◇◇◇◇◇
解説
◇◇◇◇◇

情報教育のための講義管理システムの構築とその教意義

三浦　元喜[1]

1　はじめに

九州工業大学工学部では，2004 年 (平成 16 年) 度の改組から，1 年次と 2 年次の全学生を対象に，4 つの情報系必修科目 (情報リテラシー，情報 PBL，情報処理基礎，情報処理応用) を開講している．1 年次前期の情報リテラシーでは，情報科学センターの教育用計算機および学内ネットワークの利用方法を学び，以降の情報系科目履修の基礎となるコンピュータ活用能力を身につけることを目的としている．1 年次後期の情報 PBL では，表計算や数式処理といったアプリケーションの活用法を学んだあと，少人数 (3〜6 名) のチームごとにテーマの調査，作品の製作，プレゼンテーションを行っている．2 年次前期の情報処理基礎では，C 言語等のプログラミング言語を題材に，構造化プログラミングとアルゴリズムの基本を学ぶ．2 年次後期の情報処理応用では，PIC マイコン搭載のロボットカー制御，アセンブリ言語，Fortran を用いた数値計算法のうち，学科が選択した内容を学習している．

これらの情報系科目については，すべて情報科学センターの計算機室で実施している．そのため，学生はひとり 1 台の PC 端末を利用し，演習や実習を中心として実践的な学習を行うことができる．しかし，講義を円滑にすすめたり，きめ細やかで効果的な学習機会を提供するには，情報基盤に加えて，Moodle をはじめとする LMS(学習管理システム) が不可欠である．

本稿では，筆者が担当する情報系科目で利用するために開発・改良してきた講義管理システム (出席システム) を紹介し，最後にどのような点を考慮しながら授業を行っているかについて述べる．

2　出席システムによる個人学習・教育支援

出席システムは，当初学生の出席記録を管理することを目的として筆者が 2009 年に開発・運用を開始した Web システムである．しかし授業実施にあたって必要となった学生の小テストやレポート，演習，試験や成績の管理，アンケートや相互評価の機能が段階的に追加され，現在では統合的な講義管理システムとして運用している．

2.1　出席記録

九州工業大学工学部学修細則の第 11 条第 2 項には,「科目の履修には，原則としてその総授業時間数の 3 分の 2 以上出席しなければならない.」と定められている．2010 年 4 月から，工学部では学生証をかざす IC カード方式の出欠管理システムを導入・運用しているが，講義室の出入り口に設置してあるためリーダに学生証をかざしてすぐに退室してしまうといった例も聞かれる．そこで筆者が担当する講義では，IC カードによる出席に加え，出席プログラムを起動して出席報告することにしている．

[1]工学府基礎科学研究系　准教授　miuramo@mns.kyutech.ac.jp

毎回の講義の最初に，受講生は計算機の電源を入れて Linux にログインしたのち，出席プログラムを実行する．出席プログラムはコンソールから “shusseki” と入力することで起動する (**図 1**)．出席プログラムは Java Web Start 経由で起動し，学生が使用する計算機の IP アドレスと学生のアカウント名，および出席時刻を教員が管理する出席管理サーバに送信する．計算機の IP アドレスは基本的に固定で割り振られているため，座席も確認することができる．

