

奨学金業務プログラムの作成

松木 孝幸

(平成5年10月7日受理)

Computer Programming for Scholarship Works

Takayuki MATSUKI

(Received October 7, 1993)

1. はじめに

東京家政大学の事務には Unisys 社製のオフィスコンピュータが1台、NEC 社製のパソコンが12台、その他にワープロ専用機が25台余り導入されている。オフコンは教学(学籍、履修、成績)、就職、図書館(図書管理、貸出、返却、検索)、財務、入試、保健、及び同窓生の名簿作成等の業務に使われている。文書作成は主にワープロ専用機によってなされており、パソコンに関してはそこそこに使われているが、十分に活用されているとはいえない。

これまで、東京家政大学及び短期大学のコンピュータに関してはずべて事務の企画調査室が関わってきている。この理由としては、ひとつには当校には計算機の専門家が当初おらず、企画部の人間を内部養成してそれに当たったのであり、組織のコンピュータ化にあたっての初期及び途中過程で常に起きる現象である。多くの会社においても同様に企画部という名の部署で会社内のコンピュータ化を始め、大きな会社或いは組織では情報処理部門を新設しそこへと業務を引き継いでいくことになる。

当校では情報処理の科目を担当している教員は存在するが、事務の各部門の計算機に関しては殆ど関与していない。企画調査室では、毎年講習会を開いて事務に携わる個人の能力を高めようとしているが、なかなか各部署が独立してパソコンの活用を励むという状況にはなっていない。能力のある個人は存在するが自分の所属する部署の仕事で手一杯であり、部門を越えて手伝うということは現在のところ見られない。今のところ各部署が個々の人材に頼み込んで、助けてもらうという状況である。

栄養学科 情報処理研究室

教員間ではカリキュラムの大綱化及びそれに伴う組織の見直しの問題等が指摘されているが、事務部門でも役所に習った各種原議書による許認可・承認の方法の是非、大学及び短大の部門新設(環境情報専攻、国際コミュニケーション学科)に伴う事務に即応できる体制があるかという問題がある。世界全体が変革の時機に遭遇しているとき、ひとつの組織である当校もその動きに無関係では有り得ないであろう。時代に取り残されずに、自ら動く事のできる体制或いは組織とはどのようなものであるかは色々議論があろうが、例えば、一時期だけでも能力のあるものを集めて部門間に拘らずにコンピュータ化を進める部門を新設する等の、事務の組織の見直しも一つの手段であろう。当校に相応しい組織というのがどのようなものであるべきか、このままでよいのかという重要な問題は慎重にであるがかつ早急に事を運ぶべき事柄である。

東京家政大学学生部が、毎年4月から6月に掛けての期間と10月の一ヶ月にこれまで手作業で行ってきた、各種奨学金候補者の選抜作業の一部のコンピュータ化を、私は試みた。その為の道具としてパソコン上で動くデータベースソフト1)を使い、学生部の担当者に対する面接による要求仕様定義から始まり、実際のコーディング、およびそのプログラムのテスト期間終了までの作業は、平成4年12月中旬より始め平成5年6月始めまで要した。実働日数は延べ2ヶ月・人程である。

以下にプログラム完成までの過程、及び業務のコンピュータ化を事務担当者だけで行なうときの注意点等をまとめて書き記す。

2. プログラム作成方法

プログラム作成方法としては、ウォーターフォール・モデルとプロトタイプ法とがある。ウォーターフォール・モデルは従来からよく使用されている方法で、上流から下流に向かって滝が流れるように各作業を順番に行なっていく方法である。各作業には以下のものがある。①要求仕様設計、②概略仕様設計、③詳細仕様設計、④試験仕様設計、⑤プログラム仕様設計、⑥プログラム作成、⑦テスト・プログラム作成、⑧モジュール試験、⑨統合試験、⑩保守仕様作成および保守の実施

この方法で重要なことは、滝の上流での作業が下流を支配するので、上流工程で高品質の作業をしないと大変なことになるということである。すなわち、最上流の要求仕様時に、要求項目を取り違えていたり、そもそも一環性のない、あるいは矛盾した要求を取り込んでしまうと、下流の工程でいくら高品質の仕事をしても取り返しがつかないということである。上流工程が下流工程の品質の上限を定める、というのがこのモデルのポイントである。

