

ソフトウェア保護の基礎的研究

工藤市兵衛・鈴木 達夫・近藤 高司

A Study on the Basic Factors for Legal Protection of Computer Software

Ichibei KUDO, Tatsuo SUZUKI and Takashi KONDO

In recent years, a demand of computer softwares has being grown as the computer systems were widely used in advanced countries. The more advanced computer systems require the development of a large scale software. The development of this kind of softwares requires a large amount of capital investment. Nowadays, the software is recognized as valuable property. However, it seems that the current laws do not provide softwares with enough protection against legally dishonest act. This is due in large part to the characteristics of the software such as intangible property and its new existance which is not directry covered by the current laws. The present study examines the right protected and the feature of software, and discusses the basis for the legal protection of softwares.

1. 緒 言

今日、コンピュータは社会のありとあらゆる分野で使用され、コンピュータ技術の進歩は社会に大きな影響を与えている。その反面新たな問題が発生しつつあり、その一つとしてソフトウェアの法的保護のあり方が議論されている。我が国の汎用コンピュータ設置数は昭和57年6月には11万台を超え¹⁾、産業界においてOA、FAと叫ばれる合理化のため汎用大型コンピュータからマイクロコンピュータまで広範囲に利用され、将来とも増加する傾向にある。コンピュータのハードウェアは半導体技術の飛躍的進歩により、その経済性は年々改善されて、より高性能になって来ている。その、コンピュータの持つ能力を有効に引き出すソフトウェアは、そのほとんどを人間の知的労働により生み出されている。そしてオンラインシステム等の高度なシステムになればなるほど複雑化、高度化したソフトウェアを必要としている。高度な機能は大規模なソフトウェア、例えば、IBMの汎用コンピュータに使用される基本ソフトウェアであるOS/MVSは約500万ステップの大きさ²⁾であり、多数の人がその作成に従事し高額な投資を要している。しかし今日の法律制度においてはコンピュータ・ソフトウェアの法的保護の明確なる条文はない。急速に増加するソフトウェアは新しい時代の新しい社会秩序を維持するため、よ

り合理性のある保護のあり方について、新たな理論を如何に構成するかという問題に直面している。知的所有権であろうソフトウェアは今、著作権法や工業所有権法でその思想を拡大解釈し適用せんと努力がなされているが、ソフトウェア保護という課題の基本的問題について展望することにした。

2. 法工学的アプローチ

コンピュータに係わるソフトウェアの問題が社会科学、自然科学的な研究対象となることから法律学の各分野における共同研究は勿論、経済学、社会学、会計学など社会科学分野ばかりでなく、自然科学的解明をも不可欠とするものであり、自然科学の各専門領域との研究、所謂学際的研究が強調される³⁾。ソフトウェアの保護という問題はその対象が自然科学理論により成立しているコンピュータの構成要素の一部分であり、その実能に合致する法律的な社会秩序が形成されなければならない。つまり、法律学自体の学問的、内在的な枠組を脱却し、対象を自然科学的アプローチに広げ、本来的な法律学の守備範囲の追求を見直す必要のある点を強調したい。法工学とは社会秩序の維持を目的とした「法学」と物的生活上を目的とする「工学」を融合せしめた概念であり、将来において発生するであろう紛争問題を解決する補助として工学の力を貸し、更に法工学により規制された法

律、その他の諸規定と共に企業経営のための意思決定を援助する学であるとしたい。

3. 保護される権利

科学技術が急速に進む中でソフトウェアの重要性は高くなり様々な分野で利用される今日、企業経営の観点からいかなる法秩序が必要であるか、その基礎をなす権利について考える。営利を目的とする企業においてその存続・発展するため経済的価値を有する営業財産を断続的・集团的に活用している。我国において、その社会体制は資本主義の体制であり、日本国憲法において財産権の不可侵について「財産権は、これを侵してはならない」(29条1項)と規定している。この規定は私有財産制を保障したものであり、財産権には物権、債権、無体物財産権その他いっさいの財産的権利をさしている。そこで計画的・継続的に経済行為を営む企業にその経済的欲望の目的を満足させるため有形・無形の諸手段である権利のうち物権は目的物を直接支配でき、その所有する権利者は誰に向かっても権利を主張できる物権法定主義(民法175条)を有する。これは社会秩序の維持のため望ましいと認め、人(法人)がある行為をし、その利益を受けること、そして支配しうる利益を適当に配分することができるとして法律で保護される法秩序である。有史以前から、牛馬など動産、土地などの不動産に対し、その経済的価値は社会から認められ財産権が自然に発生していた。しかし近代になってから、有形ではない直接的に支配不可能な無体物あるいは知的な創作物が出現して来た。物的生産を高める発明や芸術的表現による創作物が社会的に経済価値を持ち重要となって来ている。現在のように科学技術の発達する産業界においては、人の知的創作による無体財産としての工業所有権と著作権が認められ社会秩序として確立されている。しかし、1950年代から実用化して来たコンピュータという電子装置は有体物であり直接に支配できたが、コンピュータ技術の発達は新規の概念を持つソフトウェアという無体物を産んだ。コンピュータが社会で使用されはじめた頃は、装置の付随物であり、その社会的認識は低かったけれど、高度なコンピュータシステムの利用を望む社会はその貴重な経済価値を持つ対象として独立して取引される様になって来ている。なぜなら、ソフトウェアの開発作成に大きな資本の投入が必要で、そこで支払われた代価の回収が必要である。ソフトウェアのほとんどの部分は人の知的労働の成果であり、これを売買取引することにより利益を受けるであろう商品となりうる経済的価値を持つ財産であることは社会に認められている。無体物としてのソフトウェアはその特殊で取り扱い困難な点から現在法