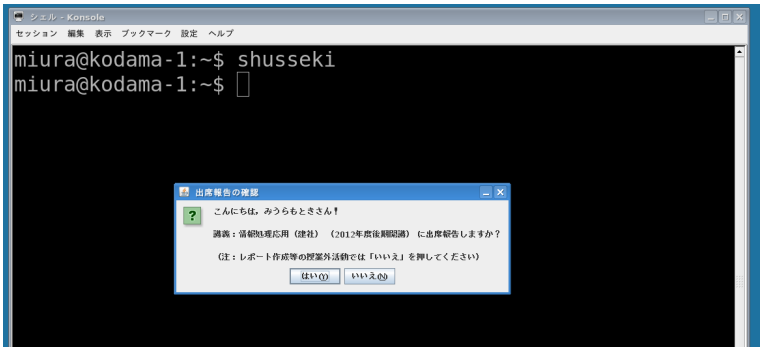

図 1: 出席コマンドとダイアログ

出席プログラムは出席報告のほか，教員が授業に関連して提示したい Web ページの URL を配布したり，教員画面を受講生の PC 画面に表示させるといった目的でも使用するため，受講中は終了せず，継続して動作しておくように伝えている．URL の配布は教員用 PC で開いている Web ブラウザ (FireFox) の URL 左横のアイコンを，出席プログラムのウィンドウにドロップするだけの簡単な操作で行える．URL を配布すると，学生の画面には「リンクを開きますか？」というダイアログが表示され,「はい」を選択するとブラウザが直接 URL を開く仕組みである．こうした仕組みはあらかじめ講義資料に準備していなかった特定のページにすばやく誘導・注目させたいときに効果的である．教員画面の配布は，出席プログラムのメニューから画面送信を選択すると，画面をキャプチャした画像を**図 2** のように受講生の画面に表示する．この機能を用いるとプロジェクタ未使用時[2]でも教員画面を提示できるため，授業時間後に残って演習をさせたい場面や試験中のヒント表示などに活用している．こうした URL や教員画面の配布機能については，事前に作成しておいた講義資料以外の資料をすぐに出せるため，授業中に臨機応変に対応できるという点で有益である．

2.2 演習

毎回の授業の最後には，課題を解く演習をおこなっている．受講生がプログラムを作成してソースコードを提出すると，まずコンパイルチェックを自動で行い，全角スペースの不適切な利用や include 忘れを指摘する．その後，教員のほうで動作をチェックし，条件を満たさなかったプログラムを提出した学生には締切前にメールで通知するようにしている．本来は学生が提出した時点で細かな動作チェックやフィードバックできるようにするのが理想なので，今後改良していきたいと考えている．

2.3 成績の提示

演習の結果は蓄積され，受講生は**図 3** のように現状の点数とクラス内での得点分布グラフを見ることができる．また,「このままのペースだと (最終の段階で) ○○点になりますよ」という表示を見せること

[2]節電および部屋の温度上昇を抑えるため，演習時にはあえてプロジェクタの電源を切ることも多い

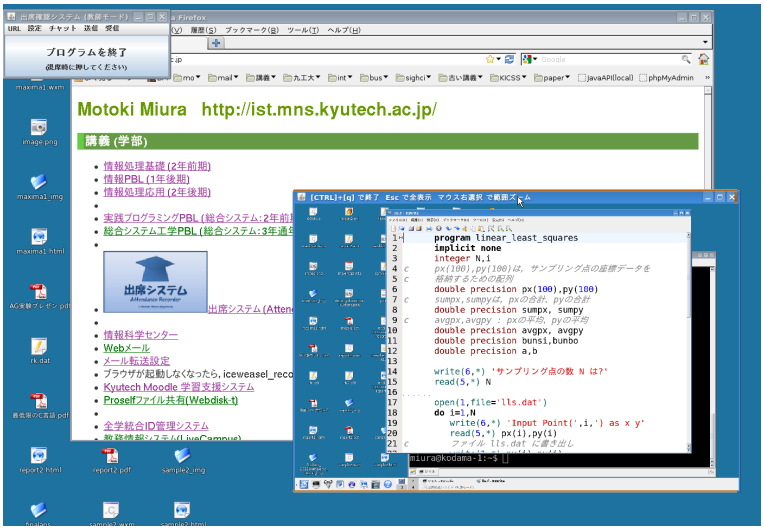

図 2: 教員画面の表示

によって，継続的な努力を促すようにしている．ドロップダウンリストとしているのは，学生の点数というセンシティブな情報を常時画面に表示させないようにするためである．

2.4 画面レポート

情報処理応用では，ソースコードを編集したうえで，gnuplot でグラフを表示させて確認したり，考えを促すような問題を課す場合がある．そういった課題では，ソースコードを貼り付けるだけでは実際に受講生に体験させたい活動の一部しか把握することができない．そのため，受講生が送信したい画面をキャプチャしてレポートとして送信することができるようにした．gimp 等を用いれば画面をキャプチャしてファイルに保存することができるが，学生間でのファイルの流用を防ぎ，教員が体験させたい活動を確実に行ってもらうため，学生用 PC で動作する出席プログラムに画面をキャプチャして出席管理システムに送信する機能を組み込んでいる．教員側では，提出された画像を確認しやすくするため，5 秒毎に次々に表示していく確認用インタフェースを作成している．

