

「社会と情報」、 「情報の科学」における実習アイデア

兵庫県立西宮香風高等学校主幹教諭
松本 吉生

1. 教科「情報」の10年と実習

平成15年度にはじまった教科「情報」は今年で10年目を迎えた。筆者はもともと理科(化学)の教員であり、現職教員免許取得講習会によって情報科の免許を取得し、平成15年度に異動と同時に理科から情報科にかわり、現任校で情報科の授業を中心にやってきた。幸いにも現任校は前期後期制をとっており、2単位の授業は週4時間で4月から9月の半年で完了し、1年で2回の授業サイクルが経験できた。教科「情報」の出発時には様々な試行錯誤があったが、他の多くの学校に比べて早いペースで授業の見直しができた。情報の授業を担当する教員も複数おり、互いにアイデアを出し合いながら教材開発をした。教材開発の中心は「実習」だった。

教科「情報」の授業担当者は、平成15年度の出発時にはもともと理科や数学の教師だった者が免許取得講習会で情報科の免許を取得した者ばかりだった。私を含めてこの免許取得講習会で情報科教員免許を取得した教員は、大学等できちんと情報についての学習をしていない。また、本来持っていた理科や数学、工業といった免許の違いもあり、実習についての考え方に大きく違いがある中で、各学校でそれぞれ独自の試行錯誤が行われてきた。

研究発表会やWebでの情報発信から様々な実習事例を得ることができるが、「デジタル技術」や「コンピュータネットワーク」、「プログラミング」を正面から取り扱う実習の少ないことが残念である。現代において「著作権」や「情報モラル」、「セキュリティ」などの問題が顕在化しているのは、コンピュータの普及により、個人レベルで「デジタル技術」や「コンピュータネットワーク」を扱えるようになったからである。またそれらを実現するのは「プログラム」である。

筆者はもともと理科の教員であったこともあり、教科「情報」の実習でも、教科書に書かれていることを生徒が自分で手を動かし、体験的に確かめることができるような実習を意識してやってきた。そのア

イデアのいくつかを紹介する。

2. Wordマクロによる曲線

Microsoft Officeに実装されているVBAマクロは、主にExcelで活用されている。それはExcelで数値計算をする場合に、表計算の範囲内ではできない処理を行うために適しているからである。しかし、VBAマクロはExcelだけに実装されているものではなく、WordやPowerPointなどでも使うことができる。そこで次のようにShapes.AddShapeメソッドを使い、小さな点を打つことでサインカーブを描くことができる。

```
Sub サインカーブ ()
    xmin = 100
    xmax = 1100
    For i = xmin To xmax
        Zoom = 20
        Location = 100
        x = I
        y = Sin(i / Zoom) * Zoom + Location
        ActiveDocument.Shapes.AddShape 9, x, y, 1, 1
    Next i
End Sub
```

x, yの位置を求める関数を変え、コサインカーブやサイクロイド曲線を描くと次のような結果が得られる。

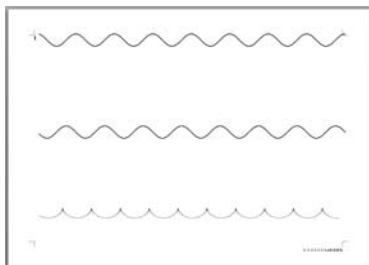


図1 Wordマクロによる曲線描画の例

3. バイナリエディタと文字情報

コンピュータでは文字がコードで取り扱われる。文字をテキストファイルとして保存し、メモ帳などのテキストエディタで開くと文字を読み書きできるが、バイナリエディタはファイルをバイナリデータとして表示するものである。フリーソフトで様々なバイナリエディタが公開されているが、筆者は「BZ エディタ」を使っている。

メモ帳などのテキストエディタを使い「JOHO 情報」の文字を書き、ファイルに保存する。それをバイナリエディタで開くと次のように表示される。



図2 バイナリエディタ BZ による表示

対応する文字コードをIMEパッドなどで調べると、文字と文字コードが対応していることがわかる。また、JOHOのアルファベットは1バイト、漢字は2バイトで表現されていることもわかる。

さらにバイナリエディタ上で0x8B5Aと0x8F70を追加して保存し、メモ帳で開くと、追加した文字コードはそれぞれ「技」と「術」の文字コードであるので、元の文が「JOHO 情報技術」となって表示される。

