[OpenGL 2D 2018 第01回]

コンピューターの仕組み

# コンピューターの機能

## はじめに

　みなさんはコンピューターについてどんなイメージを持っているでしょうか。コンピューターは日本語では電子計算機と呼びますが、基本的にはその名前のとおり、なんらかの計算を行うための機械です。現在のコンピューターは単なる計算だけでなく、ゲームをしたり、動画を見たり、音楽を聞いたりすることができるようになっていますが、コンピューターにとってはそれらも計算対象なのです。計算結果をディスプレイに出力すれば画像が表示されますし、スピーカーに出力すれば音声が流れます。プリンターに出力すれば文書が印刷されます。つまり、計算結果を人間が求める形で表現しているだけなのです。

　コンピューターは数字や文字、画像や音声などの様々な情報を処理することができますが、みなさんがそれらを操作するとき、文字や画像がコンピューター内でどのような仕組みによって表されているのかということはあまり考えていないでしょう。実際、文字や画像を自由に操作したいだけならば、キーボードの配列を覚えたり、お絵かきソフトやワープロソフトの使い方に習熟すれば十分です。

　しかし、ソフトウェア開発のように高度な作業を行おうとした場合、ソフトウェアの「使い方」を知っているというだけでは十分ではありません。ソフトウェアによってコンピューターを自由に操れるようになるためには、単にコンピューターやソフトウェアの使い方を覚えるだけではなく、その背後にある「コンピューターの仕組み」を理解することが必要となります。これは入学して間もないみなさんにとっては実感の湧かないことかもしれませんが、今後様々な授業やプロジェクトなどを通して、コンピューターの仕組みを理解することの重要性を実感していくことになるでしょう。

## 基本的な機能

コンピューターは、次に示す5つの機能を持っています。

（１）制御  
（２）入力  
（３）記憶  
（４）演算  
（５）出力

　コンピューターは命令を解釈し、それにしたがって演算を実行することで処理を行っています。コンピューターの心臓部を構成するのは、中央処理装置(Central Processing Unit(せんとらる・ぷろせっしんぐ・ゆにっと)=CPU)と主記憶(一次記憶)装置です。主記憶装置には一般的にはDRAM(でぃーらむ、Dynamic Random Access Memory(だいなみっく・らんだむ・あくせす・めもりー))が使われます。これらの装置は、半導体素子を高密度に集めた「集積回路」と呼ばれる部品を組み合わせて作られています。

　中央処理装置(CPU)は制御機能と演算機能をもっています。制御機能とは、命令の流れやコンピューターを構成する装置の動作を管理する機能のことです。演算機能は命令を解釈して演算を実行する機能のことです。　コンピューターは、実行する命令やデータを中央処理装置に近いところに蓄えておき、これらを命令に応じて読み出して処理を行います。このために、コンピューターにはそれらの情報を記憶しておく機能が必要です。主記憶装置は一時的にプログラムやデータを記憶し、中央処理装置にそれらを直接提供します。

　パーソナルコンピューター(PC)などのある程度以上の規模を持つコンピューターでは、プログラムやデータを保管するために補助記憶(二次記憶)装置と呼ばれるものを使用します。補助記憶装置は一般的に主記憶装置よりも大容量で、多くは周辺機器として接続れます。そのため、外部記憶装置とも呼ばれます。しかし、主記憶装置と比較すると読み書きの速度は遅い場合がほとんどです。外部記憶装置にはハードディスクドライブ(HDD)、ソリッドステートドライブ(SSD)、SDカード、USBメモリなどがあります。

　コンピューターはそれ単体ではなんの仕事もできません。人間、あるいは外部のなんらかの事象とコンピューターとのあいだでコミュニケーションを行うために、入力装置と出力装置を設けます。入力装置は外界の事象や人間の意図を伝えるために使われ、キーボードやマウス、マイクなどのほか、光センサーや温度センサーのような装置も含まれます。出力装置はコンピューターのデータ処理の結果を表示するもので、ディスプレイやプリンタ、スピーカーなどの装置が使われます。

