

情報科教育法 II 2003 # 9

久野 靖*

2003.12.24

はじめに

前回、画像をページに入れる演習をやったとき、HTMLと画像とが別のディレクトリにあって「表示されない」と悩んでいる人がいました。ページに画像を埋め込むってことは、その画像ファイルがHTMLファイルと一緒にないと無意味だと思いませんか？ 計算機は「あなたがこうだと都合がいいなあと思ったように自動的にやってくれる」とは限りません (Windows系のソフトばかり使っているとそのことが分からなくなる傾向にあると思います)。その辺、よく考えてみてください。

さて、今回は画像は取り寄せにしましたが、今回は画像を「プログラムで生成」してみます。これをやると画像がコンピュータ内でどう表現されるか分かるようになると思います。

今回の講義は「情報倫理」および関連する話題を取り上げることにします。このあたりのことは、ある意味「正解」はなくて、教師になる人が各自で「自分の考え」を確立しておくことが必要だと思っています。

1 情報倫理

1.1 情報倫理とは

「倫理」(ethics) → 哲学の1領域。哲学って？

「哲学」(philosophy) → 世界や人間の知恵・原理を探求する学問。真理認識の学問一般を指していたが、そこから個別の科学(sicence)が独立分離したので、現代ではそれを除いた部分を指す。だから哲学は科学の母。米国では博士は基本的にすべて「Ph. D」(Docotor of Philosophy)。日本では「学術博士」とか訳されるけど。

哲学の諸分野:

- 存在論(形而上学) — ものが存在するということはどういうことか
- 認識論(論理学) — 論理のしくみ
- 感性論(美学) — 美しいとはどういうことか
- 実践論(倫理学) — どう行動するべきか

もともとの倫理学 → 「道徳規範のあり方」「善とはなにか」

応用倫理学 → 個別の分野「医者はどう行動するべきか」etc. (職業倫理)。情報処理技術者の…

- 情報処理学会倫理綱領 — <http://www.ipsj.or.jp/04tosho/ipsjcode.html>
- 電子情報通信学会倫理綱領 — <http://www.ieice.or.jp/jpn/about/code.html>

要するに「技術者として反社会的なことはしない」

しかし「情報倫理」といった場合は「技術者の」というわけではない。むしろ高校の教科「情報」で教える以上、だれでもが関係するような場面。たとえば「Webを見る」「メールを送る」「BBSに参加する」などの場面で考慮が必要 → 「情報の受信、取り扱い、発信に関わる倫理」と考えるのがよい。(「情報科教育法」11.1節も参照のこと。)

*筑波大学大学院経営システム科学専攻

1.2 情報倫理のよりどころ

「どのように行動するか」という問題をとりあげる場合、「それは規則だから」で話が終わってしまうと相手(生徒)に納得してもらうのは難しい。また、規則で決まっていなかったことは何をやってもいいことになってしまう。それでは困るでしょう？

だから、「なぜそうなのか」という判断原則(=倫理的問題)について、自分なりの考えを持つこと、それを相手に開示して「説得」することが必要。相手が納得して共感してくれれば、その共通の土台に立って話をすることができる。

例: アシモフの「ロボット3原則」(ロボット倫理?)

1. ロボットは人間に危害を与えてはならない
2. ロボットは上記に反しない限り、人間の言うことに従わねばならない
3. ロボットは上記に反しない限り、自分を守らなければならない

では人間はどのような「原則」に従って行動したらいいの?(=倫理的問題)

- (a) 自分がしたいようにする。
- (b) 他人に迷惑を掛けない。
- (c) 他人に危害が及ばないようにする。
- (d) 他人が喜ぶよう行動する。
- (e) ...

