b L1	r L2	「 KC	イ KC	ツ KC	メ KC
3	# S1	3 NC	C U1	S U2	c L1	s L2	」 KC	ウ KC	テ KC	モ KC
4	\$ S2	4 NC	D U1	T U2	d L1	t L2	、 KC	エ KC	ト KC	ヤ KC
5	% S2	5 NC	E U1	U U2	e L1	u L2	・ KC	オ KC	ナ KC	ユ KC
6	& S1	6 NC	F U1	V U2	f L1	v L2	ヲ KC	カ KC	ニ KC	ヨ KC
7	' S4	7 NC	G U1	W U2	g L1	w L2	ア KC	キ KC	ヌ KC	ラ KC
8	(S1	8 NC	H U1	X U2	h L1	x L2	イ KC	ク KC	ネ KC	リ KC
9) S2	9 NC	I U1	Y U2	i L1	y L2	う KC	ケ KC	ノ KC	ル KC
A	* S3	: S2	J U1	Z U2	j L1	z L2	エ KC	コ KC	ハ KC	レ KC
B	+ S4	;: S2	K U2	[S1	k L2	{ S1	オ KC	サ KC	ヒ KC	ロ KC
C	、 S2	< S1	L U2	¥ S1	l L2	 S3	ヤ KC	シ KC	フ KC	ワ KC
D	- S4	= S3	M U2] S2	m L2	} S2	ユ KC	ス KC	ヘ KC	ン KC
E	. PD	> S2	N U2	^ S1	n L2	~ S3	ヨ KC	セ KC	ホ KC	マ KC
F	/ S3	? S2	O U2	 S3	o L2		ッ KC	ソ KC	マ KC	ン KC

L1:"a"~"j" L2:"k"~"z"
 U1:"A"~"J" U2:"K"~"Z"
 S1: 前置的記号 S2: 後置的記号 S3: 独立的記号 S4: 従属的記号
 PD: ピリオド (小数点)
 NC: "0"~"9"
 KC: カナ及び関係記号
 CC: 間隔及び制御コード
 OC: その他 (2バイト文字の1バイト目など)

ではなく、仮名・記号遷移符を用いる。仮名・記号遷移符は、状態を基本状態に戻す。

3.点字出力ソフトウェアの開発

3.1 点字への変換処理

点字出力を行うためには、プログラムリスト等のキャラクタ列を点字記号の列に変換する必要がある*3。変換先の点字記号列の内部表現には、NABCC（北米点字コンピュータコード）を用いることとした。変換はキャラクタ列の先頭から順次進め、JIS X 0201図形キャラクタと間隔は対応する点字記号に変換して、必要な箇所状態変更フラグを挿入する。また、変換元の2バイトキャラクタは識別記号を前置した16進コードまたは新設の2マス記号に変換し、制御コード等は無変換のまま、点字記号列に加える。この変換を行うために以下の表を作成した。

- a)キャラクタ属性表：変換元の1バイトコードを、キャラクタ集合や結合特性で13通りに分類（表1参照）。
- b)コード変換表：各図形キャラクタに対する点字記号（表1参照）。
- c)状態遷移表：変換元の次キャラクタの属性による変換先の状態の遷移（表2参照）。
- d)状態変更フラグ表：変換先の現状態と変換元の次キャラクタの属性との組合せに対する状態変更フラグの要否とその種類（表2参照）。

3.2 点字プリンタ用ソフトウェア“HTOCB”

本ソフトウェアは、パーソナルコンピュータPC-9801・9821シリーズ及びその互換機のMS-DOS上で動作し、NABCC対応の点字プリンタ（全機種）に対して有効である*3。

(1) 基本的な変換機能

JIS X 0201の1バイトキャラクタからなるテキストファイルを、情報処理用点字の記号列に変換し、点字プリンタに出力する。入力をキーボード、出力をNABCCファイルまたはスクリーンにすることもできる。