　では、それぞれの装置について動作の仕組みを見てみましょう。まず、キーボードやマウスなどの入力装置を使ってソフトウェアを実行する指示を行うと、ソフトウェアがハードディスクなどの補助記憶装置から主記憶装置に読み込まれます。文書処理ソフトウェアや表計算ソフトウェアは、文書データや表計算データを作成、編集し、これらのデータも主記憶装置に置かれます。

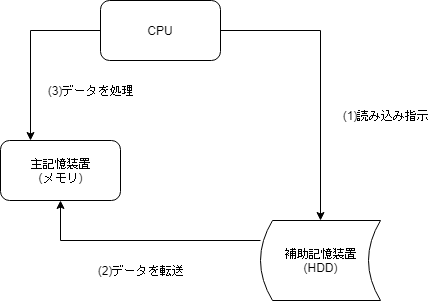


図１ CPUと記憶装置のやりとり

　キーボードからデータを作成、編集するための命令を打ち込むと、ソフトウェアがこの命令を中央処理装置に伝え、必要なデータの一部が取り出されて、中央処理装置がデータを使って演算を行います。演算の途中経過や結果をディスプレイ上に表示したり、作成した文書や表、計算結果をプリンタなどに出力することもできます。

　コンピューターの電源が切れると主記憶装置に記憶した内容は消えてしまいます。そのため、作成、編集が済んだデータを保存する場合は、主記憶装置から補助記憶装置に移し、そこに保存します。

## コンピューターの機能を実現する装置たち

　コンピューターの制御・演算・記憶・入力・出力の基本的な機能が、どのようなハードウェアで実現されているのかを見ていきましょう。

**・中央処理装置（CPU)**

　中央処理装置は、演算機能と制御機能を担い、主記憶装置上のプログラムから一つ一つ命令を読み出して実行(演算)する装置です。その性能は、一つの命令で処理できるデータのビット数や動作速度で表されます。

　一つの命令で処理できるデータのビット数に応じて３２ビットCPUや６４ビットCPUなどがあり、データ処理の速度や規模に対応しています。また、中央処理装置の各部分の基本動作のタイミングを合わせるために、一定の時間間隔でクロックと呼ばれるパルス信号を発生させており、このクロック信号の間隔が短ければ短いほど中央処理装置の各部分の動作は速くなります。クロックが１秒間に何回あるかを示す数値としてクロック周波数が使われます。一般に同じ種類のCPUであれば、その周波数が高いほど中央処理装置の動作速度も速いということができます。

**・記憶装置**

　記憶装置には、主記憶装置と補助記憶装置があります。主記憶装置はDRAMのような高密度の半導体記憶素子で構成されることが多く、その容量はコンピューターの性能を示すものの一つです。補助記憶装置には、ハードディスクのほかに、フロッピーディスクドライブやMO(光磁気ディスク）ドライブなどが使用されます。

　DVDドライブやCD-ROMドライブは、映画やゲームなどの大容量のソフトウェアの読み出しを行う補助記憶装置ですが、パーソナルコンピュータからDVDやCD-ROMを作成することができる装置でもあります。

　記憶装置の性能はその記憶容量や読み書きの速度で測られますが、容量や速度の組み合わせは様々で、用途・目的によって使い分けられています。

記憶装置は、入力装置または出力装置としても使われます。

**・入力装置**

入力装置にはキーボードやマウス、マイクのほかにも、カメラや指紋認証装置、トラックパッドなどがあります。温度センサーや明るさセンサーなども入力装置のひとつです。また、記憶装置も入力装置になり得ます。

