

時間割作成支援ソフトの開発Ⅱ

村井利行(理科・物理)
十九浦美里(数学科)

1. はじめに

開発中の時間割作成支援ソフト*1を大幅に改良した。使用言語は昨年同様 VisualBasic 2010 である。機能の充実のほか、マウスクリックのみでほとんどの操作が行われるよう、操作性の向上にも努めた。また、昨年からの課題であった「科目情報の変更」への対応の面も、完全とは言えないが、かなり良くなった。このような改良点について、2. で概要を紹介する。

今回、特筆すべきは、本年度時間割作成担当の十九浦教諭(数学科)が、このソフトを使って実際に2013年度の時間割を組んだことである。昨年度は「自分だけが使うソフト」だったのだが、今回のようになると、操作性の点だけでも大幅な改良が必要であった。せっかくの機会だったので、なるべく私が手伝うことなく動かしてもらえるよう、工夫に努めた。さいわい、十九浦教諭のみ込みの良さの御陰で、かなりのレベルまで自立したソフトとして動かしてもらうことが出来た*2。また、いろいろと改良の注文も出してもらい、極力それに応える努力もした。私の感覚としては「信じられない速さ」で時間割表のコマを動かしていて、結局「様々な操作を試してはバグを見つけ出ししていく」という作業をしてきていたのである*3。つまりソフトの“テストパイロット”をやってくれていた。実に有り難い存在であり、実質的に共同開発者だった。本稿においても、4. において、ソフトの使い勝手や今後の改良案等を述べていただく。

何人かの方々から「コンピュータでどうやって時間割を組むの?」「コマが入る入らないをどうやって判断するの?」といった質問を受けた。プログラムのコード自体をここで披露しても、かなりの行数でもあり多くの方には意味不明の代物と思われるので、今回のプログラムにおいて要となっていた手法(考え方)のひとつを3. で紹介しようと思う*4。

以下の文章には、本ソフト専用の造語も度々出てくるが、昨年度の紀要*1に説明してあるものは「*1」を付けるに止めた。ご容赦願いたい。

昨年度に引き続き今回の開発も、お茶の水女子大学教員研究費の支援を受けて遂行したものである。

2. 改良した点

今回、昨年度使用したソフトを元に多くの改良・機能充実に努めた。細かな点までは書き切れないが、比較的大きな改善点について記しておこう。ただし、「細かな点」の改良があったからこそ、大きく改善できたというのも確かだった。

(1) 操作性向上

1. でも述べたが、まず操作性の向上を目指した。これには、いわば人間工学的な発想が重要である。いかに使い易くするかである。基本方針は、マウスクリックのみで操作していけるようにすることであった。昨年度は、例えばプログラムを実行する前に使用環境に応じてコードの内容に若干の手直をしていたが、今回それは御法度であり、メニュー画面を用意し、マウスクリックで項目を選択していく形に整えた。メニュー画面でクリックするボタンの数をなるべく減らすようにもした。例えば、時間割表を Excel のシートに“清書”する場面で、はじめの内は Excel を立ち上げるボタンを用意していたが後に、“清書”のボタンをクリックすれば Excel が自動で立ち上がるようにした。また、時間割表にコマを入れていく作業において、複数のウインドウを操作する必要があるのだが、「各ウインドウの表示ボタン」のみを集めた小さな1つのウインドウを用意した。これも操作性の向上につながったと思われる。

操作性向上の作業は、プログラミング自体より前述の通り、人間工学的なセンスがものを言う。市販のソフトなどに比べると、ボタンの配置などデザインの面でまだまだ劣っている。もっと洗練された操作性を追求したいものである。これは今後の改良の課題とする。

(2) 科目情報変更への対応

時間割表にコマがかなり入った時点で、科目情報に変更が生じることがある。例えば、選択科目の開講中止や非常勤講師の人数変更などである。このような場面でも、それまで作ってきた時間割表を使って作業を進めていくのが普通である。しかし、これが「手作業」とコンピュータを使った作業の違いで、意外とやっかいなのである。問題は時間割表をどんなデータとして収録してあるかにかかっている。昨年度は時間割表に入れた「枠」*1の番号を収録していたが、これが良くなかった。「枠」の番号は科目情報変更の影響を大きく受けるケースが多いのだった。今回はコマの番号を収録するように修正し、科目情報変更にもほぼ完全に対応できるようにした。実は、収録データの変更は単にそれだけのことで済まないのである。と言うのも、プログラム自体は既に変更後の科目情報に基づいて動いているが、読み込んだ時間割表のデータは変更前の科目情報に基づいているからである。この点の整合性をとる作業を仕組む必要があった。