ウォーターフォール・モデルに対する最大の批判は、この方式では後工程での問題の原因が前工程にあったときの処置がうまくゆかないという点にある。この点は、プログラムができる前に、それに対する要求を完全に並べたてることが難しいという体質的な問題のために、しばしば指摘される。要求仕様通りにソフトウェアを作ってみて、初めてその要求の一部が間違っていたり、不用だということがわかって他の要求を追加したりすることが多い。こういう批判を受けて、代替案として、次に述べるプロトタイプ法という手段が用いられるようになってきた。

プロトタイプというのは、試作品のことである。普通の製品開発では、自動車にせよ家電製品にせよ、本当の市販品を作る前にこのような試作品を作っては、実際の現場、あるいは研究開発の場でさまざまに試験、試用して本番の製品を完成させる。ソフトウェア作成にもこの手法を応用すべしというのが、このプロトタイプ法の主旨で、仕様の作成とプロトタイプの作成とを同時に行う手法が特徴とされている。

私は、初期の段階ではウォーターフォール・モデルを頭に置いてプログラムの開発を始めたのであるが、開発を進めるにつれて結局プロトタイプ法に落ち着いていった。

その最大の原因はやはり要求仕様設計が、最初の面接によっては最終的なものとならず、プロトタイプを作成しながら徐々に完全なものに近づいて行かざるをえなかったためである。これは、私自信が業務に精通していないということと、実際の業務担当者も普通の業務をコンピュータ化した場合になにが省力化可能であるかを把握していないためでもあった。最初の面接を注意深く行えば、最終的な要求仕様を一番初めの段階で設計できたかも知れないが、それは非常に時間のかかることであったに相違ないし、従ってそれは最も有効な方法とは言えないであろう。

3. 要求仕様設計

始めに狭山校舎の、後に板橋校舎の学生部に所属する担当者に面接し、最終的な仕様を設計することから始めた。しかし前章で述べたように、要求仕様定義はプログラムの完成後も変更が相次いだ。これは面接の際の質問事項の設定の仕方に今後の課題を残した。またデータベースソフト1)の能力の問題もあり、仕様が制限を受けたり、あるいはその能力を熟知するにつれて仕様を拡張することもあった。主な仕様は以下の通りである。

- I) 候補者についてのデータの入力を省力化するために、候補者の学生証番号を入力するだけでその他の必要事項も自動的に入力されるようになる。
- II) 学生証番号から、各学生の個人データを教務課の保存しているファイルから複写し、さらに各学生の奨学金関連の必要データを入力する。
- III) 各候補者のデータに基づいた各項目の評点と総合点を自動的に計算する。
- IV) 各評点と総合点による候補者の整列を各項目別に行なう
- V) 整列データをファイルに保存する。
- VI) 整列データを表の形で印刷する。

上記のうち特にI)の項目については試行錯誤を繰返し、どれが最善の入力方法であるかは、⑧モジュール試験、に入ってから最終決定した。III)については、もともと科学の素養のある者にとって解決する(データ・ベースソフトの用意した関数を使って必要な式を定義する。)のは容易であるが、通常の事務担当者がこれらを行なうのは困難であるため、この部分だけは専門家による補助が必要であろう。IV)はデータベース本来の作業である。

4. 詳細仕様設計

上記の仕様に基づいて、さらにソフトウェア1)の特徴を考慮して詳細設計を決定していった。このとき必要となるファイルは、教務課からは

①学籍マスターファイル

②部科専マスターファイル

の2つである。①は学生の学籍関係のデータ・ファイルであり、②は各学部、各学科、各専攻のコードが規定されている。また、1)のソフトが用意しているファイルの種類は

③一括処理ファイル

④表ファイル

⑤帳票ファイル

の3つである。③は③、④と⑤の全てのファイルを使いながら、処理を記入しておくファイルである。主なプログラミングはこのファイルを作成することである。④は項目を設定してその項目のデータを記録しておくファイルであり、かつ項目間の関係式をも記入することが出来る。即ち、一種の表計算ソフトがデータベースの一部となっているのである。但し、表計算ソフトと異なり、一個々のセル同士の関係はつけられない。④のファイル作成のためには項目間の関係式の導出が必要であり、③のプログラミングに次いで時間を必要とした。⑤は入出力画面を設計して保存しておくためのファイルである。今回は、入力画面を設計しそれを保存しておくためのファイルとして使用した。この分類では教務課のファイル①と②は④の表ファイルとして扱われる。

