

1 文字の表現

1.1 文字コード

文字を表現する文字コードには、いくつかの種類があります。アスキー（ASCII）コードは、英数字を表現するための文字コードです。英数字は、文字数が少ないため、256通りの情報を表現できる1バイトで文字を表現することができます。10進数で表現すれば、0から31までの数値で改行やタブといった特殊な文字を表します。32がスペースを表します。33から126までの数値が英数字に対応します。例えば97という数値は小文字の「a」を表します。49という数値は「1」という文字を表します。127はデリート（削除）を表します。

アスキーコードは、1バイトで文字を表現します。このような、1バイトで文字を表現する文字コードは、1バイトコードと呼ばれます。1バイトの8個のビットのうち、一番頭のビットは、最上位ビットと呼ばれますが、アスキーコードでは、最上位ビット以外の7ビットが文字の表現に利用されます。このため、アスキーコードで表現できる文字数は、2の7乗で128文字になります。このように1バイトのうち、7ビットしか文字の表現に使わない文字コードは、7ビットコードと呼ばれます。

アスキーコードで使用しない最上位ビットを利用することで日本語のカタカナを表現することができます。これは、最上位ビットが1である場合に表現できる128から255までの128文字分を使ってカタカナ等を表現するものです。通常の日本語の半分の幅で表示されることが多いことから、通称、半角カタカナと呼ばれます。ただし、コンピュータによっては、半角カタカナを正しく扱えない場合もあることから、半角カタカナを使う場合は注意が必要になります。

次に、漢字の文字コードを見てみましょう。日本語は、ひらがな、カタカナ、漢字など文字数も多く、1バイトでは、すべての文字を表現することができません。このため、日本語を表現するための漢字コードでは、通常、1文字あたり2バイトで文字を表現します。このように、2バイトで文字を表現する文字コードは、2バイトコードと呼ばれます。2バイトコードでは、16ビット、すなわち16個の01で表現できる、約6万5千文字を表現することができます。このような漢字コードには、JISコード、シフトJISコード、EUC-JPがあります。

JISコードは、文字を表現するのに7ビットのみを使います。JISコードでは、3文字の特別な文字（エスケープシーケンス）を使って、数値が英数字を表しているか、日本語を表しているかを判断します。具体的には、「ESC (B) または「ESC (J) の後の数値は、英数字であると解釈します。また、「ESC \$ @」または「ESC \$ B) の後では日本語であると解釈します。ここで、例えば、48 33という2バイトのデータがあった場合を考えてみましょう。エスケープシーケンスで、英数字として解釈すると指定された場合は、「0!」と解釈します。一方、エスケープシーケンスで、日本語として解釈すると指定された場合は、「亜」という漢字であると解釈します。

JISコードでは、エスケープシーケンスを見落とした場合、それぞれのバイトを英数字として解釈するか、日本語として解釈するかを誤り、文字化けを起こす場合があります。シフトJISコードでは、JISコードでは使用されていなかった最上位ビットも使うことで、エスケープシーケンスを使用しないように修正されています。JISコードの数値を、部分的にシフトさせて作られています。1バイトのうち8ビットを使う、8ビットコードになります。日本語を表現する場合は、第1バイトの最上位ビットが1になるように修正されています。このため、第1バイトを見ただけで日本語であるか英数字であるかが、分かるようになっています。

EUCは、主にUNIXと呼ばれるOSで利用されます。OSは、機械部分を直接制御するソフトウェアで、WindowsやmacOS、Unixなどは、OSにあたりません。日本語を表すEUC-JPは、シフトJISと同様に8ビットコードになります。

ここまでお話ししてきた漢字コードで扱うことができるのは、基本的に、英語と日本語になります。これに対して、日本語や英語、中国語や韓国語といった多数の言語を1つの文字コードで表現するための文字コードとして、Unicodeがあります。Unicodeは、プログラム言語のJava言語などで標準の文字コードとして利用されています。また、多言語を扱えることからWebページを作成する場合などにも利用されます。

1.2 文字の表示

次に、コンピュータ上で、文字をどのように表示するかについて見てみましょう。文字コードは、それぞれの文字を数値として表現するものです。しかし、コンピュータ上で実際に文字を表示するためには、それぞれの文字に対応する文字の形状やデザインのデータが必要になります。このような文字の表示に用いられるデータがフォントになります。