結局「それはいいことだけど、そればかり優先できない」(トレードオフ、利害対立)

たとえば「黄金律(golden law)」というのがある。

- 自分がしたいように、他人にもせよ。

これは「お互い様」とか「互惠」とかいろいろなことが含まれていそう。自分では気に入っているが、他人が他の考え方を採ることは別にその人の自由。宗教とか神とかいう人もいるんでしょうね…

2 模擬授業

今回の模擬授業は「情報C」p50~51「暗号技術と情報の保護」だそうです。担当の方はよろしく。

3 情報倫理(2)

3.1 さまざまな倫理的問題

倫理的問題は教育の場面では多く出て来るし、それに対して何も答えられないと教師をやっていくのはつらいと思う。

人を殺してはなぜいけないの？

- 法律で決まっているから(規則なら何でも守るのか?)
- 人を殺すことは悪だから(「悪って何?」という質問にすり変わるだけ)
- 実は殺してもいい(本当にそんなんでいいの?)
- 神の教えに背くから(宗教な人はそうなのかも…)
- 自分が殺されたら嫌だから(←黄金律)

あなたは何で生きているの？

- 死ぬのは苦しくて嫌そうだから (苦しくない死に方があったら死ぬの?)
- 生きていると楽しいことがあるしだから (なさそうだったら死ぬの?)
- 本能だから (それじゃ質問の答えとしては Give up でしょ)
- ...

学生は (生徒は) なぜ勉強しなくちゃいけないの？

- 勉強しないと立派な人になれません (アナクロ…)
- 勉強するといい仕事につけて幸せになれる (本当か?)
- 勉強すると生きて行く役に立つ (役に立たない科目は勉強しないでいい?)
- ...

まだまだあります…

- なんで校則を守らなくちゃいけないの？
- なんで授業中にケータイ鳴らしちゃいけないの？
- なんで授業が退屈ならおしゃべりしちゃいけないの？
- なんで先生の言うこときかなくちゃいけないの？
- ...

「なんで授業に遅刻しちゃいけないの？」

- 別に終了 5 分前に行って出席提出したっていいじゃん (そう?)
- 朝起きられないんだからしかたない (起きれないのに単位は期待するの?)
- 遅刻すると先生に失礼である (先生を敬うなんてアナクロ?)
- 単位は講義を全部聞くことの代償だから (そうなのか?)

3.2 グループコミュニケーションとコミュニティ

黄金律「自分のしてほしいように、他人にもせよ」は有効だが、実際に何が「望ましい」かは人によって異なる (価値観の相違)。では、相手がしてほしいことが分からない場合にはどうしたらいいのか？

答え (の 1 つ) としては、意志疎通を図ることで相手の価値観と自分の価値観をすり合わせる、また相手に合わせなくてもいいから、とにかく相手の価値観を理解する。

コミュニティ (共同体) → やりとりしている (= 意志疎通している) 人の集まり。地域社会、クラス、研究室、クラブなどさまざまなコミュニティがある。意志疎通が成り立っていれば、互いの価値観がある程度わかっている → 行き違いは少なくなる。

ネットワークの発達 → グループコミュニケーション (掲示板、ネットニュース、メーリングリスト、その他) が発達 → その参加者もやはり「コミュニティ」となる。その場合の問題点:

- ネット上のコミュニティは物理的な接触がないので、互いの意志疎通を図る手段がネット上のやりとりしかない → 意志伝達が難しい、誤解や勘違いが生じやすい
- 地域的に離れた人がコミュニティを形成 → 共通のバックグラウンド (住んでいる場所、受ける授業、活動体験) がない → さらに意志疎通を難しくする。

ではどうするか？

- 実体験の共有 (例: オフミ)
- 誤解を受けないような文章を書く
- 自分の発言には厳しく、相手の発言には寛容に

まあ言うのは簡単だけどやってみると易しくない。たとえばこのクラスの掲示板の内容だって…

3.3 コミュニティと文化・常識

常識とは→「誰もが知っているような事柄」。その「誰もが」というのは、「コミュニティ内の」が暗黙についている (が、普段そのことは意識されていない)。

常識も含めた、考え方や行動のスタイル (の共通部分) → コミュニティの「文化」

あるコミュニティ (ML、掲示板、等) からネットをはじめてそこの文化になじんだ人がよそのコミュニティに行った時にトラブルが起こりやすい

- 他のコミュニティの文化をそのまま持ち込もうとする
- しかもそれが「正しい」「常識」だと思っている
- 自分は正しくて他の人が間違っていると考え

「正しさ」「常識」は相対的なものであることを肝に命じておくべき。たとえば「1+1=2」は正しいか?