③のファイルは、起動用と本来の処理用と2つのファイルを作成する。起動用ファイルの目的はパスワード入力画面を起動するためである。④のファイルは、教務課から与えられる①と②以外に、作業用ファイルが1つに、出力用データを保存する6つのファイルと、日本育英会から与えられる表を更新するための保存用のファイル1つを作成する。⑤のファイルは、学生証番号と、候補者の各データと、日本育英会の表の更新用の3種類の入力用ファイルを作成する。以上の具体的な詳細仕様は以下の通りである。

I) 画面はメニュー形式にし、各命令は矢印キーで選択をする。各画面では、必ず第一層の画面に戻れるメニューを含める。(キャンセル用メニューである。)

II) メニュー画面は3階層にする。(図1参照)

III) 学生証番号入力画面を設計する。データは表ファイルに記録する。

IV) 学生証番号をキーとして、必要データを学籍マスターファイルから作業している表計算ファイルに、部科専マスターファイルを通じて複写する処理を、③の一括処理ファイルにプログラミングする。

V) 候補者自身のデータ項目以外に選抜を評価する項目を設定し、その項目の値を他のデータ項目の値から計算する関数式を導出し、表ファイルに埋め込む。これと同時に、他のデータ項目の入力画面も設計する。

VI) 6つの出力ファイルに、各基準キー項目毎に昇順或いは降順に並べ替えたデータを保存する処理と、印刷形式を設定するための処理を、③の一括処理ファイルにプログラミングする。

VII) 出力ファイルを印刷する処理を、③の一括処理ファイルにプログラミングする。

5. その他の作業

ここで行なったプログラミングは規模が小さいので、2. プログラム作成方法であげた幾つかの作業のうち、②概略仕様設計、④試験仕様設計、⑤プログラム仕様設計は行なわなかった。⑦テスト・プログラム作成と⑧モジュール試験と⑨統合試験は、4. 詳細仕様設計であげた各項目のプログラミングをモジュール化して、即ち各機能を関数としてまとめあげてプログラミングすることによって解決した。なお⑦と⑨を実行する際、去年の1年のデータをテスト・データとして用いた。最後に、⑩保守仕様作成および保守の実施では、一括処理ファイルを作成する際にファイル中に現れる一定した名前(例えば、ファイル名や項目名等)は、ファイルの一番先頭で変数に代入し、それ以降はそれらの変数で処理を記述し、名前の変更等を容易に行なえるようにした。また同様にして印刷形式のパラメータもファイル先頭で変数に代入した。それ以外の保守としては、4. で述べたように日本育英会から渡されるデータの更新をするためのプログラミングも行なった。また狭山校舎の学生部はマニュアルを現在作成中である。