フォントにはビットマップフォントとアウトラインフォントがあります。ビットマップフォントでは、文字を点の集まりとして表現します。図1-1(a)がビットマップフォントの例です。ビットマップフォントは、文字の輪郭がなめらかではありません。しかし、コンピュータ上で、表示を高速に行えるという利点があります。輪郭部分に濃淡をつけることで、輪郭部分を、より滑らかに見せることができます。このような処理は、アンチエイリアス処理と呼ばれます（図1-1(b)）。

図1-1(c)は、アウトラインフォントの例です。アウトラインフォントでは文字の輪郭を代表する点と、それらを結んだ線により文字の形状を表現します。点と点との間は、滑らかな線で結びます。実際に表示する際は、輪郭線の中を塗りつぶす処理が必要となるため、ビットマップフォントに比べて、表示のための処理は重くなります。しかしながら、文字の輪郭がなめらかで、文字をきれいなまま拡大、縮小できるという利点があります。



図1-1 ビットマップフォントとアウトラインフォント

フォントには、欧文を表示するための欧文フォントや、日本語を表示するための和文フォントなどがあります。図1-2(a)に示しているのは、欧文フォントの例です。欧文を表示するための欧文フォントをデザインの点で分類すると、セリフ系フォントや、サンセリフ系フォントなどに分類できます。セリフ系フォントには、セリフと呼ばれる文字の飾りが付いています。サンセリフ系フォントは、文字の飾りであるセリフが付いておらず、文字の線幅はほぼ一定です。これ以外のフォントとしては筆記体風のフォントなどがあります。

図1-2(b)は、和文フォントの例です。和文フォントはデザインによって分類すると、明朝体、ゴシック体などに分類できます。明朝体にはウロコと呼ばれる文字の飾りが付いています。ゴシック体には文字の飾りであるウロコが付いておらず、文字の線幅がほぼ一定です。これ以外のフォントとしては、筆で書いたような楷書体などのフォントがあります。和文フォントと欧文フォントを混在して使う場合には、デザインの相性が良いことから、通常、明朝体とはセリフ系のフォント、ゴシック体とはサンセリフ系のフォントが使われます。



図1-2 フォントのデザイン

フォントの字幅に関しては、和文フォントでは、一般に文字幅が一定の等幅フォントが用いられます。一方、欧文フォントは、例えば、WとIでは文字幅が違ふことから、通常、文字によって文字幅が変わるプロポーションアルフォントが使われます。また、Aの後にVが来る場合など、文字の組み合わせによっては、文字の間隔が空いて見える場合があります。このため、文字の組み合わせによっては、カーニングと呼ばれる処理が行われます。これは、文字の組み合わせによって、文字間隔の微調整を行う処理です。以上のようなフォントの情報を用いることで、コンピュータ上で文字を表示することができます。

1.3 文書データの記述

文書データを扱うためのソフトウェアとしては、テキストエディタや、ワープロソフト、DTPソフトなどがあります。テキストエディタは、テキストデータの編集に用いられます。ワープロソフトでは、文字の大きさや、色、レイアウト等の編集も行うことができます。コンピュータを使った出版物の編集は、DTP (Desktop Publishing) と呼ばれます。ワープロソフトやDTPソフトでは、通常、編集中にコンピュータ上に表示されるページのイメージがそのまま印刷イメージになります。これは (What You See Is What You Get) の頭文字をとって、WYSIWYGと呼ばれます。

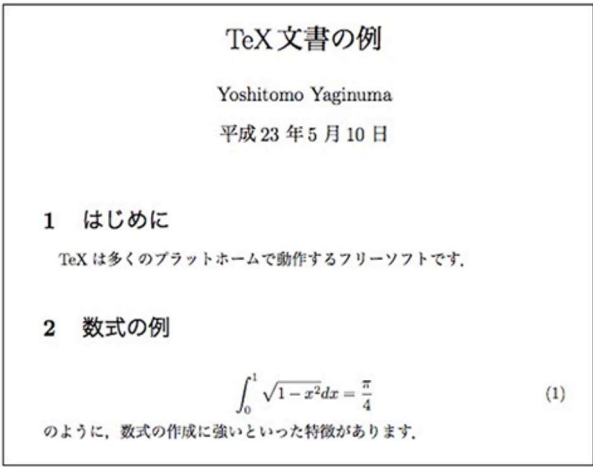
文書データの保存形式としては、さまざまなものがあります。PostScriptでは、文字や図形がベクタデータ (すなわち、図形の特徴的な点を線で結んだ形のデータ) として扱われます。このため、拡大縮小しても滑らかさが保たれ、高解像度の美しい印刷が可能になります。PDFは、PostScriptをベースにした形式で、多くのプラットフォーム上で、ほぼデザインを変えずに文書を表示できるという利点があります。このため、特に文書のデザインを重視する場合に、pdf形式が用いられます。