(3) 交換機能

時間割作成作業では「コマの交換」が頻繁に行われる。特に、すべてのコマを一応入れた後、あるいはそれに近い時点で必要になる。昨年度はこの作業に最少で5回のクリックを要していた。これを3回で済ますように改良した。1回は「交換モード」の指定のため、あとの2回は該当する2つのコマのクリックである。この改良に伴い、極めて大きな余録にあずかった。「交換可能性」を表示するようになってきたからである。コマを交換するからには、その交換は禁則に抵触してはならない。そこで、1つめのコマをクリックした時点で、そのコマと交換が可能(禁則に抵触しない) コマを探し出す作業を組み入れたのである。これは“大ヒット”であった。単にコマの交換を行う機能としてではなく、交換できるコ

マを探し出す機能として極めて有用な働きをしてくれたのであった。

(4) 記録機能

時間割表を収録する機能は以前から備えていたが、クリック 1 回毎の作業記録を自動的に残すようにもした。ワープロ・表計算等でお馴染みの「戻る」の機能と似たもので、「ログ記録」と呼んでいる機能である。ログ記録の ON・OFF ボタンを用意したので、必要と思われる時点で記録を残すことができる。この機能は時間割作成作業も終盤に入った頃に組み込んだもので、残念ながら今回の作業にはほとんど役立たなかったが、要望に応じて用意したものであり、来年度はきっと役立つだろうと思っている。小刻みに作業過程を戻したり、あるいは再現していく機能だが、実際にこれを働かせてみると「どんな操作でどのコマをどうしたか」という情報が表示されると助かることも分かった。現時点でもある程度の表示は出すようにしてあるが、来年に向けて「直感的に分かりやすい表示」を用意しようと考えている。どのようにすると役に立つか、それを実現するにはどのようなプログラム組めばよいか……挑戦しがいのある、なかなか魅力的な課題である。

3. コマが入る入らないの判断：その考え方

何人かの方が「どのようにして時間割を組んでいるのか」という質問を投げかけて下さった。ここでプログラムの仕組み全体を文章で表現するだけの力量はないが、「これは良いアイデアだ!」と自分で感じている事柄をかいつまんで説明することは出来ると思う。そのアイデアとは、少々大風呂敷に言うと、物理学の「場の考え」の応用である*5。時間割表にコマを入れる際、「入れられる」場所と「入れられない場所」がある。その判断をどのように行っているか、その際にどのように「場の考え」を使っているかを説明する。「そのアイデアをプログラムでどう表現したのか?」と尋ねて下さる方もあった。そこで、たとえ話や抽象化した一般的な説明ではなく、プログラムコードを示しながら、具体的に説明してみようと思う。なるべく分かり易く説明するつもりである。3 ページ程度の分量なので、プログラミングなどには縁がないという方にも、是非お付き合い願いたい。「プログラムコードってこういうものか」と感じて頂けるだけでも幸いである。

(1) 禁則の例

例として図 1 のように、時間割表の月曜日 1～3 時限の部分で説明する。図 1 では教員「村井」の「2R *6」の授業が 2 時間目に入っている。この授業はとりあえず『物理基礎』と考えていただく。

ここで、以下の説明で用いる記号を次のように定義しておく。

tc : 教員番号 d : 曜日 (月, 火, 水, …順に 1,2,3, …) p : 時限
k : コマ番号 (各コマに付けた番号で、各教員ごとに通し番号を付けている)
ch : Charge (後述) の番号

上記の記号を用いると図 1 の状況は

tc = 3、d = 1、p = 2 として「座標 (tc, d, p) にコマ [2R] が入っている」

ということになる。科目『物理基礎』は時間割表には表示されないが、プログラムの中では、図1のコマ [2R] に付随する属性となっている。

図1の例の場合、たとえば次のような禁則がある：

禁則① 座標 (3, 1, 2) にはどんなコマも入れられない。

禁則② 座標 (tc, 1, 2) にクラス2Rの授業を入れられない (tc = 1, 2, 4, …)。

禁則③ 座標 (3, 1, p) にクラス2Rの『物理基礎』を入れられない (p = 1, 3, …)。

		月		
tc	教員名	1	2	3
1	...			
2	...			
3	村井		2R	
4	...			

