

点字電子ファイル校正支援システム

情報処理学科 柴田 進吉

要 旨：点字コードで書かれた電子ファイル（フロッピーディスクなど）の校正が点字の読みとり、文法の習熟度が低くても効率的に行えるように、特殊記号、特に数学記号などを分かりやすく表示するプログラムを作成した。

キーワード：点字、点字コード変換、視覚障害教育支援

点字電子ファイルの作成

現在、点字教科書、資料の作成には大別して、(1) 原書をOCRまたはキーボード入力により、漢字かな混じり文の電子ファイルを作成し、それを専用ソフトウェアにより分かち書きされた仮名文に変換し、さらに仮名点字変換をして点字プリントをつくる。(2) 最初から点字ワープロなどで点字入力により点字電子ファイルを作成し、それを点字プリンタで打ち出す、などの方法がある。点字は1列3点が2列並んだ6点で1文字を構成しているの、64文字ですべてを表現する必要から組み合わせによる特殊記号、意味の解釈が多い。通常ワープロでは点字規則にある特殊文字、数学記号などは入力できないので、点字入力、編集ができるソフトウェア（点字エディタ）を用いている。

点字電子ファイルの校正

点字電子ファイルの内容をみるためには、点字符号による再表示、点字仮名変換をして表示する等の方法がある。晴眼者が点字ファイルの内容を見る、あるいは校正をする場合は点字符号に高度に習熟した者を除けば、点字より仮名文字表示、実記号表示が一般的に効率がよく、誤りも少ない。しかし現在一般に利用されている点字エディタで点字電子ファイルを仮名表示すると後述べるように、1:1の対応をするもの以外は間違った表示をするか、あるいは点字記号をそのまま表示する。そこで特殊記号、数学記号なども意味が判る表示をして校正を効率化する方法を試みた。

点字電子ファイルは多くの点字プリンタが対応しているNABCCコード（北米点字コード）で作られていることが多いので、ここで対象としたファイルもNABCCコードである。

点字の仮名、英字、記号表示の問題点と対応

点字は英字の始まり、数字の始まりはそれぞれ英字符、

数符をつけることになっているが前後の意味関係で判別できるときは省略してよいなどということもあり機械的な変換が難しい。点字文法が多義性に対応するためには知的辞書が必要であるがそれでも完全を期することは困難である。そこで最初の試みとして、成るべく原文の意味を再現することを優先するが、英字符号の無い英字とかなの区別は文脈判定をしないと無理なので、仮名優先モードと英字優先モードを用意し、英字モードのときは英字符がなくても英字として読むことにし、判定は人間に任せることにした。

点字の表現について

点字は1列3点が2列並んだ、6点で1マスを構成している。向かって左上から下に1, 2, 3の点、2列目の上から4, 5, 6の点と呼んでいる。ここではこの文章を点字化するときのことを考えて、また短く表現するために次のようにする。

1, 2, 3の点をそれぞれ 2^0 , 2^1 , 2^2 に対応させ8進法で表現する。4, 5, 6の点も同様とし、一つの点字を（1列目の8進数, 2列目の8進数）で表現することにする。例えば1の点と6の点の点字は、(1, 4)とする。

NABCC—かな変換の例

以下にいくつかの例を示す。

1) は原文, 2) と3) は代表的な点字エディタのかなモードと英字モード, 4), 5) は本方式のかなモードと英字モードである。

変換例1.

1) えんの めんせき

=はんけい×はんけい×3. 14

2) (えんの めんせき)

— (はんけい) か (はんけい) か(4,7)3. 14

- 3) 省略
- 4) エンノ メンセキ
= ハンケイ × ハンケイ × 3. 1 4
- 5) 省略

変換例 2.

- 1) がんり ごーけい
= がんきん × (1 + りりつ × きかん)
- 2) (がんり ごーけい)
-- (がんきん) か(6,4) (4,7) 1 ? (りりつ) か (きかん) ん
- 3) 省略
- 4) エンノ メンセキ
= ガンキン × (1 + リリツ × キカン)
- 5) 省略

変換例 3.

- 1) 1 0 g₁₀ 0. 0 3 ≡ - 2. 4 7 7 1
- 2) にたれ(0,4) (2,0) ん 0. 0 3 (0.2) (2,2) _ 2 (4,0)
(1,3) れれあ
- 3) 1 0 g (0,4), (4,6) 0. 0 3 (0,2) (2,2) _ 2 (4,0)
d g g a
- 4) ニタレ) 0. 0 3 ≡ - 2. 4 7 7 1
- 5) 1 0 g. 1 0. 0. 0 3 ≡ - 2. 4 7 7 1

5) の”。” は下付きを示すために用いたが、小文字または下付きの制御記号に置き換えることもできる。

変換例 4.