**・出力装置**

出力装置にはディスプレイやプリンタ、スピーカーなどがあります。また、記憶装置も出力装置になり得ます。

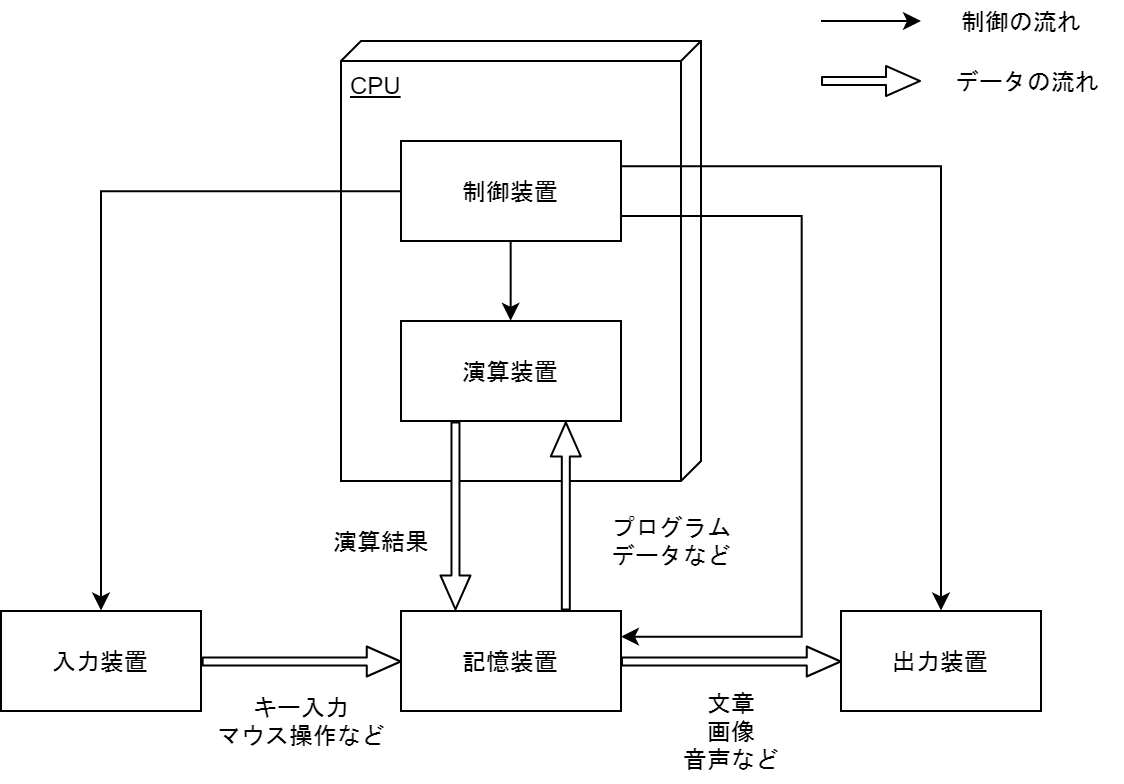


図２ 5つの機能の関係

## データとプログラム

　データやプログラムは、ハードディスク装置などの補助記憶装置にファイルとして保存されていたり、キーボードなどの入力装置から直接入力されたりします。これらのデータやプログラムは、そのまま中央処理装置が処理するのではなく、主記憶装置に送られ、そこに蓄えられます。

　中央処理装置は主記憶装置に蓄えられたプログラムやデータを読み込んで処理を行い、結果を再度主記憶装置に書き出します。そのため、データをハードディスク装置などにファイルとして保存するためには、保存操作を別に行う必要があります。主記憶装置は、一般にバイトという単位でデータを扱っています。１バイトは一般的に８ビットです。データを読み書きするために、このバイト単位のデータを収める小さな区画が論理的に設けられています。主記憶装置上のそれぞれの区画には、場所を示す数が割り当てられており、それぞれの区画の場所を示すそのような数を、アドレス（番地）と呼びます。

　中央処理装置や出力装置は、指定したアドレスから命令やデータを読み込んで処理を加えたり出力を行います。また、中央処理装置の処理結果や入力装置からの入力は指定されたアドレスに書き込まれます。

