

大学におけるソフトウェア開発管理技法

— フログラムシンセサイザと数学的ソフトウェア —

九大大型計算機センター 松尾 文碩, 高木 利久

フログラムシンセサイザは、フログラムの仕様をあたえる述語(仕様述語) $R(x_1, \dots, x_m, y_1, \dots, y_n)$ を満たすフログラムを自動的に生成するフログラムである。ここで、

$$R(x_1, \dots, x_m, y_1, \dots, y_n) \equiv P(x_1, \dots, x_m) \supset Q(y_1, \dots, y_n);$$

$x_1, \dots, x_m$ は入力変数; $y_1, \dots, y_n$ は出力変数;

$P(x_1, \dots, x_m)$ は入力述語; $Q(y_1, \dots, y_n)$ は出力述語である。

シンセサイザは、つぎのような利点がある; (i) シンセサイザの正当性が証明されたとすると、シンセサイザが生成するフログラムは正当性が証明されたものとみなすことができる; (ii) 現在の言語プロセッサの最適化よりもさらに高度な最適化が可能である。 (iii) フログラム仕様が厳密に論理的に書かれることになるので、フログラムの管理がしやすく、変更、拡張が容易である。

反面、仕様述語の記述には、フログラム言語によるコーディングに比べ、より高い知的能力が要求される。しかし、大学のように組織力が弱いところで、職業的フログラマでない

研究者によってソフトウェアの開発と管理を行う場合の開発用の道具としては、シンセサイザは最適である。

シンセサイザを実現するためには、つぎのことが研究されなければならない。

1) シンセサイザは、言語プロセッサと異り、仕様述語が文法的に誤りがない場合でも、それに対応したオブジェクトコードを生成しないことがある。たとえば、仕様述語が充足不可能ならば、それに対応するプログラムは存在しない。このため、仕様述語記述の立場から、プログラムが存在するための条件を解明しなければならない。

2) 記述言語の関数記号と述語記号、すなわち primitives の選択の問題を、1) とプログラム技法との関連を追及する必要がある。シンセサイザは、原理的には、論理記号に対し流々の制御構造を対応させ、primitives に対してはサブルーチンプログラムを対応させることになる。シンセサイザの構成要素としてのサブルーチンプログラムは、正当性が証明されていることはもちろん必須であるが、効率、精度の面で十分に洗練されていなければならない。このため、数学的ソフトウェアを含め、各種ソフトウェアの評価が重要である。