4. バイナリエディタと画像情報

ビットマップ画像などもバイナリエディタで開くと、コンピュータ上に記録されるバイナリデータを直接見ることができる。まず画像を「ペイント」などで作るが、大きな画像ではデータ量も多いので、3×3ピクセルといった小さい画像を作って保存する。そのままでは小さすぎるが、表示を800%程度に拡大すると作業がしやすくなり、ピクセル単位で色をつけることができる。たとえば左上から右にむけて赤、緑、青、にそれぞれ色を塗って保存する。それをバイナリエディタで開いてみる。

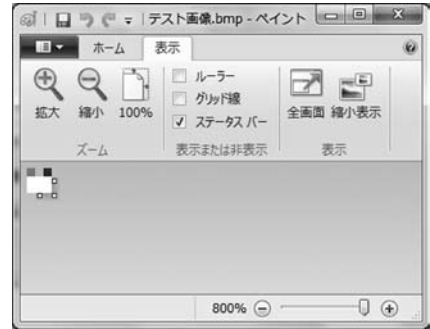


図3 ペイントで作ったサンプル画像

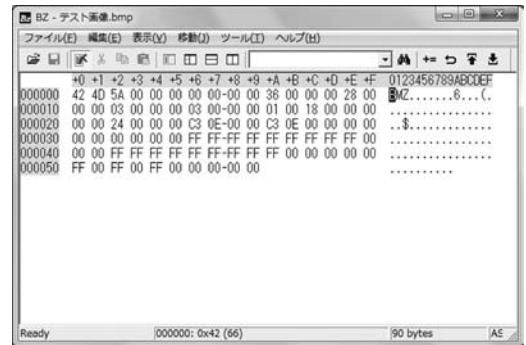


図4 サンプル画像をバイナリエディタで開く

バイナリエディタでビットマップ画像を開くと、データのはじめ54バイトはファイル固有のデータで埋められている。55バイト目から色情報になり、RGBに相当する輝度が3ピクセル分並ぶ。00を3つ挟んで2行目のデータが、さらに00を3つ挟んで3行目のデータが並ぶ。ここで注意したいのは、ペイントの画面上で左上のピクセルがバイナリデータとしては最後にきているところだ。これもバイナリエディタ上でデータを変えて保存し、再びペイントで開くことで色とビット値の対応が確認できる。

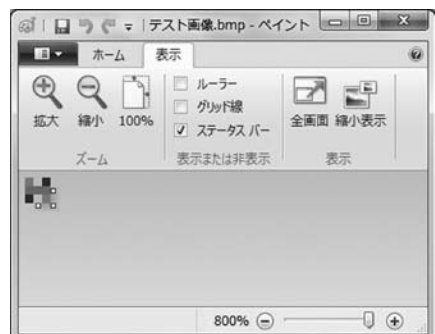


図5 バイナリエディタで色を変えて保存した

5. QRコード

QRコードは日本で開発された2次元コードで、携帯電話などで利用されることも多い。QRコードはフリーソフトで作ることができ、パソコン画面上で読み取ることができるフリーソフトもある。



図6 「easyQR」によるQRコード生成の例



図7 「QR太郎」によるQRコードのデコード

QRコードの特徴に誤り訂正機能がある。印刷されたQRコードに多少の汚れがあっても、ある程度までのエラーは訂正され、正しい情報を表示できるようになっている。そこで作ったQRコードを画像として保存したのち、いくつかの部分を塗り替えてエラーを発生させ、どの程度までのエラーなら正しく表示できるかを試してみるのも興味深い。アルゴリズムの工夫によって、デジタル情報はノイズが入っても元のデータを正しく保持できる例として体験できる。



図8 エラーのあるQRコードを作った例



図9 エラーのあるQRコードを読み取る例

6. 画像の圧縮

画像ファイルにはいくつかの圧縮形式があり、色数に制約があるもの、フルカラーを記録できるもの、可逆圧縮と非可逆圧縮など様々な特徴がある。JPEGはフルカラーを扱える代表的な非可逆圧縮の保存形式だが、圧縮率を変えて保存することもできる。フォトタッチソフトウェアなどを使い、幾何学的な図形やデジタルカメラで撮影した写真、グレースケールで白黒写真に変換したもの、それらをいくつかの異なる解像度で用意し、ファイルサイズを比べ画質を確認させるとよいだろう。できれば生徒自身に画像の形式を変えて保存させて確かめさせたい。JPEGでは圧縮率を上げると「ブロックノイズ」が目立つようになり、エッジ周辺では「モスキートノイズ」が現れる。またGIFでは連続する同じ値を検知し、位置と位置長で置き換える圧縮をするので、白黒の縞模様画像では縦縞よりも横縞のほうが圧縮率の高いことなどがわかる。