## コンピューターはプログラムをどのように実行するか

　プログラムはたくさんの命令が順番に並んだ集まりです。それを一つずつ「取り出し」、「解読し」、「実行する」という動作が繰り返し行われます。コンピューターは、基本的にこの命令を一つずつ実行していきます。そのため、命令を実行する順序は非常に重要です。

　命令は一般に処理の種類を表す「命令部」と処理するデータのある場所を示す「アドレス部」の二つの要素から構成されます。制御装置には「次に実行する命令のアドレス」を記憶している場所があって、これにしたがって命令を取り出す仕組みになっています。命令は次のような手順で実行されていきます。

1. 制御装置はプログラムによって設定された実行開始アドレスを「次に実行する命令のアドレス」として指定し、以下の２～６を繰り返す。
2. 「次に実行する命令のアドレス」にある命令を制御装置に読み込む。
3. 制御装置に設定された「次に実行する命令のアドレス」を一つ進める。
4. 制御装置は命令部を取り出して解読する。
5. 制御装置はアドレス部からデータのアドレスを決める。
6. 解読された命令とデータのアドレスにしたがって処理を行う。命令が実行順序を変更する命令のときは「次に実行する命令のアドレス」を変更する。

　このように、コンピューターは主記憶装置に蓄えられた命令を順序よく一つずつ取り出しては実行し、その流れは停止命令が実行されるまで続きます（実際には、プログラムは「それを管理している上位のプログラムに制御をゆだねる」という形で終了します）。複数の命令があっても、それぞれの命令を同時に行うのではなく、一つずつ順序立てて実行していくことに注意してください。このようなデータ処理方法を逐次処理といいます。

**TIPS:** 実のところ、現代のコンピューターは複数の命令を可能な限り同時に実行しようとします。なぜかというと、そのほうが処理がはやく進むからです。しかし、演算結果を出力する時は、あたかも命令を順番に実行したかのように出力されます。というのも、プログラムを書く人がどこが同時に実行されるかを常に注意しなければならない場合、順番に実行される場合と較べて、プログラムの作成がとても難しいものになってしまうからです。

# コンピューターはどのように情報を扱うか

## 2進法

　この章では２進法と呼ばれる数字記法について説明をします。これは、コンピューターの仕組みを考える上での最も基本的な事項ですので、少し難しいかもしれませんが、頑張って理解しましょう。普段わたしたち日常生活で使っている数の表し方は１０進位取り記法（以下１０進法）と呼ばれる体系です。１０進法を用いると、一桁で１０個の別々のものを区別することができます。例えば、０は赤、１は青、２は緑・・・といった具合です。１１個以上のものを区別したいときは、桁を増やすことによって対応します。

　さて、人間と違いコンピューターは２進法という表記体系を用いています。１０進法に慣れ親しんだわたしたちからすると非常に変わった体系に思われますが、後に述べるようにコンピューターにとっては非常に都合のよい体系なのです。１０進法においては０から９までの１０種類の記号がありましたが、２進法には０と１の二種類しか存在しません。つまり、２進法では１桁で２種類のものしか区別することができないのです。例えば、０が男、１が女、もしくは０が偽、１が真といった具合です。３つ以上のものを区別したいときは、桁を増やす必要があります（二進法は２の累乗で桁上がりを起こすということです）。２進数と１０進数の対応を表にすると次のようになります。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10進数 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| ２進数 | 0 | 1 | 10 | 11 | 100 | 101 | 110 | 111 |

図３　２進数と１０進数の対応

　この図ように、例えば８個のもの（図では０から７までの数字）を区別する際に、１０進法では一桁しか必要ありませんが、２進数では三桁も必要となります。同様に、２５６個のものを区別しようとすると、１０進法では３桁あれば十分なのに対して、２進法では８桁も必要となります。随分２進法は効率が悪いと思われるかもしれませんが、電気回路であるコンピューターにとってはこちらの方が都合が良いのです。電気信号は一般にONかOFFの二通りの状態しか持たないので、ONを１、OFFを０とすると、ONとOFFの組み合わせで様々な情報を表すことができるのです。二進法を用いてコンピューターがどのように情報を表しているかについては後ほど説明します。