図1 時間割表の例 その1

		月		
tc	教員名	1	2	3
1	...			
2	...			
3	村井		10	
4	...			

図2 ch=1のChargeが張るB場 (ch=1) のイメージ

(2) Charge とB場

上記の禁則をどう表現するかを説明するが、そのためには多少の準備が必要である。すべてのコマには (もちろん図1のコマ [2R] も含め)、私が Charge と呼んでいるある属性を持たせている。Charge には多くの種類が用意しており、実際のプログラムコードでは Charge の種類を番号 (記号 ch) で区別している。例えば、ch = 1 の Charge は、すべてのコマに持たせている Charge なのだが、「tc = 3 の教員の k = 1 のコマは、ch = 1 の Charge を持つ」ということを

tc = 3, k = 1, ch = 1 として「komaDat(tc, k).StrengthOfCh(ch) = 1」*7

と表現している。「」内の式の右辺の「1」は「Charge の強さが1」という意味である。Charge の強さは1か0で表し、Charge を持っていれば1、持っていなければ0としている。左辺の komaDat(tc, k) は「教員番号 tc の教員の番号 k のコマ」を意味している*8。

実は、ch = 1 の Charge は上記禁則①と結びつけている。コマ [2R] の持つ「ch = 1 の Charge」が座標 (3, 1, 2) の場所に「場を張る」と考えるのだ。これを

tc = 3, d = 1, p = 2, ch = 1 として「B_Field(tc, d, p, ch).Value = 10」

と表現する。図2に「ch = 1 の Charge が張る場」を具体的に示した (これは時間割表には表示されない)。B_Field のBは「Barrier (バリヤー) 場」の略で、上式はいわば“バリヤーを張る”の意味であり、上例では場の値 (磁場の強さのようなもの) を10に設定している。値「10」は単にそう設定しただけで、数値自体に特別な意味はない。以下、例えば ch = 1 の Charge が張るバリヤー場を単にB場 (ch = 1) と略記する。

(3) Charge がB場から受ける力と「判断」

図1の状況で、コマ [2R] が置いてある場所にコマ [2K] を置こうとしたとしよう。これは禁則①に抵触する。コマ [2K] も $ch = 1$ の Charge を持つ。そしてコマ [2R] が置いてある場所には既にB場 ($ch = 1$) が存在している。そこで次のような量を計算する (この量を私は力と呼んでいて記号 Force で表している。 $k = 3$ はとりあえず振った数字)。

$$tc = 3, d = 1, p = 2, ch = 1, k = 3 \text{ として}$$

$$\text{Force} = \text{komaDat}(tc, k).\text{StrengthOfCh}(ch) \times \text{B_Field}(tc, d, p, ch).\text{Value}$$

上式を言葉で書くと：

$$\begin{aligned} \text{力} &= (\text{コマ [2K] が持つ } ch = 1 \text{ の Charge の強さ}) \\ &\times (\text{コマ [2R] が置いてある場所における B場 } (ch = 1) \text{ の強さ}) \end{aligned}$$

であり

$$\text{Force} = 1 \times 10 = 10$$

と計算される。Force = 10 は「入れることが出来ない」を意味する*9。

7行上に記した式を再度示すが、斜体で強調した2つの ch に注目してほしい。

$$\text{Force} = \text{komaDat}(tc, k).\text{StrengthOfCh}(ch) \times \text{B_Field}(tc, d, p, ch).\text{Value}$$

ch は Charge の番号であり、上記の2箇所には常に同じ ch を代入するのである。あるコマの「番号 ch の Charge」は、他のコマの「番号 ch の Charge が張るB場 (ch)」から力を受けるのである。