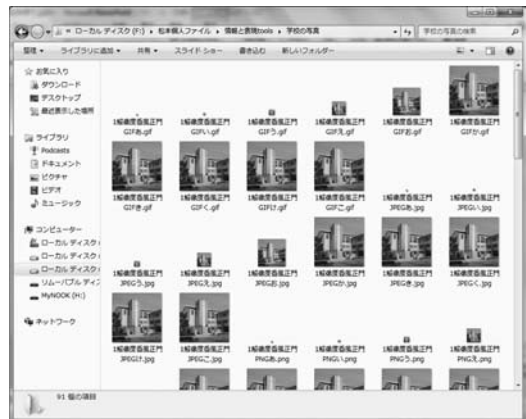


図10 様々な画像ファイル(1)

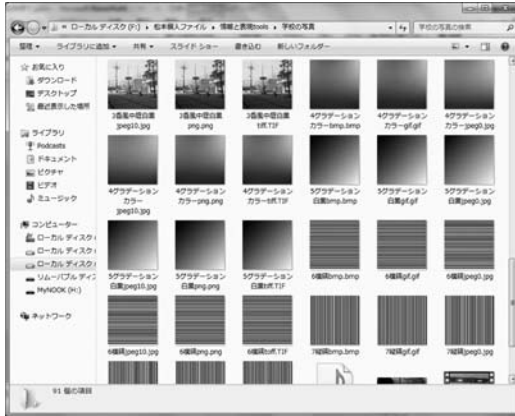


図 11 様々な画像ファイル(2)

7. データベース

データベースは表計算とは異なり、検索・抽出などの再利用をできるようにしたものである。もっともよく使われるものはリレーショナルデータベースであり、SQLを使ってデータ操作ができるようになっている。手軽にデータベースを体験するには、Microsoft Access データベースを使う方法があり、Access そのものを使わなくとも拡張子 .mdb の Access データベースファイルだけを用意し、SQL によって使う方法がある。SQL によってデータベースを操作するフリーソフトは数多くあるが、その多くはデータベース技術者の実務的な利用に適した作りになっており、初期設定や使い方が難しいものが多い。実習で使いやすい SQL クライアントツールとして「SQLite Easy Viewer」があり、これは Access の .mdb ファイルに接続することも簡単にできる。あらかじめ Access で「学籍番号」、「氏名」、「生



図 12 Access ファイルに対する SQL の実行例

年月日」のフィールドを持つ「生徒」テーブルを作って保存しておき、これに対して SQL クライアントツールから SQL 文を与えることでデータ操作ができる。

8. ネットワークとプロトコル

TCP/IP ネットワークでは、データは IP アドレスとポート番号で送受信される。1台のコンピュータで Web ブラウズと電子メールの受信が混信することなく同時に実行できるのは、ポート番号が異なるからである。また、よくセキュリティを確保する方法として、ファイヤーウォールによって「ポートを閉じる」といった表現をするが、本来コンピュータは何らかのプログラムによってポートを「リッスン」していなければ接続を受け付けられない。NET フレームワークはライブラリが充実しており、C# プログラミングにより特定のポート番号でリッスンするサーバーを簡単に作成することができる。同時に異なるポート番号でリッスンするサーバーを動かし、Telnet によって接続すると、ポート番号によるデータの送受信が実習できる。

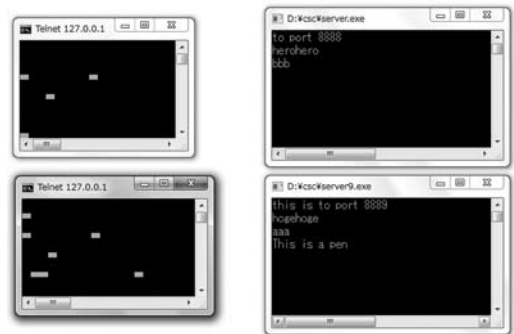


図 13 異なるポート番号への通信の例

9. おわりに

実習のアイデアは、教科書を開いて授業の組み立てを考えながら、他方でソフトウェアの活用などを試行錯誤する中で形になっていく。またアイデアを授業案にする過程でも工夫が必要だ。ひとつの実習を形にするだけでもたいへん手間がかかるが、労を惜まず教材開発をすすめたい。コンピュータとデジタル技術には無限の可能性がある。その面白さを生徒に伝え、知的好奇心をかきたてることが「情報」科教員の責務だと考える。