## 情報量

　コンピューター内で表される数字や文字、画像などの情報は、必ず一定の量を持っています。「この画像ファイルは３メガバイトもある」とか「一秒間に５６キロバイトの情報を送れる」、もしくは「７ビットコード表」などということを聞いたことがあるかもしれません。この時、「３メガバイト」や「７ビット」というのは「ビット」や「バイト」という単位によって情報の量を表しています。一体、情報量とはどのようなことを意味しているのでしょうか。ここでは、情報量と情報量を表す単位であるビットやバイトについて学んでいきます。

## 情報量の単位としてのビット、バイト

　カードやコインには表と裏があるので、例えば表が黒、裏が白のカード一枚を使って、「晴れ」と「雨」の二つの天候を表すことができます。また、このカードが２枚あれば「晴れ」、「雨」、「曇」、「雪」といったように４つの天候を、３枚あれば８つの天候を表すことができます。

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | 晴れ |
|  |  |  | 曇り |
|  |  |  | 雨 |
|  |  |  | 嵐 |
|  |  |  | 霧 |
|  |  |  | 雪 |
|  |  |  | みぞれ |
|  |  |  | 雹 |

図４ ３枚のカードによる天候の表現

　さて、一枚のカードは二つの天候しか表すことができないのに対して、二枚のカードは四つの天候を表すことができます。これは一枚のカードよりも二枚のカードの方が表すことのできる情報の量が大きいということを意味しています。三枚のカードは８つの天候を区別し、表すことができるので、二枚のカードよりも更に情報量は大きいということができます。このように、情報量とは「何種類の情報を区別し、表すことができるか」ということを意味する概念で、当然、一枚のカードより三枚のカードの方が表すことのできる情報の数が多く、情報量は大きいわけです。そして、情報量の単位として用いられるのが「ビット」といわれるもので、一枚のカードやコインのように、二つのものを区別できるものは、「情報量１ビットをもつ」といいます。表が黒、裏が白の３枚のカードを並べると８つのものを区別し、表現することができますが（図４）、この時３枚のカードによって表現されたものは情報量としては３ビットを持つといえます。

**TIP:** ビット(Bit)の語源は２進法の一桁を意味する、Binary Digitです。

同様に、図５のように１６枚のカードを並べた画像表現は１６ビットの情報量をもつといえます（すなわち、２の１６乗通りの異なるものを表現できるということです）。

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

図５ 16枚のカードによる画像表現

　さて、コンピューターの内部では、全ての情報はコインやカードの表裏のかわりに１か０かの二進数によって表現されます。先ほどのカードの表裏を二進数の１、０に置き換え、８つの天候を０から７までの数に置き換えると、図６のように対応づけることができます。

この場合、８個の数を区別することのできる二進数表現（すなわち、三桁の二進数）は何ビットの情報量をもつといえるでしょうか。二つのものを区別できるもの（例えば、一枚のコイン、一枚のカード）は情報量として１ビットをもつのでした。二進数において「一枚のコイン」の「枚」にあたるのは、桁数であるといえます。一桁の二進数ならば、１か０という二つの状態の区別ができるので、１ビットの情報量をもちます。同様に、２桁ならば１１、１０、０１、００という４つの情報を区別できるので、コインやカードで言えば２枚、すなわち２ビットの情報量をもつといえます。そうすると、０から７までの８つの数を区別するためには二進数で３桁必要なので、３ビットの情報量をもつことになります。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10進数 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 2進数 | 0 | 1 | 10 | 11 | 100 | 101 | 110 | 111 |
| ビット | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 3 |