(2) 書式制御機能

変換先点字リストのページ書式（1ページの行数と1行のマス数）を指定できる（ページ番号行を付加）。また、行の折り返しが適切になされるよう、行折返し可否表を用いて各文字や記号の結合属性で制御する。

(3) その他の機能

- a)基本状態を大文字状態にすることができる（デフォルトは小文字状態）。
- b)変換元ファイルの各行の番号を行頭に付加することができる。
- c)ページ番号行にページヘッダを付けることができる。
- d)2バイトコード文字を16進数に変換（識別記号を前置）することができる（デフォルトは、2バイトコードを示す2マス記号）。

3.3 エディタソフトウェア“TEBO”

TEBOは、点字でリアルタイムにテキストを確認しながら入力・編集作業ができるエディタソフトウェアである*4。パーソナルコンピュータPC-9801・9821シリーズ及びその互換機のMS-DOS上で機能し、点字ディスプレイ装置パワーブレイル40を必要とする。

(1) パワーブレイル40の概要

パワーブレイル40(PB-40)は、8点40マスのピン表示部を持つ点字ディスプレイ装置であり、シリアルポートとパラレルポートを介してパソコンなどとの間でデータの送受信を行う。

表2 キャラクタによる状態の遷移

状態 移行 の属性	L	U	N	P	K	Z
L1	L	(L)	(L)	(L)	(L)	(L)
L2	L	(L)	△	△	(L)	(L)
U1	(U)	U	(U)	(U)	(U)	(U)
U2	(U)	U	(U)	(U)	(U)	(U)
Sn	R	R	R	R	△	(R)
PD	R	R	P	R	(R)	(R)
NC	(N)	(N)	N	N	(N)	(N)
KC	(K)	(K)	(K)	(K)	K	(K)
CC	R	R	R	R	R	R
OC	Z	Z	Z	Z	Z	Z

<状態> L:小文字状態 U:大文字状態
N:数字状態 P:小数点状態 K:カナ状態
Z:2バイトキャラクタ状態
R:基本状態（LまたはU）

<キャラクタの属性> L1~OC(表1参照)
☆ 三角はカナ・記号遷移符が必要なことを意味する。円・弧は、状態変更フラグが必要なことを意味する。円弧の位置は、下表のように状態変更フラグが必要な「場合」を示す。

基本状態	2バイトコードの16進出力	オン	オフ
小文字		()	()
大文字		()	()

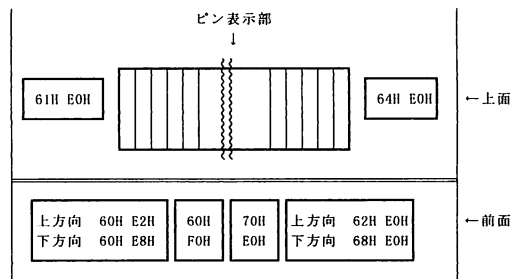


図1 操作キーの押下による送信コード

表3 カーソルスイッチ押下時の送信コード

	バイト位置	内容	
ヘッダ部	0	00H	} 固定 (16進)
	1	08H	
	2	09H	
	3	00H	
	4	00H	
	5	00H	
スイッチデータ部	6	00H	
	7	SG1	
	8	SG2	
	9	SG3	
	10	SG4	
	11	SG5	

カーソルスイッチ: 押下されているスイッチに対応するビットが1、他は0

表4 カーソルスイッチ解放時の送信コード

バイト位置	内容	
0	00H	} 固定 (16進)
1	08H	
2	09H	
3	00H	
4	00H	
5	00H	
6	00H	
7	00H	
8	00H	
9	00H	
10	00H	
11	00H	

PB-40の40マスの表示部に点字を表示するには、ヘッダ部と点字データ部からなる86バイトのデータをパソコンなどから送らなければならない。TEBOは、NABCCからPB-40固有の点字コードへの変換を行う。PB-40はデータを受信する度に、表示を更新する。