3 出席システムによるグループ学習・教育支援

前章では，主に個人を対象にしたプログラミング演習中心の授業に対するシステム機能について紹介してきた．本章では，グループによるプロジェクト学習に関連した機能について，情報 PBL の実施状況を交じえながら説明していきたい．

3.1 Processing.js

筆者が担当する 2012 年度の情報 PBL では，Processing.js を用いた Web アプリケーションの開発を含むテーマを設定するよう受講生に求めた．Processing.js とは，インタラクティブアートやビジュアライゼーションに適したプログラミング言語 Processing のプログラムを，Web ブラウザ上で動作させることができる環境である．通常 Processing のプログラムを実行するには，Processing の統合開発環境が必要

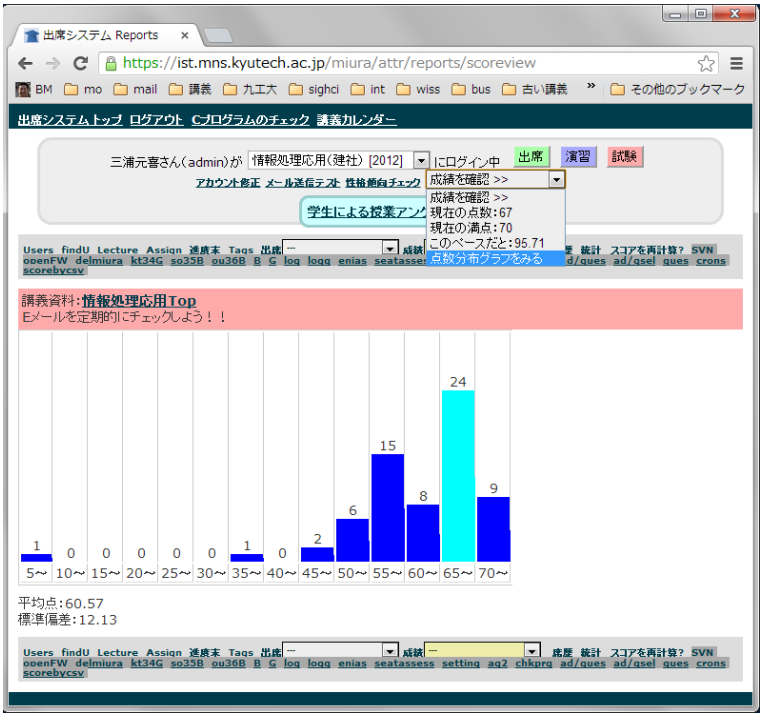

図 3: 得点分布グラフ

になるが，Processing.js は Web ブラウザ上で動作するため，インストールや設定が不要であるという利点がある．

筆者は，Processing.js の特性を活かすため，Web ブラウザだけでプログラムのソースコードを修正し，すぐに実行確認できる環境を構築した．単にソースコードを入力・編集するだけなら通常のフォームでも実現できるが，ある程度大規模なプログラムを編集しようとするとエディタのユーザビリティが重要になる．そこで CodeMirror[3] という Javascript で書かれたコードエディタを組み込んだ．CodeMirror を用いることによって，**図 4** に示すように，シンタックスハイライトや自動インデント補正，ショートカットキー，行番号表示などの機能を備えた高度なエディタを提供することができる．これにより，受講生が Web ブラウザさえあればどこからでも気軽にプログラミングを作成し，実行確認し，保存できるようになった．元々Processing が初心者でも比較的簡単にアニメーションやゲームを作成できる環境ということもあり，意欲さえあればどんどんネット上の情報を調べて動作を試していけるという点で意義深い．