メニュー画面

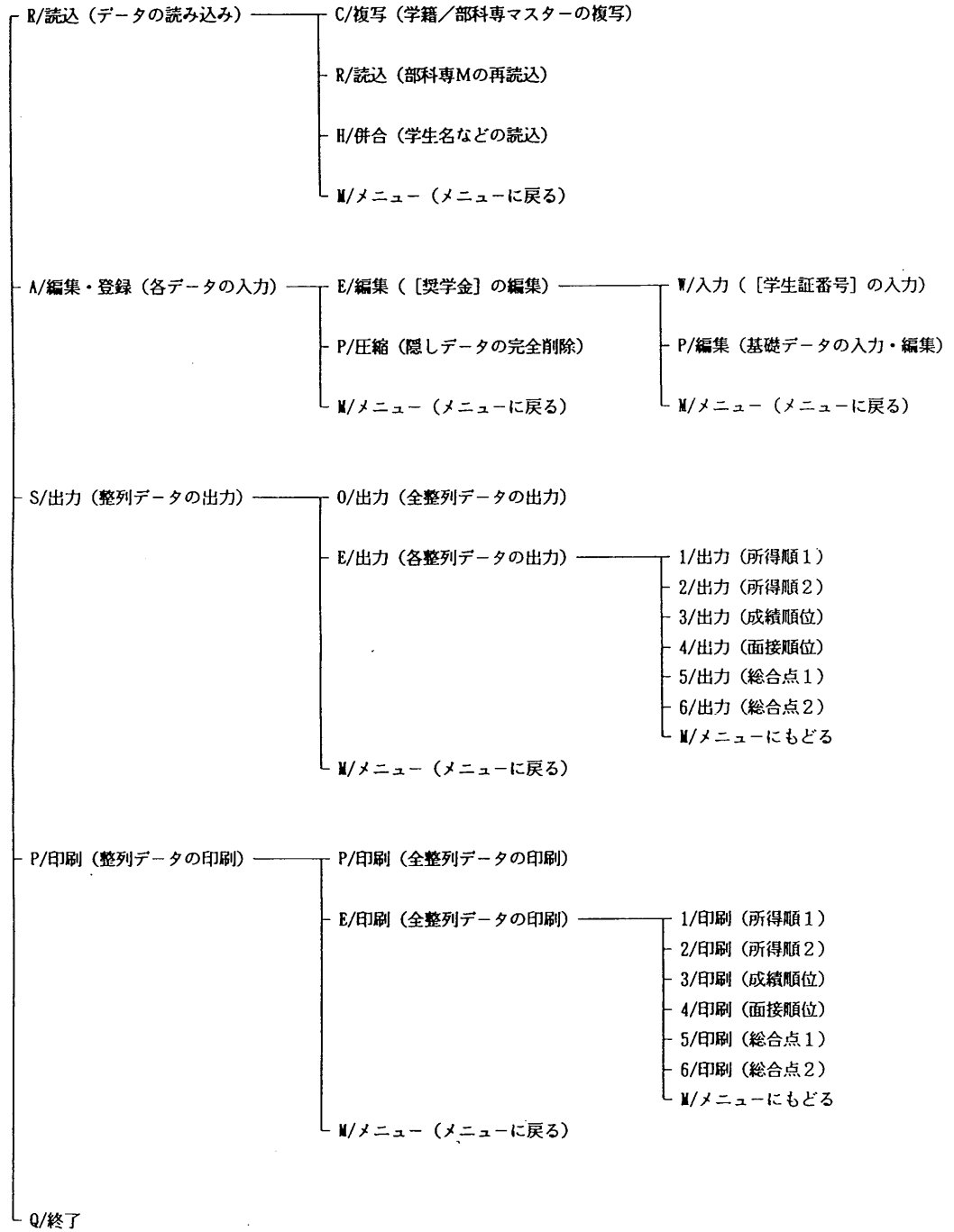


図1 階層化されたメニュー

6. まとめ

この「奨学金業務プログラムの作成」では一括処理ファイルのプログラミングは817行であった。表ファイルでは、キー項目の「学生証番号」と、「部科専マスター・ファイル」を通した「学籍マスター・ファイル」の項目を関係付ける関係式の導出と、候補者の順番付けのための評価式の導出が時間を要した。プログラム作製のために使用した参考書は2)と3)である。

次に、1)のソフトの欠点或いは限界を以下に記す。

- I) まず一括処理ファイルがテキストファイルでないという点である。これは、このソフト組み込みのエディタでこのファイルを作成しなければならないという制限がでてくる。大きなソフトである1)が起動できるコンピュータのあるところでは、作業が出来ないということになる。
- II) 表ファイル作成においては、表ファイルに組み込む式が文法に完全に合ったものでなければファイルを保存することが出来ない。即ち、式を途中まで作成して保存し、続きを明日行なうということが出来ない。

III) 一括処理で使用できる変数の種類が少ない。例えば、配列変数が許されない。

IV) 帳票ファイル作成においては、使える関数が少なすぎて画面作成の大きな制限となった。例えば、入力項目の最後から最初の入力項目へ戻るということが出来ない。

このプログラムは日本育英会奨学金用に作成されたが、東京家政大学では他に8つの奨学金制度があり、それらの判定基準は日本育英会と同じものを使用しており、したがってこのプログラムを流用できる。最後に、このプログラミングをきっかけとして他の事務部でも同様な動きが出ることを期待する。

参考文献

- 1) 桐 Ver. 4, 管理工学研究所 (東京), 1993
- 2) 管理工学研究所: 桐 Ver. 4 リファレンス 1, 2, 管理工学研究所 (東京), 1993
- 3) 岩崎治: 桐 Ver. 4 ハンディ・マニュアル, ナツメ社 (東京), 1992