一方、PB-40からの送信は、

- a) 操作キー（図1）を押下すると、図中の2バイトのコードが1回送信される。
- b) 40個あるカーソル・スイッチを押下すると、その間0.1秒ごとに表3の12バイトのコードが送信され、スイッチが放されると表4の12バイトのコードが1回送信される。

(2)TEBOの概要

TEBOは、点字ディスプレイへの点字出力機能を備えたテキストエディタである。処理対象は、現在のところ、1バイトキャラクタのみからなるテキストに限られている。

- a) コマンドやテキストの入力は、通常のキーボードから行う。機能キーの働きは、一般的なエディタのそれとほぼ等しい。
- b) 編集は、テキスト上でカーソルを移動しながら行いいわゆるスクリーン方式である。
- c) 点字ディスプレイには、カーソルが現在位置している部分のテキストがリアルタイムで表示される。表示されるのは情報処理用点字であり、カーソルは、ピン表示部の7・8の点で示される。表示部が40マスであることから、一時に表示されるのは最大40キャラクタであり、行内の非表示部分を読むためには、PB-40のキー操作で

表示窓を移動する必要がある。

d) PB-40からカーソルの移動ができる。また、タッチ・カーソル・スイッチを押下すると対応する点字マスにカーソルが瞬時に移動する。この機能は、とりわけ有効であり、迅速な操作を可能にし、処理能率を向上させる。

4.開発成果の活用

4.1 プログラムリストの触読実験

C言語のプログラムリストをHTOCBで点字出力し、筑波技術短期大学情報処理科の点字使用学生5人に提示して伝達精度を調べた*3。その結果、記述内容（2バイトキャラクタは除く）や記述形式が正確に読み取られることを確認した。また、以下のような評価を得た。

- a) 変換元のレイアウトが良くわかる、
- b) 行の折り返しが適切になされている、
- c) 無駄なフラグがなく読みやすい、
- d) 大文字と小文字を正確に区別できる、

4.2 点字出力と音声出力の比較実験

被験者6人に、パソコン上で音声出力と点字出力を用いて等価なテキスト処理をさせ、処理の速度や精度、処理実感を調べた*4。点字出力には、エディタソフトウェアTEBOを使用した。処理内容は、1行内にある三つの文字列（英数字、記号）の組の中から異なる一つを見だし他の二つと同じになるよう修正をして確認をすることの反復であり、音声出力のみでの処理と点字出力のみでの処理を分けて行った。

その結果から音声出力と点字出力を比較すると、

- a) 処理能率は、平均的には点字の方がやや速いが大差はない。
- b) 心的負担は点字出力がはるかに少ない。
- c) 処理内容による違いがあり、発音読みをする単語や比較的短い文字列は音声出力の方が迅速で処理しやすい。一方、発音読みをしない綴りや記号を含む文字列、数字列などで比較的長いものは、点字出力が確実に有効である。

このようなことから、重度視覚障害者のプログラミング環境には、音声出力に加えて点字出力を導入することが望ましいと結論できる。

4.3 教育場面での使用効果

全盲の教官である筆者は、プログラミング実習の授業でこれらのソフトウェアを学生のためと自らのために使用している。重度視覚障害を持つ学生に点字出力を提示することは、プログラミングにおける能率や精度の向上の大きな助けとなっている。また、筆者は学生が作成したプログラムを点字で円滑かつ確実に確認し、適切な指

示や助言を与えることができる。

参考文献

- 1)長岡英司：「重度視覚障害者に対するコンピュータ・プログラミング教育の現状と課題」,日本特殊教育学会第31回大会発表論文集,pp86-87 (1993).
- 2)長岡英司:「情報処理用点字のてびき」,第1版 (1997)、社会福祉法人視覚障害者支援総合センター.
- 3)長岡英司：「プログラムリストの点字普通文字相互自動変換」,情報処理学会第50回全国大会講演論文集(1),pp61-62 (1995) .
- 4)長岡英司・黒川哲宇:「点字使用者のためのプログラミング環境の整備に関する研究」,平成7-8年度文部省科学研究費補助金(基盤研究C)研究成果報告書(1997) .