図６ ビットと桁数の対応表

　また、２５６個の情報を区別するためには８ビット必要ですが、一般に８ビットを１バイトと呼ぶことになっています。さらに、１０００バイトを１キロバイト、１０００キロバイトを１メガバイトと呼びます。よって、「このファイルは３メガバイトもある」という表現は、「このファイルは３メガバイト＝３０００キロバイト＝３００００００バイト＝２４００００００(2400万)ビットの情報量を持つ」ということを意味しています。カードの例で言えば、２４００万個の表が黒、裏が白のカードを並べて表現されたものであるといえます。そして、２４００万ビットの情報量を持つ、とは２の２４００万乗個のものを表現できるということを意味しています。

**TIP:** バイト(Byte)という名前は、IBMという会社のコンピューター科学者がビットのかたまりをBite(バイト=「ひとくち」という意味）と呼ぼうとしたところ、綴りがBit(ビット)と似ていて間違えやすいので、わざとiをyに置き換えて書いたのが始まりです。使いやすいのでみんなが使うようになりました。

## 16進数

　２進法の０と１を使ってコンピューターは様々な情報を表しています。このように、情報を表すために使用されている０と１の配列のことをビット列といいます。ビット列で情報を表す場合、情報量が増えると桁数もそれにつれて増えていきます。桁数が増えすぎると読みにくいため、ビット列を４桁ごとに区切って表現することがあります。このとき、４桁の２進数を１６進数で表します（図７）。

|  |  |  |
| --- | --- | --- |
| 2進数 | 10進数 | 16進数 |
| 0 | 0 | 0 |
| 1 | 1 | 1 |
| 10 | 2 | 2 |
| 11 | 3 | 3 |
| 100 | 4 | 4 |
| 101 | 5 | 5 |
| 110 | 6 | 6 |
| 111 | 7 | 7 |
| 1000 | 8 | 8 |
| 1001 | 9 | 9 |
| 1010 | 10 | A |
| 1011 | 11 | B |
| 1100 | 12 | C |
| 1101 | 13 | D |
| 1110 | 14 | E |
| 1111 | 15 | F |

図７ ２，１０，１６進数の対応表

　１６進数は１６とその累乗で桁上がりを起こしますが、０から９までの１０個の数字だけでは表現できないので、これに加えてA～Fの６個の英字を使って数を表しています。そのため、１０進法の１０は１６進法のA、１０進法の１５は１６進法のFとなり、１０進法の１６が１６進法での１０となります。１６進数の位取りの考え方は次のようになっています。

## 2進法と16進法の変換

　２進法と１６進法の相互変換は次のように簡単にできます。たとえば、２進法の１００１１１００１１０を１６進数にする場合、以下の手順で変換します。

１． 与えられた２進数を、下の位から順に４つずつのグループに分ける。

１００　１１１０　０１１０

２． 各グループを、対応表を基に１６進数で置き換える。

１００→４　１１１０→E　０１１０→6

３． 答えは4E6

　１６進数から２進数に変換するには、これと逆の操作をすればよいわけで、１６進法と２進法との変換は１０進法と２進法の変換に比べてずっと簡単です。

## 演習

次の２進数を１０進数に変換しなさい  
（１）１１０  
（２）１００１００

次の１０進数を２進数に変換しなさい  
（１）１５  
（２）２８７

次の２進数を１６進数に変換しなさい

（１）１０１０１１１１

（２）１００１００１００１

次の１６進数を２進数に変換しなさい

（１）７Ｆ

（２）３Ｅ８

# コンピューターはどのように文字を扱うか

　さて、以上で文字や数値などの情報を２進法で表現するための準備が整いました。いよいよコンピュータの中で実際に使われている表現方法を学んでいきましょう。ここでは文字情報をコンピュータ上で表現する原理について学びます。

## コード化

　コンピュータで情報を表す第一歩は、様々な記号の中からコンピュータで表すべき記号を選択することです。例えば、漢字ならば使用頻度の高いものを選択する必要があります。次に、選択した記号をどのようなビット列に対応づけてコード化（符号化：encoding）するかを決めなければなりません。例えば、００が晴れ、０１が曇り、１０が雨、１１が雪という対応づけをすることも、立派なコード化です。コード化された記号全体をコード系（character encoding system）と呼び、通常は表の形でまとめられます。この表のことを、コード表（character encoding table）といいます。