コミュニティによって大幅に「常識」が異なっていることがらの例…
あるメッセージに対する「返事」「応答」のことを「レス」と呼ぶ。

- BBS、掲示板では多く見られる
- ネットニュースでは嫌われる「レスって何ですか?」
- メールでは人によりけり (どちらの文化の人によるか)

→ そのコミュニティの「常識」に応じて使うか、使わないで済ます

Web ページすべてを「ホームページ」と呼び、さらに「HP」と略す

- WWW を古くからやっている人 (= プロに多い) では嫌われる (ホームページとは本来入口ページ、トップページの意味)
- 語感がいいのか、世の中には広く流布している
- 教科「情報」では 1 番目の意味で使うこととしている

→ これも使わない方がよい言葉と言える

自分の質問メッセージに対して回答があったら「お礼」のメッセージを出すべき

- これが「常識」になっている掲示板もあるらしい…
- ネットニュースでは単なる「お礼」はよくないとされる (情報が無いから)。お礼を兼ねて、質問と回答のまとめを投稿するのがよいとされる。
- 別に何もしなくてもいい、というコミュニティも多い

3.4 マナー、規則、エチケット (ネチケット)

「何がマナーか」はコミュニティによって大幅に異なっている可能性がある。まあ「黄金律」(人が迷惑に感じることはやらない) とかは共通していると言えるが。

コミュニティによっては、規則が決まっている (例: 学校の校則)。この場合、参加する人は参加時にその規則に同意しているものと考えられる。また、その規則を持ったコミュニティだからこそ参加する、という参加

者がいるはず。なので、規則を破るのは「約束違反」。もちろん、正統な手続きで規則を改良しようとするのはよい(しかし簡単ではない)。

エチケット(ネチケツト)もマナーの一種なのでコミュニティ次第という側面は大きいが、ある程度多くのコミュニティで支持されている共通部分、ということ。参考ページ:

<http://www.cgh.ed.jp/netiquette/> (ネチケツトホームページ)

<http://www.big.or.jp/~roadist/netiquette/> (ネチケツト千夜一夜)

4 画像の表現 — プログラムを用いた実習

普通教科「情報」ではその範囲としてプログラミングを含む部分はない(「情報B」ではアルゴリズムが含まれるので、そこでプログラミング言語を使ってプログラムを動かすことはできるが、原理だけ分かればよいというのが指導要領の趣旨)。しかし、計算機のしくみを納得するにはプログラムを使って色々作るのがよい方法だと自分は思っている。専門教科「情報」があるような高校であれば、プログラミングも前提として授業が設計できるはず。

ここでは一例として、「画像の表現」をプログラムを活用した形で扱ってみる。計算機上では画像(のデジタル表現)は次のような形で表現される。

- 画像は色のついた点(ピクセル)の集まりとして表現される。
- ピクセルは縦横に格子状に並んでいる。縦、横のピクセル数をあわせたものを「画像のサイズ」という。たとえば200x300ピクセルくらいの画像がWebページに入れるのには使いやすい。
- 1つのピクセルごとに、その色を指定する。これは既にやったように、赤/緑/青を0~255の数値(8ビットの数値)で表す。