禁則①は単純すぎるので、次に禁則②のような若干複雑な状況の説明をしよう。禁則②には $ch = 21$ の Charge を結びつけている。「21」はクラス名「2R」に由来する。コマ [2R] はこの Charge を持つ。即ち図1のコマ [2R] は

$$tc = 3, k = 1, ch = 21 \text{ として}$$

$$\text{komaDat}(tc, k).\text{StrengthOfCh}(ch) = 1$$

という属性を持つ。図3に、 $ch = 21$ の Charge が張るB場 ($ch = 21$) を示した。つまり

$$tc = 1, 2, 4, \dots, d = 1, p = 2, ch = 21 \text{ として}$$

$$\text{B_Field}(tc, d, p, ch).\text{Value} = 10$$

である。いま、図1の時間割表において、 $tc = 2, d = 1, p = 2$ の場所にコマ [2R] を置こうとすると、どうなるか？これは禁則②に抵触する。いま入れようとしているコマ [2R] も「 $ch = 21$ の Charge」を持つ。つまり

$$tc = 2, k = 5, ch = 21 \text{ として}$$

$$\text{komaDat}(tc, k).\text{StrengthOfCh}(ch) = 1$$

のようになっている ($k = 5$ は、とりあえず振った数字に過ぎない)。したがって、このコマ [2R] が $tc = 2, d = 1, p = 2$ で受ける力は

$$tc = 2, d = 1, p = 2 \text{ として}$$

$$\begin{aligned} \text{Force} &= \text{komaDat}(tc, k).\text{StrengthOfCh}(ch) \times \text{B_Field}(tc, d, p, ch).\text{Value} \\ &= 1 \times 10 = 10 \end{aligned}$$

となり「入れることは出来ない」となる。一方、同じ場所にコマ [2K] を入れるのなら、禁則②には抵触しない。コマ [2K] は「ch = 21 の Charge」は持たないから

tc = 2, k = 8, ch = 21 として

komaDat(tc, k).StrengOfCh(ch) = 0

である。ただし、komaDat(2, 8) は上記のコマ [2K] を意味しているとする。さて、このコマ [2K] が B 場 (ch = 21) から受ける力は

tc = 2, k = 8, ch = 21 として

Force = komaDat(tc, k).StrengOfCh(ch) × B_Field(tc, d, p, ch).Value
= 0 × 1 = 0

となって、コマ [2K] は、この B 場 (ch = 21) からは「力を受けない」。もし、コマ [2K] がもつすべての Charge が「力を受けない：Force = 0」のなら、コマ [2K] は「入れられる」と判断されるのである (図 4)。

tc	教員名	月		
		1	2	3
1	・・・		10	
2	・・・		10	
3	村井			
4	・・・		10	

図 3 ch = 21 の Charge が張る B 場 (ch = 21) のイメージ

tc	教員名	月		
		1	2	3
1	・・・			
2	・・・		2K	
3	村井		2R	
4	・・・			

図 4 時間割の例 その 2

(4) Charge と B 場の効用

図 1 において禁則③に結びついているのは ch = 2101 の Charge と設定してある。「2101」の「21」は「2R：2年欄組」に由来する。「01」は「村井が担当する番号 1 の科目」という意味で具体的には『物理基礎』のことであり、これは科目情報として予め設定してあるものである*10。図 1 のコマ [2R] が持つ「ch = 2101 の Charge」が張る B 場 (ch = 2101) を図 5 に示した。この B 場 (ch = 2101) により、禁則③が守られるのである。

以上のように、様々な禁則はそれぞれに結びつけた Charge (番号 ch) とその Charge が張る B 場 (番号 ch) との相互作用として表現される。Charge には多くの種類がある。例えば、2 時間続きの授業に関わる Charge、授業を行う場所に関わる Charge、体育実技の授業に特有の制約に関わる Charge 等々、今回のプログラムで用いた Charge は恐らく数十種類に上る*11。

いま「恐らく」と書いたことには意味がある。実際、Charge の数など数えていないのである。それは、どんな Charge でも、その Charge が張る B 場はほんの数種類の類型に分類でき、案外システムチックにプログラミングしていけるからだ。つまり、上述の「Charge