的にはほとんど認められておらず、いかなる権利により保護するのかが明確ではないが、先行開発者の投下資本に対するソフトウェアの所有権を有するのは明白ではなからうか。

4. コンピュータ・ソフトウェア

公正なコンピュータ社会の発展と秩序の維持のための保護の対象となるソフトウェアという言葉の定義を考えてみる。ランダムハウス英和辞典によれば「①コンピュータに関係するプログラム、手順、規則およびそれに関連する文書類の総称、②機械、商品などの付加価値を高めるための手段、方法」とあり、日本工業規格情報処理用語に「データ処理システムの運用に係る計算機プログラム、手順、規則及びそれらに関連する文書」と定義されている。この用語はハードウェア(データ処理システムを構成する装置の総称)という用語に対応して作られたもので、昭和30年代になってからソフトウェアという用語が確立した⁴⁾。ソフトウェアを広義の意味とする利用技術と訳される場合があるが以下については狭義の意味と解する。米国のコンピュータ産業では、この言葉をプログラムとデータベース(両方とも情報)を意味するものとし他方では磁気テープ、磁気ディスクそして印刷されたドキュメンテーションという情報を含む物理的媒体を意味するものとし無差別に使用されている⁵⁾。ソフトウェアという用語は狭義に解しても大変抽象的な概念であり、知的な無体物財産と考えると法的保護の権利がいかなる対象範囲に及び、社会的に認識するか考えることは重要であると思う。そこで、コンピュータシステムの具体例を用いて考えてみる。コンピュータの構成をハードウェアとソフトウェアに分けると表1の様になる。ここでは、ソフトウェアを狭義に解し、プログラム類とドキュメント類と大別することとした。図1は、銀行で運

表1 コンピュータ・システムの構成要素

ハードウェア	
中央処理装置	
入力装置	カードリーダー ターミナル
出力装置	プリンター ターミナル
補助記憶装置	磁気ディスク装置、磁気テープ装置
通信装置	
ソフトウェア	
プログラム	基本プログラム 適用業務プログラム
ドキュメント	マニュアル類 システム仕様書 フローチャート図

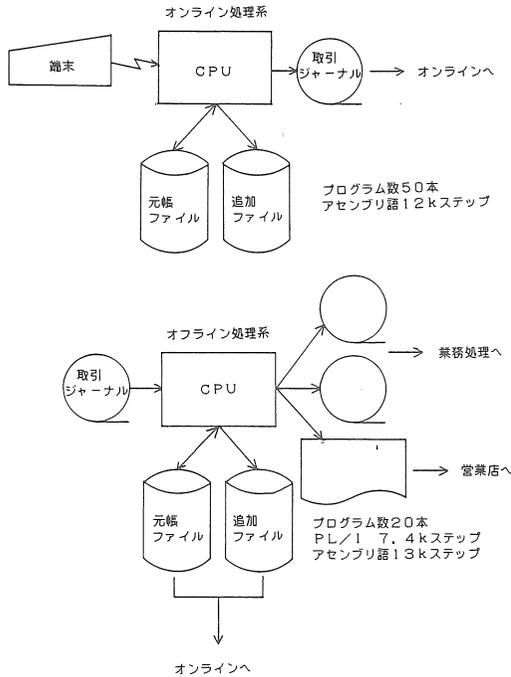


図1 適用業務処理システムの概要

用される適用業務処理システムの一処理（取引）の情報の流れを簡略に示した図である。処理はオンライン処理系とオフライン処理系から構成されて、銀行の営業店に設置してある端末から取引の情報を入力し元帳ファイルの内容と照合しつつ追加ファイルの中と磁気テープへ情報を書き、そのテープの情報を読み込み元帳ファイルの内容を変更したり各業務に必要な情報と磁気テープへ書き、営業店に対して報告書の印刷を行う。このシステムは現在、コンピュータで処理されている業務へ新しく追加する機能であり、その開発に15か月を要し、投入されたコンピュータ要員は200人月であった。新しい業務を追加するにあたり、新業務を分析するシステム設計が行なわれた。表2は当システム開発に必要なとした開発日数と開発工数割合を示す。設計は銀行の新業務の方法や企業経営のノウハウが導入され、業務知識やコンピュータの知識に基づくシステム設計書（基本と詳細設計）が作成さ

