

<発展学習>

プログラム言語の基礎知識

1. コンピュータとアルゴリズム

コンピュータ科学では、日常生活の仕事、家庭でケーキを焼くための調理法や携帯電話でメールを見るための操作法を例にすると、仕事を「処理（プロセス）」、仕事を行う方法を「アルゴリズム」という。そして処理を実行する担い手のことを「プロセッサ（processor）」という。プロセッサは、人であったり、機械的な装置であったり、コンピュータであったりする。アルゴリズムはプロセッサが理解でき実行できるような形式で表現されていなくてはならない。

2. プログラムとプログラム言語

コンピュータがアルゴリズムを理解し、実行することをインタプリート（解釈する[interpret]）という。プロセッサがコンピュータのときは、アルゴリズムはプログラム（program）と呼ばれる文書で表わされる。一般に処理（プロセス）は一連の細かいステップで構成されている。プログラムは、アルゴリズムの各ステップに対応して、一連の命令文（statement、単に文、または命令ともいう）で表され、それぞれの命令文がコンピュータの行うべき演算を想定して構成される。プログラムはプログラム言語で書かれる。仕事の目的や規模に応じて多くのプログラム言語が開発されており、それぞれ特有の文法と文の性質を持っている。

コンピュータはパソコン（PC）だけではない。携帯電話、液晶テレビ、電子炊飯器、自動車など多数の電子機器にも様々なプロセッサが搭載されている。それぞれのプロセッサで、直接、解釈できるように作られたプログラム言語が必要で、これらを機械語という。

コンピュータや電子機器の利用拡大にともない、機械語よりも格段に便利な高水準言語と呼ばれるプログラム言語と、それを機械語のような低水準言語に変換するプログラム言語処理システム（これも大きなプログラム）が開発され、現在では高水準言語でプログラムを書くことが一般的である。プログラム言語はプログラミング言語といわれることがある。

高水準言語の例：Cobol、Algol、Pascal、PL/I、Smalltalk、Java、C++、Visual Basic、Ruby

中水準言語の例：Fortran、Basic、C、Lisp

3. ソフトウェアとハードウェア

プログラムは、もっと総称的な用語であるソフトウェア（software）とも呼ばれる。ソフトウェアに対して、コンピュータシステムを組み立てている物理的な装置のそれぞれ及びシステム全体をハードウェア（hardware）と呼ぶ。コンピュータシステムの利用者は仕事の目的があって、コンピュータシステムを道具として使い、普通はアルゴリズムが分かっている、プログラムが既に用意されているような仕事を行う。

文書作成、販売や経理の計算、顧客名簿の管理、データの統計分析、建造物の強度計算や Web ページの閲覧・検索など、特定の分野で使う 1 つに纏ったプログラムは、応用パッケージプログラム（略して、アプリケーション[application]ということが多い）といわれる。アプリケーションは通常、高水準言語で書かれるので、これをコンピュータで利用するには、適当なプログラム言語処理システムが必要なのは当然であるが、特定のプロセッサやハードウェアを制御するための多数のプログラムが必要である。こうしたプログラムは、システムソフトウェアと総称される。システムソフトウェアの代表的な

ものが、Linux、Windows、iOS や Android などのオペレーティングシステム（operating system, 略して OS という）である。

ここで、ソフトウェアとハードウェアの役割には明確な区別がないことを指摘しておこう。例えば、乗算の処理は OS の一部として書かれたプログラムの実行という形で行うこともできるし、乗算のアルゴリズムを CPU というハードウェアの中に埋め込み、CPU 内で単一の演算として実行することもできる。実用上は、ソフトウェアとハードウェアの選択は速度と経費の兼ね合いによるので、技術の進歩と共に変わりうる。

4. プログラム言語の分類

プログラム言語は、プログラムを機械語へ変換する方式で区別すると、アセンブラ、コンパイラ、インタプリタ、などに分類される。

- アセンブラは、機械語と 1 対 1 に対応して書くことができるアセンブリ言語のプログラムを機械語に変換するプログラムであり、低水準言語に属する。
- コンパイラは、特定のプログラム言語で書かれたプログラムを一括して特定の機械語に変換してから実行するプログラムである。コンパイラでは、プログラムの文法チェック、文や式の解析、コード生成と最適化を行い、実行可能なプログラムを生成する。コンパイラ方式の言語には、Fortran、COBOL、C、Pascal、C++、Java などがある。
- インタプリタは、特定のプログラム言語で書かれたプログラムを 1 行ずつ読み込み、記述された命令の解釈・実行を行うプログラムである。インタプリタでは、処理速度は遅くなるが、文法チェック時の手軽さが利点である。インタプリタ方式の言語には、Lisp、Basic、Perl、Ruby などがある。

これまで例として挙げたプログラム言語は、いずれもどんな目的の仕事でも記述できる汎用言語である。しかしそれ以外にも、データベース管理システムで管理されたデータを利用したアプリケーションを記述するための SQL 言語、統計処理や数式処理のパッケージを利用するための言語などの特定用途のいわゆる簡易言語があり、これらもインタプリタ方式の言語である。また、HTML と CSS は WWW のページを記述するためのマークアップ言語であり、XML はプログラム同士で厳密かつ的確なデータ交換を目的とした拡張可能なマークアップ言語である。WWW（または Web とも略す）のブラウザはこれらの言語で書かれたテキストファイルをインタプリタ方式で表示・実行するアプリケーションである。

簡易言語やマークアップ言語などは、使用されるソフトウェア環境によって、スクリプト、マクロ、などに分類される。

- スクリプトは、簡単な記述でプログラムを作成することが可能なインタプリタ方式の言語である。特に、OS のシェルや Web 関連のプログラムで利用される場合はスクリプトと呼ばれることが多い。例えば、JavaScript や PHP で記述されたプログラムは、Web のページを記述する HTML 文書に組み込まれたり、文書内から呼び出されて利用される。他にも、Tcl/Tk、Perl、Ruby、Flash、VBScript、Python、Script-fu などのスクリプト言語がある。
- マクロは、文書作成や表計算などの Office アプリケーション（例：VBA、OOo Basic）や Photoshop や Gimp などの画像処理アプリケーション（例：PS アクション、Script-fu）の利用において、一連の操作をまとめて 1 つの処理として保存し、頻繁に行う仕事や面倒な仕事を便利にするプログラムである。マウスやアプリケーションの操作・機能に応じたイベントが起こったときの処理だけを記述し、それ以外の処理はアプリケーションに任せるというように、簡単にプログラムを作成できるため、プログラミングの初心者でも利用可能である。

なお、本書ではアプリケーションに備わっているマクロ機能またはマクロ言語を利用して作るマクロ

プログラム言語(2013)

プログラムを区別せずに、マクロと呼んでいる。

このマクロ機能を悪用して作成されたコンピュータウイルスにマクロウイルスがあり、Word や Excel に感染して大きな被害が出ている。データベースアプリケーションの Access では、Excel と同様にマクロを作成することができるが、それに加えて直接、SQL でスクリプトを作成することが主な活用法となる。

参考文献

1. L.Goldschlager, A.Lister (武市正人他訳)「計算機科学入門」近代科学社(1987)、参照は 1.序論
2. 大内東・岡部成玄・栗原正仁編「情報学入門」コロナ社(2006)、参照は第 8 章