　文字のコード化は、文字記号の少ないアルファベット圏では比較的容易ですが、漢字圏のように多数の文字があるところでは様々な問題があり、困難です。どの文字を採用するかを考えることも大変ですし、採用した文字とビット列の対応づけも面倒な作業になります。加えて、後で文字を追加するときには、先に作られたコード系といかに整合性を保つかといった多くの作業をこなさなければなりません。

## コード化の規約

　さて、実際にコード化はどのように行われているのでしょうか。みんなが勝手なコード化を行ってしまうと、ソフト間や通信によるデータのやりとりに混乱が起こることは容易に想像がつきます。電話回線や通信回線を使用して文字データを送受信する場合にはお互いに「どのように文字をコード化するか」を決めておかないと、送信した文字情報を受信した文字情報とが一致せずに意志を伝えることができません。ですから、コード化の作業は、関係者や関係諸団体が集まって議論をしながら行われています。

　コンピュータは英語圏のアメリカで誕生したので、まず英語のアルファベットのコード系がつくられました。今日の社会では英語は事実上の国際語として機能しているため、このコード系はすべてのコンピュータ社会で使えるようになっています。英語圏の代表的なコード系としてはASCII(アスキー)があります。

　しかし、コンピュータが広く日常的に使われるようになると、英語圏以外の国は自国の言語に応じたコード系を作るようになりました(当時は通信手段が少なく、他国と情報をやり取りすることがあまりなかったため、それで十分だと考えられたのです)。しかし、それらのコード系はその言語圏でのみ利用することを前提として作られたので、インターネットのように国際的なコンピュータネットワークが普及しはじめると様々な問題が表面化してきました。特に、文字数の少ないアルファベット圏と漢字のような文字数の多い言語圏との間では、記号に割り当てるビット数の違いなど、大きな問題が生じました。

　現在、日本で使われている代表的なコード系としては、iso-2022-jp、シフトJIS、EUC-JP、Unicode(UTF-8, UTF-16) などがあります。

## Unicode(ユニコード)

　コンピュータが世界的に普及するに伴い、様々な言語の各種アルファベットと漢字などの表意文字を統一的に扱おうという提案がなされており、ISO(あいえすおー、国際標準化機構)という団体が、これを取りまとめて国際規格として採択しました。この規格はユニコード（Unicode）と呼ばれ、一部の例外を除き全ての文字を取り扱うことを目指しています。当初はUTF-16という1文字に16ビットを割り当てるユニコード規格が制定され、これによって、最大65536種類の文字を表現することができるようになりました。しかし、文字は各国、各民族に固有の文化が関係してくるため、16ビットでは全ての言語で使われる文字を統一的にもれなくコード化するのは難しいことでした。また、英語圏で広く普及していたASCIIコード系との互換性がなかったこともあり、なかなかUTF-16は普及しませんでした。その結果、現在ではASCIIコード系との互換性を保ちながらも、最大31ビットの情報量(文字にすると約21億種類以上)を持つ、UTF-8というコード系が普及しつつあります。

# まとめ

ここまで、コンピューターがどのような仕組みで動き、どのように情報を扱うかを見てきました。  
コンピューターは電気信号のONとOFFを２進数に見立てて、さまざまな計算を行っています。  
また、数値や天気、画像、文字などの情報も、適切にコード化することでコンピューター上で扱うことができるようになります。  
情報は補助記憶装置から主記憶装置へと読み出され、さらに中央処理装置(CPU)へと送られてさまざまな演算が行われた後、主記憶装置を経由して補助記憶装置に書き戻されたり、ディスプレイやスピーカーなどの出力装置に送られます。  
そして、プログラムはこの流れを制御する命令が順番に並んだ集まりです。プログラムを変えることによって、コンピューターにさまざまな動作をさせることが可能となります。また、プログラム自体も、コンピューターで処理される情報の一種です。