とB場」の方法は、様々な制約（禁則）に対して、同じような手法で対処することを可能にしているのである。

tc	教員名	月		
		1	2	3
1	・・・			
2	・・・			
3	村井	10		10
4	・・・			

図5 ch=2101のChargeが張るB場(ch=2101)のイメージ

(5) 実際の時間割作成では

実際にプログラムが行っている作業は、「コマ」を単位としているのではない*1。その意味では、ここまでの説明は、基本ではあるが単純過ぎるのである。実際は、「コマ」の組み合わせである「枠」*1を単位として作業をしている。したがって、「枠を構成している各コマが張るB場」の重ね合わせが「枠が張るB場」になり、このB場が他の「枠」に対して力を及ぼすわけである。一見複雑そうではあるが、実はコマを「原子」と見なし、「枠」を、原子によって構成される「分子」と見なして考えていくと案外単純なのである。

4. “テストパイロット”からのメッセージ

開発途上ソフトの“テストパイロット”としてご協力いただいた十九浦教諭から、ソフトの使い勝手等に関するメッセージを受け取った。アンケート形式での回答である。

(1) このソフトを使ってみようと思った動機は？

時間割を作成することが初めてで、また本校のカリキュラム（3年生の選択授業をはじめとして）が複雑であることはわかっていたので、藁にもすがる気持ちで支援してもらえらるなら、ぜひ、という感じで、使わせていただこうと思いました。

(2) 実際に作成作業全体を遂行した立場から、作業の各過程について、その内容の概略、所要時間、負担の程度を記して下さい。また、所要時間や負担はソフト操作に慣れることでどのように変化しましたか。

① 科目情報の入力：各教科から提出された、個人別の担当科目、単位数、条件などを入力。すべて終わってから考えると、この作業が一番大変であったように思う。理由としては、連携（同じ授業を複数の教員がもつ授業）や、合同授業、使用教室の条件など、様々な側面から条件がここに集約されていくことになり、複雑で神経質にならざるを得ないのは仕方ないように思う。ただ、ここで苦労しておくことで、自分の中でも、時間割を作る上での様々な条件が頭の中の一応は入っていくし、「時間割を作る」

という意識が高まっていったように感じる。よって必要な苦労ではあると思う。

- ② ①の入力後、個人別に担当科目の入った表を出力し、会議で各教員に確認していただく。⇒間違いが数か所みつきり、訂正をいれた。

所要時間：5，6時間（訂正なども含めて）

- ③ 時間割作成：以下の順番で（基本的には、条件の厳しいものから）授業をいれていく。その際、支援ソフトの機能の1つである《自由度》の表示がたよりとなる。

1. 3年選択授業
2. 3年必修授業
3. 2年生連携科目
4. 1年生連携科目
5. 1，2年生の残りの授業（非常勤の先生の授業や、体育、家庭科など条件が厳しいものから順にいれていく）

ソフト側が、入力された条件を加味して、コマをいれられる箇所を提示してくれるので、まずそれを頼りに作業を進める。特に最初は、自分でいれようと思ったところに、「いれられません」表示がでており、「なぜ？」と思ったが、よくみると確かにいれられないことがわかり、なるほど、となったことが多々あった。まさにソフトに教えてもらっている状況である（しかも、途中から改良されなぜ入れないのかまで表示してくれるようになった）。その作業の中で、条件をみる優先順位などが自分の中で確立されていき、最終的には、ソフトでは吟味できない条件についても、自分で考えられるようになっていった。

所要時間：5時間～12時間

（1作目は、作業に慣れていないことに加え、要領も何もわかっていなかったので12時間ほどかかったが、2作目からは、所要時間が大幅に減った。）

- ④ 完成（仮）後、会議にて教員に確認していただく。問題点がある場合、③（必要があれば、①に戻り訂正をいれることもあった）に戻る。問題がなければ、完成！！

- (3) 時間割案（試行も含め）をいくつ作成しましたか。また、それらに要した時間はどのくらいでしたか。

大きくわけて4つ。3年生の選択授業と必修授業が入るかどうかが最初の大きな壁であり、どのパターンもこの部分の作業にはかなりの時間を費やした。逆にいえば、このパターンができた分だけ時間割ができるといっても過言ではない。

