

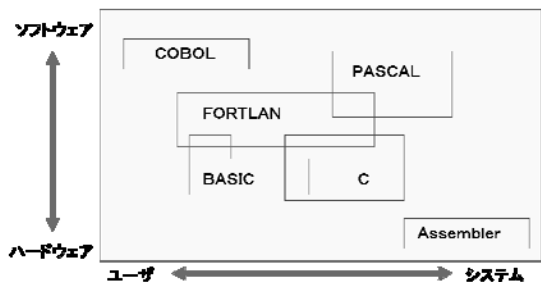
2. C 言語の仲間たち

システム記述用言語

C 言語は、UNIX を記述するために誕生した言語ですから、一般には「システム記述用言語」と呼ばれています。

OS 等を記述するには、ある程度ハードウェアが分かっていないと都合が悪いことがあります。しかし、C 言語は、ハードウェアの制御、タスク制御等も書け、より高速なプログラムが書けるというだけで、記述方法は一般のプログラミング言語と同じです。ですから、言語を理解する上では、ハードウェアまで理解する必要はありません。

「実習用」とか「よく分かる」と銘打った本に、こんな絵が出てくることが多いようです。



ずいぶん、アセンブラに近くて難しそうなイメージですが、FORTRAN や BASIC より分かりやすい制御構造になっています。

データの集まりを配列で表現するしかない旧来の言語より、データの集まりを構造体で表現できる C の方が分かりやすいと思います。Visual Basic には、ユーザ定義型がありますが、これは、オブジェクト指向が普及した後のことです。

この種の図は、言語レベルではなく「ふつう、こんなところに使われている」という意味にとりましょう。もちろんエンドユーザが使うようなソフトを作ることでもあります。私は、経理部門用のソフトを C で作成したこともあります。

C, C++, C#, Java

オブジェクト指向の考え方が普及するにつれて、C にオブジェクト志向の考え方を取り入れた処理系 C++ が開発されました。C に数多くの機能を追加したという意味で「++」と名付けたそうです。

一方、C や C++ のポインタの概念が分かりにくいということで、ポインタの概念をとっぴらった Java も提供されるようになりました。

Windows の GUI を前提とした Visual C++ が Microsoft 社から提供されるようになりました。C++ は、細かいところを除いて従来の C との互換性が保たれていますが、Pascal 等のデータ抽象化の概念を導入し、Microsoft Visual Studio .Net の考え方を基本にして、新たに設計された処理系 C# が提供されます。.Net では、C# の方を推奨しており、記法もかなり改善されています。

なぜ、シャープ(#) かって？ プラスプラス(++) を縦に並べれば、# に見えるかと思えます。



C++ の機能を「大幅アップしたよ」というメッセージなのです。