表2 システム開発の工程別日数と工数

		開発日数	開発工数割合
システム設計	基本設計	180日	8%
	詳細設計	120日	20%
プログラム開発	フローチャート コーディング 単体テスト	90日	30%
	テスト	結合テスト	90日
総合テスト		45日	16%

れる。この書類には業務の内容が十分に盛り込まれており貴重な情報が含まれるドキュメントとなる。次の工程はプログラム開発又はプログラミングと呼ばれ、前工程で作成された設計書に基づき、コンピュータのプログラム言語の知識を用いてフローチャート図を作成する。そのフローチャート図に忠実に従いプログラム言語に書きなおし、コーディング用紙に書く、用紙に書かれたプログラムをコンピュータに入力し誤りがないかテストを行い、操作説明書、利用者マニュアル（書類）を作成し完成する。この様に適応業務処理システムを開発するためには、その各工程から知的労働により、多種類のドキュメントが作られ、磁気テープ等の磁気記憶媒体の中には完成したプログラム（情報）が納まっている。表3はソフトウェア開発の標準工程と作成されるプログラム、ドキュメント類を示している。当開発で発生した費用は約1億円で、大部分は開発に従事した要員の人件費であった。現実的に発生した費用は当企業の無形財産であろう。今日のコンピュータ技術は迅速な処理能力を有しており磁氣的記録（磁気テープ媒体等）である無形財産の複製は非常に容易であり、他の企業が不法手段で入手し使用すれば上述の開発コストの大部分が不要であることは明白である。しかしコンピュータのプログラムは複製が容易であり先行開発者のプログラムがなくなる、つまりその経済的価値が無になる損害が起きることではない特徴がある。そして、ドキュメント類に関しては著作権法2条1項「著作物、思想又は感情を創作的に表現したものであって文芸、学術、美術又は音楽の範囲に属するものをいう。」とあり、言語の著作物、図表の著作物に該当することは明白であると考えられる。しかし、その内容（情報）までを保護する権利を受けるのではなく、高度な技術的情報までは保護されない。この様にしてコンピュータを利用する企業は各業種業務の目的に合致する適用業務処理システムを作成、開発している。しかしコンピュータには基本ソフトウェアと言うコンピュータのハードウェアと共にコンピュータメーカーから供給されるオペレーティングシステムが存在している。オペレーティングシステムとは「計算機プログラムの実行を制御するソフトウェアであって、スケジューリング、デバッキング、入出力制御、課金処理、コンパイル、記憶割振り、データ管理、及びこれらに関連した諸サービスを行うもの。」とJISでは定義されている。基本的な役割りは、コンピュータの資源を利用者が共有して仕事をする様に資源を適正に配分しコンピュータの処理能力を最高に維持することである。それは制御プログラムと処理プログラムからなり図2、図3で示す機能を持つプログラムの集合である。オペレーティングシステムの重要な機能の一

表3 ソフトウェア開発における標準工程とドキュメント、プログラム類

標準工程	作業範囲	アウトプット
調査分析	現状分析 予備設計 開発計画 技術計算の場合の理論解析	調査報告書 理論解析説明書
システム設計 基本設計	機能設計 システム構成とサブシステム分析 ファイル設計 入出力帳票と基本ファイルの設計 システム評価方法の設定 テスト仕様の計画・内容	システム基本設計書 詳細設計書
詳細設計	プログラム構成の設定 プログラム基本設計 プログラム機能仕様の作成共通エリア、 メッセージ、詳細ファイル、内部コード等の定義 技術計算の場合の数値解析	数値解析説明書
プログラミング プログラム設計	プログラム仕様の作成 メッセージ覧、エリアの定義含む 単体テスト仕様の作成	プログラム設計書 プログラムテスト仕様書
プログラム作成	プログラム・フローチャートの作成 コーディング 机上デバッグ、アセンブル、コンパイル 単体テストデータの作成 単体テスト (= デバッグ)	プログラム・フローチャート プログラムリスト (媒体含む) テスト結果
総合テスト	サブシステム総合テスト システムテスト テストラン 総合テストデータ作成	総合テスト結果報告書 プログラム (完成品)
マニュアルの作成	操作説明書、利用者マニュアルの作成 納品物件整理を含む	操作説明書 利用者マニュアル

(出典 ソフトウェア産業振興協会)