次に、体育、家庭科、教養基礎など条件がきついものがすべて入りきるかどうか、なんとか入っていったとしても、最後の1個がどうしても入らず、たちゆかなることもあり、最後の1個まで、気は抜けなかった。

しかし、作業を繰り返すうちに、効率のよい作業の順番や、様々な条件が自分の頭の中でも整理されていき、作業も早くなった。細かい違いまでいれると、かなり多くのパターンを作成したことになるが、3年生の選択授業、必修授業のパターンで分けると4

つである。実際に作業をしていた時間は

- 1 作目 12時間程度
- 2 作目 6時間程度
- 3 作目 5時間程度
- 4 作目 5時間程度

(4) ソフトの構成（作業手順）・機能は分かり易かったですか？

分かりずらかったもの、慣れてきたら易くなったもの等々…具体的に記して下さい。

基本的に手順としては手作業で時間割をつくるイメージにとっても近い形で、作業などの面倒な部分はソフトがやってくれるという印象である。ソフトが勝手にいろいろやってくれて、こちらが何をやっているかわからないということがなく、あくまで自分で進められているという感覚が常にあり、それがとてもよかった。

とはいっても、前にもかいたように、あらかじめ入力された条件から、コマが入れるところや、交換可能（途中から導入された機能）のコマを表示してくれる、といった機能は、人間がやろうとすると、それだけでかなりの労力を使う部分であると思うが、そこをソフトがしてくれるのはかなりありがたかった。このソフトを使ってみて、「自動作成ソフト」ではなく、「時間割作成支援ソフト」の「支援」の意義がよくわかった。

1つ1つの操作に関しては、慣れるまでは戸惑ったところもあった。例えば、コマをいれる作業も操作としては3回クリックが必要で、最初は少し戸惑ったが、今となっては必要な3回であることがわかったし、慣れればまったく問題がない（やっているうちに必ず慣れる）。また、ソフトの中での独自の用語や科目の分類があるので、それは慣れるのに少し時間がかかったが、必要最低限のものであると思う。

(5) 作業時間短縮という観点において、ソフトのどんな機能が有効だったですか？

一番に挙げられるのは、画面上での時間割完成が、そのまま完成につながる、ということ（手作業の場合は、完成したものを清書する（打ち込む）必要がある）。今回、一度提示したものが、様々な理由でやり直さなければならなくなるのが数回起きたが、そこまで苦に思うことはなかったのも、このことがかなり大きな理由にある。

(6) 良い時間割を作成するという観点において、ソフトのどんな機能が有効だったですか？

クラス時間割と、教員ごとの時間割などをチェックできる機能：この2つをみながら、偏りがないかなどのバランスはチェックし、さらにより良い時間割になるように調整をする作業が中心になる。

交換機能：コマがすべてはまった後での調整の際、調整が厳しそうに思えたところも、答えはでてなくても交換機能を頼りにとりあえず、コマを動かしていき、新たな可能性を探ってみる、ということを試みた。失敗に終わることも多かったが、思いつかなかった解決策が見つかることもあった。

(7) 作業プロセスに関して、改善案をご提案下さい。

最初の、科目情報入力の部分の書式（Excel ファイル）を整理する必要がある。

- ・情報がかぶっていて、入力の際に無意識にぬかしてしまっており、作成の際に不具合（コマがない）がおきてしまったケースが何度かあった。
- ・教員の数に変更が生じたときに、ずれてしまう危険性があり、それをなくしたい。
- ・これは大変かもしれないが、入力した情報が、ソフトの中でどのように整理されているのかが、少しでもわかっていると、不具合がでたときに少し対応ができるかもしれない。（今回は開発者に頼ってばかりでしたので……）

