



| | |
|--------------|---|
| Title | 船舶海洋工学の“情報活用基礎” |
| Author(s) | 鈴木, 博善; 箕浦, 宗彦 |
| Citation | サイバーメディア・フォーラム. 2006, 7, p. 42-43 |
| Version Type | VoR |
| URL | https://doi.org/10.18910/70229 |
| rights | |
| Note | |

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

船舶海洋工学の“情報活用基礎”

鈴木 博善・箕浦 宗彦（工学部 地球総合工学科）

a)はじめに

工学部地球総合工学科は、3つの学科目で構成されている。現在のところ、それぞれの学科目が要求する“情報活用”の“基礎”の目指すところが異なるので、“情報活用基礎D”は、“D-I”、“D-II”、“D-III”に分かれている。私たちは、そのD-Iの担当をしている。この講義は、私たちの他に、助手2名、TA2名を加えた6人体制で行っているが、今回はその代表として、2名で情報教育についての思いなどを書かせていただこうと思う。

b) “情報活用基礎”の中の“情報処理”

地球総合工学科船舶海洋工学科目 / 地球総合工学専攻船舶海洋工学部門での教育・研究の柱となるものに構造力学・流体力学がある。実際、研究あるいは教育の場面で、構造解析、流体解析を実施することが多い。

昨今、これらの解析を、いわゆるアプリケーションソフト（汎用コード）を用いて実施することが次第に多くなってきている。さらには、この構造解析、流体解析の前後にも、アプリケーション（プリ・ポスト）を用いることが一般的であるから、汎用ソフトで計算すれば、可視化アプリケーションを使って、たちどころに綺麗な絵が描ける。

ここで、船舶海洋工学を教える大学での教育として、“情報を活用する”とは何であるかを考える必要がある。これには、3つの立場があるのではないかと思う。

1. 数値計算の手法そのものの学習、アプリケーションの中身がどうなっているのかを考えるような訓練をおこない、ものを考える能力を養う。
2. アプリケーションの使い方を教え、アプリケーションをとにかく使えるようにする。
3. アプリケーションの出した結果から、真理(物理)を見る能力を養う。

意見のあることは重々承知であるが、私たちは、あくまで、1を目指している。

2は、大阪大学に入るような学生であれば、すぐにアプリケーションソフトを扱えるようになるだろうし、そもそも、学生の時にアプリケーションを動かしたことがあるなしが、一生、少なくとも働き手として生きて行く期間、致命的な問題となるとは思わない。3については、これは、他の学問体系をきちんと学んでから実施することであろうから、2年生配当の科目にはなじまないし、情報活用基礎の時間程度でできることでもない。

よって、私たちは、C言語でコーディングをするのである。“Hello World”ぐらいを書く基本的なプログラムに始まり、ある程度プログラムがかけられるようになってから、具体的な問題をコーディングする。（今回、初めてANJUTA上でのプログラム開発を行ったが、2バイトの文字を受け付けないのは良かった。geditでは、中括弧の全角半角がわかりにくい）

このようにして、どのような理論でアプリケーションの中身ができているのかを考える機会（きっかけ）になれば・・・と言う思いで講義を進めているのである。

もちろん、最初にネチケット、サイバー犯罪、WEB情報の不安定さを教えた上での情報処理教育であるが。

c) 授業の内容

授業の内容について、簡単に紹介する。「ものを考える力を養う」ことを目的に、大きく次の5つの課題で授業を進めている。

- 1) 文字のランダム表示
- 2) シンプソン積分
- 3) ヤコビ法とガウス - ザイデル法による連立1次方程式の球根
- 4) 二分法とニュートン法による高次方程式の球根
- 5) 円柱周りのポテンシャル流れの可視化

1) は、C言語に慣れるための課題である。ここで、基本的なコーディング技術（および概念）である入

力、表示、繰り返し、条件分岐、配列、関数を学習する。具体的にはランダムな位置に文字を表示させるだけであるが、連続して表示すれば、文字が画面の下から上に流れていくように見える。いくつかの文字をまとめて自動車の形にすれば、自動車レースのように見える。これにより、コーディングの面白さを体感する。

2) ~ 4) は、反復計算をどのように行うか、条件分岐をどのように実現するかに、頭をひねることになる。それぞれの課題で解析解と数値解を比較し、コードの出来を、学生自らが確認する。

5) は、本授業のクライマックスである。一様流中の円柱まわりの複素ポテンシャルを与え、流線や速度場を求めて、それを画面上に表示する。流体力学の教科書ではおなじみの図であるが、きれいな流線が描けると、「できた！」という達成感に満ちた声が返ってくる。

それぞれの課題で、特に複雑で難しいことはしていない。C言語の文法も詳しくは教えない。最初に

サンプルコードを与え、目的に応じて、そのコードを学生自らが試行錯誤し、改良していくスタイルである。サンプルコードは、課題を重ねるたびに簡素になっていき、その分、学生が改良しなければならないところが増えてくる。最後にはほとんどの学生が、ちゃんと流線を描けるようになる。

d) さいごに

我々が「もの考える」ときに心がけなければならないのは、「論理的である」ことである。例えば、この文章の冒頭で述べている「C言語でコーディングする」理由は、論理的である(と思う)。すなわち、物事を順序だてて、筋道をつけて、条件を仮定し、ある原因から、特定の結果を導き出さなければならない。実は、これはコンピュータ言語によるコーディングの作業そのものである。積分を足し算に変換し、それを繰り返し命令でコーディングする作業は、論理的思考をしないと実現するのは難しい。この授業を通じて、学生が論理的思考を身につけ、「ものを考える力」を養ってくれることを期待する。