制御プログラム

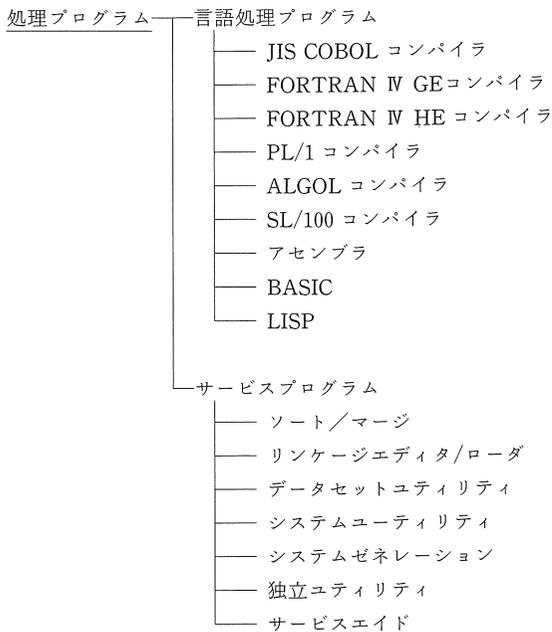
- スーパーバイザ
- ジョブ管理
- データ管理
- VSAM
- 通信管理
- 回復管理
- TSS
- AIM

(出典 FACOM OSIV/F4解説書)

図2 制御プログラムの構成

つとしてジョブ管理がある。図4にその機能の流れを示す。システムの総合的処理能力の向上はこの機能に依存している。ジョブは、カードリーダー、磁気テープ、大記

憶装置やリモート端末機から入力される。システム入力制御部は、入力ジョブストリームをシステムデータセットに書き込み、利用者の指示したジョブ制御の情報はジョブキューデータセットに記憶される。イニシエータの解釈部はスプールデータセットより入力を読み込みながら、ジョブ制御情報を必要に応じ取り出しコンピュータ資源の割当てなど行う。利用者のジョブはイニシエータが持つ仮想空間で処理され、処理された結果情報はスプールデータセット (SYSOUT) へ一時的に記憶され、プリンター、磁気テープ装置へ情報が出力される。このような複雑な操作機能の全体を制御プログラムが持ち、ハードウェアと一体となって機能している。初期のコンピュータはオペレータの手操作時間をできるだけ減らし、コンピュータの異常により停止させない目的で開発され



(出典 FACOM OSV/F4解説書)

図3 処理プログラムの構成

ていた。しかし、より高い処理能力が要求されマルチプログラミング、仮想記憶、仮想計算機等の新しい概念に基づく技術の開発により今日は大規模なプログラムとなってしまうている。IBMが1964年発表したIBM360のオペレーティングシステムOS/360は150万ステップの規模で、システム設計、プログラミング、ドキュメント作成等の作業を含むものであったとしている⁹⁾。莫大な費用の発生はオペレーティングシステムを初めとするコンピュータのソフトウェアの有償化であり、1969年IBMは、ハードウェアとソフトウェアの価格分離の営業方針すなわちアンバンドリングを発表し社会に独立した経済的価値を有する商品として認識されるようになり今日に至っている。ところが、この莫大な基本ソフトウェアの開発投資を充分に行わずしてIBMのソフトが利用できるIBMコンパチブルコンピュータが1970年代米国で出現し始めた。わが国でも1971年3月特定電子工業及び特定機械工業振興臨時措置法を制定し、国産コンピュータ6社を3グループ化し、そのうちの富士通と日立製作所は、IBMのコンパチブル路線を進みIBMに急迫した。それに対抗する技術的手段としてIBMはオペレーティングシステムのファームウェア化である。従来オペレーティングシステムのソースコード等を公開してきた。従って、分析すれば同じコンピュータを製造することが可能であった。しかし、オペレーティングシステムの重要な機能をもつプログラム(基本ソフトウェア)をマイクロコー