(8) ソフトの機能に関して、改善案、追加の案をご提案下さい。

作業中に、交換機能やバックアップ機能を搭載していただきました。どちらもとても、役に立ちました。充分だと思います。

5. むすび

結局、今回も昨年同様、ソフトの改良は時間割作成の作業と同時進行になった。年度初めからの長い空白の時間があったが、結果的に見ると、具体的な作業なしで改良に取り組むのは適切ではないと断言できる。頭の中でシミュレート出来ることなど、大したことではないのだ。実際、アイディアはいくつも準備してあったが、プログラミングを始めてみると、多くはあまり役に立たない代物であった。なお、今回の改良作業を通して再び感じたことだが、私自身が「昔のPCの呪縛」からまだまだ解放されていない。今や、メモリーは何の気兼ねもせずにバンバン使ってよいのだ。オブジェクト指向という考え方にかまけて（?）、例えば時間割表（オブジェクトの例）などは、メモリーの中（サイバー空間）にたくさん作ってしまえばよいのだ……とは言っても、なかなかそういった手法が直感的にイメージ出来ない。おそらく、そこを乗り越えれば、ソフト開発の新たな視界が広がってくるのだろう。

時間割を作るソフトと言えば、普通は自動的に作成するソフトを意味するだろう。今回のソフト開発では、そのスタンスは早い時点で諦め、作成支援に徹してきた。しかし、実際にこのソフトを使ってみて、「支援」であることに意義があるとも思えてきた。例えば、交換可能なコマを探し出すことなどは、手作業ではかなりの時間と“神経”を使うが、コンピューターはそのための膨大な量の単純（かつ正確な）作業を1秒もかけずに行ってしまう。一方、「どのコマと交換したら良いか」といった価値判断はコンピューターの不得手な領域である。もちろん、「価値」をあらかじめ定義しておいてコンピューターに判断させることは可能だが、コンピューターは漠然とした定義を受け付けないから、「価値」に厳密な定義を与える必要がある。そこまでやるのは大変なことだし、必ずしも良い結果を出すとは限らないだろう。つまるところ、人間とコンピューターには、それぞれ得手不得手の領域があるので、そこを補い合っていけば良いのではないか。人間が得意とするところは、人間がやれば良いのではないか。むしろ「人間がやるべき」と強く主張しても良いのかも

しれない。時間割作成支援ソフトはそれを実現しているとも言えそうである。

十九浦教諭からのメッセージの中に、クリック一つで Excel シートに“清書”されることの意義が書かれているが、確かにそれがなかったら、時間割の改良がおっくうになるのは事実だ。Excel シートへの“清書”は、プログラミングとしては比較的簡単な部類に入るのだが、それが人間にもたらす効果は大きいわけだ。この例一つとっても、時間割作成にコンピューターを使わない手はないと言えるだろう。一度コンピューターを使ったなら、恐らくもう「手作業」には戻れないと思う。しかし、そこで問題になるのがソフトである。今回開発してきたソフトは来年度の時間割作成にも使う予定だが、5年後、10年後……となるとどうなるか怪しい……基本ソフト（Windows）のバージョンアップに果たして耐えられるか等々。早晩、代替ソフトが必要となるのは必至だが、市販ソフトやフリーソフトは汎用なので、本校仕様にするにはそれなりの手間がかかる。それを考えて、徐々にでも各種時間割ソフトの試用も考えに入れていく必要があるだろう。今回、教務部のご厚意で、時間割作成専用のノートパソコンを2台購入し、有効に活用できた。この2台をソフトの試用にも活用していくと良いのではないかと考えている。

元々は「自分が時間割作成の係になってしまったからソフトを作った」のだが、決して小さくはないソフトを独自に組み立てていく作業はいろいろな意味で有意義だった。このソフトにどれだけ独創性があるのか、あるいは社会的に見て価値があるのかは分からない。しかし、昨年も強調したことだが、とにかくこのソフトで本校の時間割が作れたことでまず満足し、さらに今回は十九浦教諭が「便利だ」と言って使ってくれて、私にも意外だったほど短時間で時間割を作り上げたことにまた満足している。このような、小さいがオリジナルな活動、小さいが確かな満足の積み重ねが教育現場の活性化につながるとも感じている。