ではこれを実際にC言語のプログラムで生成してみる。

```
#include <stdio.h>

struct { unsigned char r,g,b; } img[200][200];

main() {
    makeimage();
    printf("P6 200 200 255\n");
    fwrite(img, sizeof(img), 1, stdout);
}

makeimage() {
    int y, x;
    for(y = 0; y < 200; ++y)
        for(x = 0; x < 200; ++x) {
            img[y][x].r = 200;
            img[y][x].g = 150;
            img[y][x].b = 100;
        }
    img[100][100].r = 0; /******/
}
```

このプログラムは、サイズ200x200の画像を生成し、そのすべての点に同じ色(rgb(200,150,100))を入れた後で、1つの点だけの色を変更している。main()の最後で、生成した画像を「PPM形式」という画像形式で

書き出している。PPM形式は、UNIX上の画像表示プログラム `xv` で表示させることができる (また、`xv` を使えば別の画像形式に変更して書き出すこともできる)。

[演習 1]

注意! 今回は演習はすべて **Astec-X** 経由で `cdear` に接続して行うこと。ただしエディタは秀丸でも構わない。

- (1) 「`cp ~kuno/work/makeimage1.c` なんとか.c」で上のプログラムをコピーしてくる。
- (2) 「`gcc` なんとか.c」でコンパイルする。
- (3) 「`./a.out >test.ppm`」で実行し結果をファイルに入れる。
- (4) 「`xv test.ppm`」で画像を表示させてみる。
- (5) 表示されている画像の上でマウス右ボタンクリックしてパネルを出し、「Save」で保存ダイアログを出す。「Format」のメニューで「GIF」を選択して保存する。
- (6) 再び「Save」で、今度は「Format」に「JPEG」を選択し、すると「品質調整パネル」が出るので調整はしなくてもよいからそのまま保存する。
- (7) 「`ls -l test.*`」で3つの画像形式のファイルサイズを調べる。

ファイルサイズを見ると、PPM、JPEG、GIFの順に小さくなっていくはず。PPMは圧縮のない画像形式だが、JPEGとGIFは圧縮機能がある。GIFは単色の画像や少ししか色が使われていない画像を得意とする。

[演習 2]

先のプログラムを修正して、次のように直してみよう。

- (a) x 座標や y 座標の値に応じて連続的に色が変化するようにする。ヒント: ピクセルのRGB値を定数にする代わりに x 、 y の値に基づいて計算した式の値を入れる。
- (b) 縞模様にする。ヒント: x や y の値を定数で割った余りがいくつ以上かどうかによって入れる色を変更する。たとえば「`if(x % 10 > 5) ...`」で枝分かれするなど。
- (c) 乱数を使って一部のピクセルだけ色を変えてみる。ヒント: 「`rand()`」という関数を呼ぶと0~2147483647の範囲の1様乱数が帰ってくる。だからたとえば「`if(rand() < 200000000) ...`」というif文の内側は0.1の確率で実行される。
- (d) 一度上のいずれかの方法で画像を生成した後、その上に部分的に別の形(長方形や円など)を重ね塗りする。ヒント: 上のプログラムの「`/****/`」のところを変更する。たとえばその前の部分と同様にして2重ループで、ただし x や y の範囲を狭くして色をつければ小さい長方形が重ね塗りできる
- (e) 上と同様だが、完全に前の色を塗かえてしまうのではなく、前に入っていた色とを新しく塗る色に反映させる(たとえば平均を取ったり足し算したりする)。

いずれも、表示する方法は先の演習と同じ。また、GIFやJPEGファイルで保存してみて、ファイルサイズも比べてみるとよい。色が多数あつたりするとGIFよりJPEGの方が小さくなる。また、JPEGで品質を落とすとさらに小さくできるが、画像はキタナクなる。

[演習 3]

上の演習で作ったGIFファイルまたはJPEGファイルを自分の適当なWebページ(前回までに演習で作ったものなど)の背景画像にしてみよ。たとえばページ全体の背景にするならCSSで次のように指定。

```
body { background-image: url(test.gif) }
```

また、特定要素の背景にするのも同様。

```
h1 { background-image: url(test.gif) }
```