ドと言われる電子回路の一部品とすることができた。この機能を分析することは容易ではなく相当な時間を必要とする。IBMの大型コンピュータ3081Kで約1000語程度から3081Kで8000語程度がマイクロコード化されているようである。但し、これは、一見ハードウェアとソフトウェアの中間領域の概念に考えられるが、内容(情報)はソフトウェアであり、電子回路の部品(ハードウェア)という媒体の中に記憶されていると考えれば中間領域の概念はなくなるものと考えられよう。次のソフトウェアの保護という観点からどの範囲までをその権利を認めるかという問題点として、ソースプログラムとオブジェクトプログラムという分類がある。通常、一般のコンピュータ利用者は、基本ソフトウェアに含まれる言語処理プログラムを使いソースプログラムをオブジェクトプログラムに変換(コンパイル、アセンブルする作業を行っている。開発の工程からシステム設計書がより詳細に、そのシステムの仕様を決定し、変換で使用されるプログラミング言語の諸機能を充分に考慮してフローチャート図が作られる。場合によって、より詳細な仕様を表現したディテールフローチャート図が作られる場合もある。これらのフローチャート図からプログラミング言語に合わせて文字、記号の組合せであるソースプログラムがコーディング用紙の上に書かれる。図5はマイクロコンピュータで使用されているオペレーティングシステムCP/M-80上で作動するプログラミング言語であるPL/Iのソースプログラムのコンピュータで、印刷したりストである。ソースプログラムはコンピュータの利用者が、コンピュータの機能を使う場合に、その機能を人間の言語にできるだけ近づくように表現している。例えば、データを入力する場合、GETと書き、ファイルの中へデータを書く場合、WRITE FILEと表現する。しかし、コンピュータはそのハードウェア自身、その内部は電子回路により構成されてデジタル回路による、0か1という2進数表現を根本的に取り扱うようになっている。2進数表現はコンピュータの最も得意とする情報表現の最小単位であり、デジタル論理演算回路が経済的に製造できる。それ由に、コンピュータの最も基本となるプログラム又はコードは2進数で表現し扱われている、当然、オブジェクトプログラム——コンピュータが直接理解できる——は、機械語(マシン・コード)であり2進数表現である。今日のコンピュータはバイトを1単位とする表現で、16進数の2桁により表現される形式で人間に表示することができる。図6には、前述のオペレーティングシステムの下で動作する、アセンブラの形式表現による記号語のソースプログラム(右側)とアセンブル(変換)により作成された機械語が16進数表現で示されている。


```

GRADING:
      PROC OPTIONS(MAIN);
DCL A CHAR(10) VARYING;
DCL B CHAR(20) VARYING;
DCL C CHAR(50) VARYING;
DCL I RECORD STATIC
      2 S CHARACTER(10) VARYING INITIAL('0000'),
      2 NUMBER CHARACTER(20) VARYING INITIAL(' (NO NUMBER)'),
      2 GRADE CHARACTER(50) VARYING INITIAL(' (NONE)');
DCL N FIXED BIN;
%REPLACE
      TRUE BY '1'B,
      FALSE BY '0'B;
DCL MASTER FILE;
DCL SYSIN FILE;
DCL K FIXED BIN;
DCL EOFILE BIT(1) STATIC INITIAL(FALSE);
OPEN FILE (MASTER) RECORD UPDATE DIRECT KEYED ENV(F(80))
      TITLE('B:MASTER.DAT');
DO K=0 TO 200;
S=CHARACTER(K,10);
WRITE FILE (MASTER) FROM(RECORD) KEYFROM(K);
END;
J : PUT SKIP(3) LIST(' *** INPUT REC-NO ***** ');
GET LIST (N);
IF N>=999.0 THEN GOTO F;
READ FILE (MASTER) INTO(RECORD) KEY(N);
PUT LIST(S,NUMBER,GRADE);
PUT SKIP;
GET LIST(A,B,C);
S=A;
NUMBER=B;
GRADE=C;
WRITE FILE (MASTER) FROM(RECORD) KEYFROM(N);
GOTO J;
F: CLOSE FILE (MASTER);
STOP;
END GRADING;

```

図5 PL/1によるソースプログラムの例

		ORG	0A000H
A000?	F5	PUSH	AF
A001?	3A EA55	LD	A, (0EA55H)
A004?	F5	PUSH	AF
A005?	3E 74	LD	A, 074H
A007?	D3 87	OUT	(87H), A
A009?	3E 60	LD	A, 060H
A00B?	D3 85	OUT	(85H), A
A00D?	3E EA	LD	A, 0EAH
A00F?	D3 85	OUT	(85H), A
A011?	3A 9FFF	LD	A, (09FFFFH)
A014?	3C	INC	A
A015?	32 9FFF	LD	(09FFFFH), A
A018?	3E 01	LD	A, 1
A01A?	32 9FFC	LD	(9FFCH), A
A01D?	D3 E5	OUT	(0E5H), A
A01F?	32 EA55	LD	(0EA55H), A
A022?	32 EA56	LD	(0EA56H), A
A025?	FB	EI	
A026?	F1	POP	AF
A027?	32 EA55	LD	(0EA55H), A
A02A?	D3 E5	OUT	(0E5H), A
A02C?	F1	POP	AF
A02D?	C9	RET	
		END	