しかし今、私はここに本開発の終了を宣言しておく。あと1回だけ、来年度の時間割作成のための改良は行うつもりではある。それ以降も、もし要望があればメンテナンスは続けるつもりだが、昨年から行ってきたような開発の継続は止めることにした。もう、この開発とはお別れをする。私事で恐縮だが、今回の開発で、一度は諦めていた VisualBasic のプログラミングがある程度出来るようになった。これに味を占め今度は、専門の分野（物理）での活用に親しみ、VisualBasic 以外の言語もある程度は操れるようにしようと目論んでいるのである。

*1 お茶の水女子大学附属高等学校 研究紀要第57号（2011）『時間割作成支援ソフトの開発 I』

*2 そうは言うものの、プログラムを複数の人間で作っているわけではないので、プログラムコードに関しては、極端な表現で言えば「自分だけ分かればOK」というスタンスであり、その点ではプロのプログラマーの仕事とは比較にならない気楽さではある。良いこととは言えないが。

*3 バグとは、プログラム中の間違いのことをいう。今回の場合は、単なる文法上のミスではなく構造上の欠陥であり、概して発見しにくいものである。

*4 決して「偉そうに言う」わけではない。プログラムのコードとは例えば次のようなものである：

```
'(6)..... ターゲット koma が (tc_tg,d_0,p_0) で受ける力を検出 .....
```

```
For ch = 1 To Form1.ChargeN
  If Form1.ChargeUse(ch) = True Then
    FieldValue = Form1.B_Field(tc_tg, d_0, p_0, ch).Value
    If FieldValue <> 0 Then
      q_tg = Form1.komaDat(tc_tg, kom_tg).StrengthOfCh(ch)
      If q_tg <> 0 Then
        Force_tg = q_tg * FieldValue
        If Force_tg >= Force_tgMax Then
          Force_tgMax = Force_tg
          If ForceMax <= Force_tgMax Then
            ForceMax = Force_tgMax
          End If
        End If
      End If
    End If
  End If
End If
Next
```

これは「コマの交換可能性」を調べている箇所の一部である。ただし、そのようなことはソフト全体を知っていないと理解できないし、プログラムを作成した本人でも、月日が経つと、一部分だけを見せられても、その働きを思い出すには時間がかかるものなのである。……もちろん、プログラムは分かりやすく越したことはないのだが。

- *5 磁石と磁石の間に働く力について、磁石どうしが空間を飛び越えて力を及ぼし合うと考えずに、片方の磁石がその周りに磁場をつくり、もう一つの磁石はその磁場から力を受けるとする考え方が「場の考え」の基本である。磁場に止まらず物理学の多くの領域で、この「場」という考えが用いられ、実り多い結果を得ている。今回のソフトでは、各コマが時間割表という空間に色々な場をつくり、他のコマがその場から力を受けるという発想でプログラムコードを書いていた。
- *6 「2R」は「2年欄組」の略称。本校では、クラス名として蘭（略号R）、菊（K）、梅（U）が用いられている。2Kは2年菊組、2Uは2年梅組を意味する。
- *7 「komaDat」、「StrengthOfCh」等々、本稿に出てくる“英語もどき”の語句のほとんどは私の造語であり、一応の由来はあるつもりだが、特に深い意味や世間で通用するといった一般性はないので気軽に見て行ってほしい。
- *8 komaDat(3, 1)は構造体とよばれるメモリの一種であり、StrengthOfCh(1)はその構造体のプロパティと呼ばれるものの一例である。
- *9 「禁則①のために、何でそんなにややこしいことをする？」と思われる方もおられるかもしれない。ここでは、最も単純な例として禁則①を取り上げているだけで、他の種々の禁則に対しても同じ手法が適用出来るのである。
- *10 ch = 2101では「村井(tc = 3)」という情報が落ちている。ch = 210301のようにすべきとも考えられる。ただし、時間割表の「村井の行」には「村井のコマ」しか入らないようになっているので（「枠」*1参照）、ch = 2101で十分なのである。
- *11 原稿を書いている時点でVisualBasic2010を立ち上げ、使っているChargeの数を数え始めたのだが、プログラムで自動的に作り出しているChargeもあり、かなり煩雑なので諦めてしまった。ただし、プログラムが自動的にやってくれる作業なら、煩雑であっても別に気にすることはないだろう。