- 1) ∂² z / ∂ x ∂ y
- 2) けきまや(6,4) けふけむん
- 3) (3,5) z (4,1) (3,5) (3,4) z (4,1) ” (3,5) x (3,5) y ”
- 4) ケキマヤケフケムケ
- 5) (ラウンド d) ^ 2 z / ((ラウンド d) x (ラウンド d) y)

仮名モードでは全く意味が判らないが、英字モードでは明瞭になる。 ^ 2 は 2 乗の意味である。

変換処理プログラムの概要

1. 点字電子ファイルの読み込み

通常フロッピーディスクに点字に対応した N A B C C コードがアスキー文字形式で書かれているので、これを 1 行分づつ読みだし、配列の各要素にしまう。

2. 文字変換, 解釈

仮名モードと英字モードがあるが、仮名モードのなかの英字、数字モードと英字モードの中での数字モードを区別するために、主モードと副モードを用意して意味の解釈を行っている。

配列の先頭から 1 文字づつ取りだし、仮名モードでは仮名辞書、英字モードでは英字辞書を参照し、一意的に対応しているものか、前後関係で決まるものかを判別する。仮名モード、英字モードのなかで数字記号ができたときは数学辞書を参照する。以下に判別のための表とそれに基づいた C 言語のプログラムの一部を示す。

仮名モード対応表の一部

文字	点字	NABCC	辞書	表示記号
ワ	(4,0)	39	220	
ヲ	(4,2)	57	63	@* (57,57,20) - (57,55) (57,35) (57,48) < (32,57,57) 上記以外は ヲ
数符	(4,7)	35	63	数学モードへ移る。

注)

NABCC は 1 0 進数、辞書はアスキーコードの 1 0 進数、(a, b, c), (d, e) はその文字の前後の点字 N A B C C コードを示す。

判別プログラムの一部 (C 言語)

```
kanahen()
{
    int a=0, i;
    int jim2, jim1, ji0, ji2, ji3, ji4

    ji0=buff [k] ; ji1=buff [k+1] ; ji2=buff [k+2] ;
    jim1=buff [k-1] ; jim2=buff [k-2] ;
    a = ji0 - 32 ;
    if(kandic [a] == 63) goto sagasu ;
    c = &kandic [a] ;
    strncat (kantxt, c, 1) ;
    return ;
}

sagasu :
switch (ji0)
{
```

```

case 57 :
if(ji1= =57 && ji2= =20)
    {strcat(kantxt, " @*" ); k=k+2; break; }
if(ji1= =35 || ji1= =48 || ji1= =55)
    {strcat(kantxt, " -" ); k=k+1; break; }
if(jim1= =32 && ji1= =57)
    {strcat(kantxt, " <" ); k=k+1; break; }
strcat(kantxt, " ヲ" );
break ;

```

```

case 35 :
decipt=0 ;
psubmode=" suuji" ;
break ;
.....
.....

```

N A B C C コード a に対応する場所の辞書の内容 (kandic [a]) が特別な番号 (この例では 6 3) であれば更に前後関係を調べるswitch文で個々のコードの前後関係を調べて解釈する。特別な番号以外はそれ (kandic [a]) が対応するアスキーコードになっている。

3. 変換結果の表示

上記の例にあるように記号類は通常のプリンタで表示できるように一般の記号, 文字を使用した, フォントをもつプリンタであればより原文に近い表示が可能である。

また墨字テキストには不要でも点字テキストには付加する必要なコードがある。正確な校正にはこれらのコードも表示しなければならない。今回は表示が煩雑になると考え省略したが, 追加する予定である。

課 題

変換の方法の有効性を確認したが, 実際に校正に供するにはテキストの訂正, 追加などの編集機能が必要である。

点字電子ファイルの変換を考えた動機のひとつは点字使用者が点字ディスプレイがなくても音声読み上げで内容が理解できるようにしたいと思ったからである。そのためには音声に適した変換が必要で, 上付き, 下付き文字などの例からも明かなように校正用とは異なる音声向きの変換が必要であるがこれも辞書の記号を変更するだけでよいので特に問題はない。早い機会に実現したいと考えている。