図6 記号語と機械語

```

0100 01 BC 0F C3 3D 01 43 4F 50 59 52 49 47 48 54 20 .....=.COPYRIGHT
0110 28 43 29 20 31 39 3B 30 2C 20 44 49 47 49 54 41 (C) 1980, DIGITA
0120 4C 20 52 45 53 45 41 52 43 4B 20 20 20 20 20 L RESEARCH
0130 44 44 54 20 56 45 52 53 20 32 2E 32 24 31 00 02 DDT VERS 2.2#1..
0140 C5 C5 11 30 01 0E 09 CD 05 00 C1 21 07 00 7E 3D ...O.....!..~=
0150 90 57 1E 00 D5 21 00 02 78 B1 CA 65 01 0B 7E 12 ..W...!..x...e...~.
0160 13 23 C3 58 01 D1 C1 E5 62 78 B1 CA 87 01 0B 7B ..#.X...~bx.....(
0170 E6 07 C2 7A 01 E3 7E 23 E3 6F 7D 17 6F D2 83 01 ...z...~#.o}.o...
0180 1A 84 12 13 C3 69 01 D1 2E 00 E9 0E 10 CD 05 00 .....i.....
0190 32 5F 1E C9 21 66 1E 70 2B 71 2A 65 1E EB 0E 11 2...!f.p+q*e....
01A0 CD 05 00 32 5F 1E C9 11 00 00 0E 12 CD 05 00 32 ...2_.....2
01B0 5F 1E C9 21 6B 1E 70 2B 71 2A 67 1E EB 0E 13 CD ...!h.p+q*g.....
01C0 05 00 C9 21 6A 1E 70 2B 71 2A 69 1E EB 0E 14 CD ...!j.p+q*i.....
01D0 05 00 C9 21 6C 1E 70 2B 71 2A 6B 1E EB 0E 15 CD ...!l.p+q*k.....
01E0 05 00 C9 21 6E 1E 70 2B 71 2A 6D 1E EB 0E 16 CD ...!n.p+q*m.....
01F0 05 00 32 5F 1E C9 21 70 1E 70 2B 71 2A 6F 1E EB ..2...!p.p+q*o..
0200 C3 83 06 00 00 00 C3 4F 03 C3 24 05 2A 73 1E EB .....O...$.*s..
0210 0E 1E CD 05 00 F5 79 CD 8F 06 F1 C9 FE 20 CB FE .....y.....
0220 09 CB FE 2C CB FE 0D CB FE 7F CA 24 05 C9 0E 0D .....$.
0230 CD 15 00 0E 0A CD 15 00 C9 CD 8C 06 FE 0D CA 1B .....
0240 05 CD 1C 00 CA 39 00 0E 04 21 7A 06 36 20 23 0D .....9...!z.6 #.
0250 C2 4C 00 0E 05 21 7A 06 77 CD 8C 06 CD 1C 00 CA ..L...!z.w.....
0260 6A 00 23 0D CA 18 05 C3 58 00 3A 7A 06 FE 20 C9 j.#.....X.:#z..
0270 D6 30 FE 0A D8 C6 F9 FE 10 D8 C3 18 05 CD 98 06 ..O.....
0280 3D C2 18 05 EB 4E 23 46 79 05 04 C9 CD 7D 00 C2 =.....N#Fy.....}..

```

図7 オブジェクトプログラムの16進数表示の例

例えば LDA, (OEA55H) と書かれているソースプログラムは、3 EEA55と変換され、「EA55番地の情報(1バイト)をアキュムレータへ転送せよ」という機能を、この1行が表現している。図5、6からソースプログラムは比較的、人間の言葉に近い(英語的表現)の取り扱いが容易でそのプログラムの有する機能の情報の基本的部分は第3者でも理解しやすいであろう。通常、コンピュータを利用する場合、オブジェクトプログラム又は機械語プログラムだけあればコンピュータを稼動することができ、最近のビデオゲーム機のプログラムの盗用では、前述のファームウェア(ROM)に書き込まれている機械語を複製している点(不当に)に問題がある。EDPに使用されるコンピュータのプログラムは、ソースプログラムを販売又は公開しないで、オブジェクトプログラムだけを売る営業方針がIBMなどで多く見られる。ソースプログラムを見てその機能の詳細な情報又はアイデアの秘密を保護する一対策である。図7は、前述のCP/M-80に付属するユーティリティ・プログラムのDDT.COM(デバッグ・ツール)の最初の内容を左から、番地、16進数2桁で表示した機械語(16バイト分を一行としている)とその値に対応するアスキーコードを示す。このリストを見ても容易に機能情報を知ることは困難と考えられる。アスキーコードによる表示ではプログラムに含まれる熟語を読むことが可能であり著作権の表示が含まれている。米国製のプログラムであり米の著作権法でその

表示を義務づけているのでオブジェクトプログラムの中にまで表示させてある。このように、ソースプログラムとオブジェクトプログラムは性質が大きく異なり個別に保護の理論を考える必要がある様に思われる。