一方、WYSIWYGでない文書作成ソフトの例としては、例えばTeXがあります。TeXは、フリーの文書作成ソフトで、数式に強いなどの特徴があります。このため、特に理工系の大学や研究機関で広く利用されてきました。TeXを用いた文書の作成は、コンピュータプログラムを書くのと類似しています。TeXでは、まず、文書の内容とともに、レイアウト等を記述するコマンドを追加したソースファイルを作成します。そして、ソースファイルをコンパイル (すなわち、変換処理) することで、プレビューや印刷用のファイルに変換します。

例えば、図1-3(a)は、TeX文書の例です。TeXでは、バックスラッシュで始まるコマンドを使って、文書の構造などを記述していきます。図1-3(b)は、作成された文書になります。複雑な数式なども表示できているのが分かります。このようにTeXを利用することで、高品位な文書を作成することができます。

```
\documentclass{jarticle}
\title{TeX文書の例}
\author{Yoshitomo Yaginuma}
\date{\today}
\begin{document}
\maketitle
\section{はじめに}
TeXは多くのプラットフォームで動作するフリーソフトです。
\section{数式の例}
\begin{equation}
\int_0^1 \sqrt{1-x^2} \, dx = \frac{\pi}{4}
\end{equation}
のように、数式の作成に強いといった特徴があります。
\end{document}
```

(a)TeXのソース



(b)作成される文書

図1-3 TeX文書の例

次に、インターネット上の文書について見てみましょう。World Wide Webは、インターネット上で文書等の情報を共有するための仕組みです。Web上の情報は、Webブラウザと呼ばれるWebページを見るためのソフトウェアを使って閲覧することができます。Webブラウザとしては、Microsoft Edgeや、Safari、Firefoxなどがあります。Webブラウザ上で、知りたい情報をクリックするだけで、関連する情報を参照することができます。

Webページの記述には、HTML（HyperText Markup Language）と呼ばれる言語が利用されます。HTMLでは、タグによって、構造やレイアウトの指定が行われます。HTML文書の作成には、通常のテキストエディタやワープロソフトを利用することができます。保存する際は、テキスト形式で、ファイルの拡張子を.htmlとして保存します。

Webページを外部に公開するには、Webサーバと呼ばれるサーバが必要になります。サーバ上のWebページには、URLと呼ばれるWebページの所在を表す文字列が割り当てられます。例えば、放送大学のホームページは、<https://www.ouj.ac.jp>になります。このURLを、Webブラウザで指定することで、該当するWebページを閲覧することができます。

HTMLのタグについて、もう少し詳しく見てみましょう。図1-4(a)は、HTML文書の例です。このHTMLファイルをWebブラウザで読み込んだ場合、図1-4(b)のように表示されます。

```
<html>
<body>
<h1>HTMLのサンプル</h1>
<p><a href="http://www.ouj.ac.jp/">OUJホームページへのリンク</a></p>
<p>画像の表示例 (↓) <br>
</p>
</body>
</html>
```

(a) HTML文書



(b) Webブラウザでの表示

図1-4 HTML文書の例

HTMLでは、タグによってその属性を表現することができます。また、記述は、タグも含めて文字だけで行うことができます。これは、一般的な文書を管理するためのデータ構造として利用するのに都合が良い特徴です。HTMLのように属性データを文書中に記述する言語は、マークアップ言語と呼ばれます。そして、このようなマークアップ言語を作成するための言語（メタ言語）として策定されたのがXMLになります。XMLから作成された言語としては、様々なものがあります。BMLは、デジタル放送の番組に関連する情報などを配信するために利用されます。SVGは、2次元ベクトル画像（ベクタ画像）を記述することができます。SMILは、映像や音声の、レイアウトや再生時間などを記述することができます。X3Dは、3次元データを記述できます。このようにXMLを利用することで、さまざまなデータを記述することができます。