5. ソフトウェア保護のあり方

現在の法律制度の下でコンピュータ、ソフトウェアの主たる保護方法は、①著作権法による保護、②特許法による保護、③契約による保護、④刑法による保護、⑤不正競争防止法による保護などが考えられよう。どのような方法が適切であろうか、高名氏はこの問題に対し3つの視点を総合して決すべきことを述べている。①ソフトウェア権者に対し、出来る限り容易、迅速且つ経済的な保護が与えられるものであること。②ソフトウェア権者の保護と同時にそのユーザー(社会)の利益との調和が保たれること。③ソフトウェアの国際的保護が十分であること⁷⁾。中山氏はソフトウェアの保護の理由として、①先行投資者の保護、②重複投資の防止、③流通の促進を主張⁸⁾されている。両者共にソフトウェアの持つ特徴から、その開発に必要とされるアイデアを保護するのか、プログラム自体を保護するのかにより方法が異なると考えられる。著作権法で保護する場合、開発者が作成したソフトウェアを他人が無許諾で複製使用することは禁止できるが、他人が独自に作成したソフトウェアの実施することは差しつかえないとするか、特許権で保護される

場合、先行開発者の排他的な独占的実施権を認めるという態様がある。ソフト開発に必要とされるアイデアを保護するならば特許権を主張すべきであるが前述の様にソフトウェアの開発に莫大な投下資本を要するのでその回収を確保するという立場から、ソフトウェアを保護の目的物と考え、先行開発者以外の他人が不正に使用したり、盗用、複製することを禁止すれば現実的によい。しかし、アイデアの保護も将来的には必要であろう。国際的には、国連の世界知的所有権機関(WIPO)は1983年6月13日から17日までジュネーブの本部で専門会議を開きソフトウェア法的保護に関して保護条約草案が出された。それには防止すべき権利侵害を、①未公開のソフトウェアを権利者に無断で公開したり公開の手助けなどをすること、②未公開のソフトウェアを権利者に無断で保有したり、複製すること、もしくはその手助けなどをすること、③どのような形態手段にかかわらず、権利者に無断でソフトをコピーすること、④権利者に無断で、あるプログラムを作ったり、プログラム・リストを作成すること、⑤権利者に無断でプログラム・リストなどを利用して、同じか同様のプログラムやプログラム・リストを作ること、⑥権利者に無断でコピー、盗用、まねなどによるプログラムをマシン上で使用したり複製したり蓄積すること、⑦コンピュータ、ソフトやコピー盗用、まねによるソフトウェアを権利者に無断でレンタル、販売、輸入、輸出、リース、ライセンスしたり、営業目的でソフトを保有すること、⑧権利者に無断で賃金、販売、輸入、輸出、リース、ライセンスされた、ソフトウェアもしくはコピー盗用、まねなどによるソフトウェアを保有したり複製すること。以上の8点であるとしている⁹⁾。そこではコンピュータソフトウェアにおける知的創作の結果に直接関係する特許法、著作権法及びその他秘密保護法等を個別に検討している。そして基本目的はコンピュータ、ソフトウェアの創作及び開示に対する報酬として権利者に排他的権利の期間から生ずる利益を認めることではなく、単にコンピュータ、ソフトウェアの創作と拡布を促進し、他人の価値ある労作の成果の盗用を防ぎ、コンピュータ、ソフトウェアの取引の促進と権利者による一層の活用を促すための法的安定性を導入することに在る¹⁰⁾とされている。ソフトウェアの法的保護の必要性が高まるなか、昭和57年12月6日、テレビゲーム機の不正行為事件を審理していた東京地方裁判所民事部29部は、我国初の、プログラムの著作物性とその保護を認知する判決を下した¹¹⁾。この裁判は㈱タイトーが昭和53年8月中旬以降、販売していたマイクロコンピュータシステムを組み込んだゲームマシン「スペース・インベーダー・パートII」のROM化したオブジェクト・プログラムを㈱アイ・エ

ヌ・ジ・エンタープライゼスが顧客の注文により、タイトーに無断で顧客の手持ちのゲーム機のROMに収納、改造を行っていたとして著作権侵害の不法行為による損害賠償を請求した事件である¹²⁾。裁判所は原告の主張を全面的に認め、被告が改造したゲーム機1台あたりの利益2万円に改造数27を乗じた54万円が著作権の侵害行為により被告が得た利益の額とし、これと同額の損害を原告が受けたと推定されるとして損害賠償を命じた。原告の主張は次のとおりである。本件機械のコンピュータ・システムのROMに収納されている本件オブジェクト・プログラムは、本件プログラムに用いられている記号語(アセンブリ言語)を、開発用コンピュータ等を用いて、コンピュータが理解できる機械語(本件の場合2個の16進数を単位として表現される。)に変換した上、これを電気信号の形で本件機械のROMの記憶素子に固定して収納されていること、右記号語から機械語への変換は右両言語が1対1の対応関係にあるため機械的な置き換えによって可能であり、そこに何ら別個の著作物たるプログラムを創作する行為は介在しないこと、このROMに電気信号の形で固定して収納されている本件オブジェクト・プログラムは、ROMライター等の複製用具を用いて、他のROMに電気信号の形で収納することができるものであり、訴外電商サービスらは、右の手段で本件オブジェクト・プログラムを他のゲームマシンのROMに収納したこと、そしてROMは、プログラムを収納すると、一定の操作によってこれを消去しない限り、プログラムを記憶し続け、右ROM内の情報(プログラム)はコンピュータ・システムの電源スイッチが入ると中央演算装置(CPU)によって読みとられ、CPUが順次その命令を実行し、ゲームマシンの受像機面上本件ゲームの内容を映し出すものであることが認められる。右事実によれば、本件オブジェクト・プログラムは本件プログラムの複製物に当たり、訴外電商サービスらの本件オブジェクト・プログラムを他のROMに収納した行為は、本件プログラムの複製物から更に複製物を作出したことに当たるから著作物である本件プログラムを有形的に再製するものとして複製に該当する。この様な主張に対し、裁判所は、①原告のゲームが独自の新規なものであり、ソースプログラムも従前のものとは別個独立に、新たに作成されたものであること、②仕事の分析、解法の発明の積み重ねによる仕事の設計及びプログラミングという、本件プログラム作成の一連の過程であること、③記号語はその理解にある程度の専門知識と経験を必要とするとしても作成者の表現内容を第三者に伝達する機能を有するものであることを認定し、解法の発見及び命令の組合せの方法においてプログラム作成者の論理的思考が必要

とされ、又最終的に完成されたプログラムは、その作成者によって個人的な相違が生じるものであることは明らかであるから、本件プログラムはその作成者の独自の学術的思相の創作的表現であり著作権法上保護される著作物に当たると、判断を下した。ソースプログラムに著作権を認め、本件プログラムを機械語に変換し、記憶素子に固定しているオブジェクト・プログラムは「何ら別個の著作物を創作する行為は介在していない」という理由で、オブジェクト・プログラムに著作権を認めず、ソースプログラムの複製物に該当すると判断を示した。そしてオブジェクト・プログラムから他のROMへ収納する行為を著作権違反行為としての複製を認めた。この判決は我国のソフトウェア法的保護の方向を示すものとして注目され今後のあり方に大きく影響を与えるものであろうと考えられる。

6. 結 言

今後の社会において果たすコンピュータの役割は増々重要なものとなることが予想され、コンピュータのソフトウェアがコンピュータ・システムに大きな経済的価値を持ち、種々な知的創意工夫が組み込まれるようになる。その法的な保護の確立が社会的に急務な課題であることは明白である。そのための基礎的考案としての権利と保護の対象物であるソフトウェアについて述べ、我国の法的な方向づけに大きな影響を与えるであろう東京地方裁判所の判決について考えてみた。しかし、ソフトウェアの持つきわめて特殊な性質や技術的問題点が十分に考慮されて法的な保護が確立すればソフトウェアの開発投資の刺激にもなりソフトウェア技術の発展にもつながるのであろう。あまり急ぐあまり形式的な規制や現行の法律の拡大解釈のみが先行するのではなく、実質的な、ソフ

トウェアの内部に秘められている創意工夫の努力に対しても、ある部分についてはなんらかの保護が必要であろうと考える。前述の法工学的アプローチはその目的のためソフトウェアの有する自然科学的原理に基づく工学の基礎の止に、又立法を補助するためのアプローチとして重要であることを最後に強調したい。

引用文献

- 1) 日本情報処理開発協会編「コンピュータ白書1983」p.3, 1983
- 2) 阿部圭一「ソフトウェア入門」p.12, 1983
- 3) 工藤市兵衛「工法学の環境法へのアプローチ」愛知工業大学研究報告, No.18, p.126, 1983
- 4) 池田敏雄「電子計算機概論」p.205, 1975
- 5) Roy N Freed「アメリカ合衆国におけるソフトウェアの法的保護」ジュリスト, No.755, p.60, 1981
- 6) F. P. Brooks, Jr「ソフトウェア開発の神話」p.28, 1980
- 7) 高石義一「ソフトウェア保護の法的課題」ジュリスト, No.784, p.20, 1983
- 8) 中山信弘「コンピュータ・プログラムの法的保護」ジュリスト, No.784, p.15, 1983
- 9) 「ソフトウェア権法の行方」日経コンピュータ, 1983. 8. 8号, p.113, 1983
- 10) 綾谷暢男「WIPOのコンピュータ・プログラムの保護モデル条項」ジュリスト, No.784, p.27, 1983
- 11) 永島考明「東京地方裁判所著作権判決の意義」コンピュータピア, p.13, 1983
- 12) 判例時報, 1060号, p.18

(受理 昭和59年1月17日)