3.2 作品とプレゼンを Web 参照可能にする

情報 PBL では，受講生は作品とプレゼン資料の 2 つを作成し，発表するのが通例となっている．これまでは OpenOffice Impress 等のプレゼンテーションツールを使って作成させていたが，2012 年度ではプレゼン資料も Web から参照できるよう，エディタをつかって HTML で記述してもらうようにした．ただ，全くゼロの状態だとプレゼンテーションがしにくいと考え，Reveal.js[4] や Google HTML5 Slide Template[5] といった雛型を準備した．

[3] http://codemirror.net/
[4] http://lab.hakim.se/reveal-js/
[5] http://code.google.com/p/html5slides/

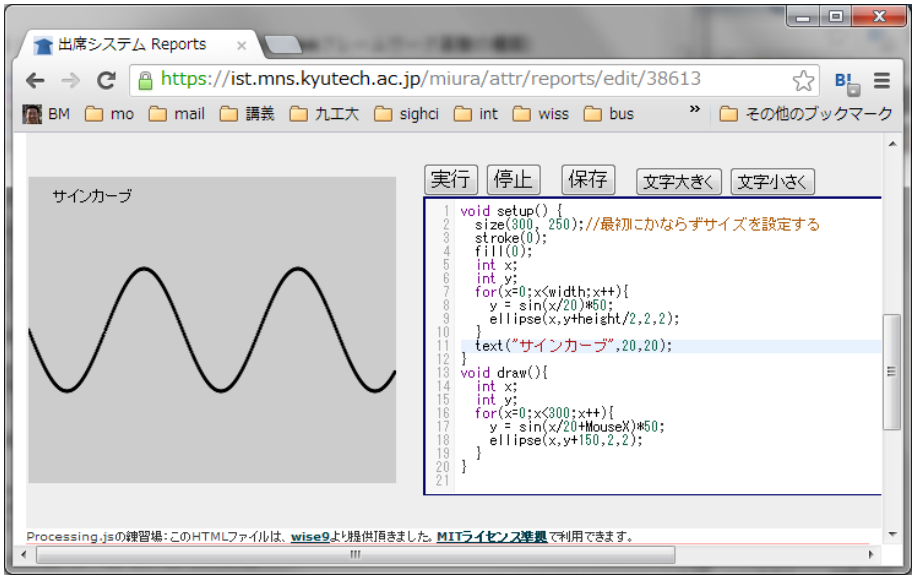

図 4: Processing.js のコードを Web ブラウザ上で編集・実行

http://ist.mns.kyutech.ac.jp/jpbl2012/sample/ に，2012 年度の受講生 (機械知能工学科とマテリアル工学科の 1 年生) が作成した作品およびプレゼン資料の一覧を示す．学生の興味が強いゲームをテーマに挙げたものが多いが，アハ体験を提供するアニメーションや脳トレーニングなど，内容に工夫がみられる魅力的な作品もいくつか見受けられた．今回の情報 PBL は情報リテラシーから情報処理基礎へのつなぎとして，なにかしらのプログラミングをテーマに含めるという制約下で実施した．グループで活動するため一部の学生しかプログラミングを担当していないといった実態はあるものの，全体としてプログラミングや Web 作品製作を通じて，単に情報を受け取るだけでなく，情報や価値を創造し発信することに対する考え方につながってくれることを期待している．

作品やプレゼン資料を Web ページとして作成するうえでは，多数のファイルを扱う必要がある．また，グループでファイルを管理する必要があるため，Subversion でサンプルファイルを配布し，作品とプレゼン資料の部分を修正してもらう方式をとった．教員は Trac[6]のリポジトリブラウザをつかって，どのチームがどのような活動をしているかを比較的簡単に把握することができた．

3.3 メーリングリスト作成

情報 PBL では，授業時間外におけるグループでの活動を推奨している．メンバーが全員集まれなかった場合でも連絡をとれるようにするため，qwikWeb[7]を利用したメーリングリストおよびグループ Wiki 機能を提供している．いくつかのグループは情報共有やファイルの交換等で使用していたが，基本的には授業時間中に調整をしているため，あまり活用はされていないのが現状である．

3.4 相互評価機能

PBL 講義においては，受講生 1 人 1 人が，他の受講生やグループに対する評価 (相互評価) を導入することで，良い発表をしたり，完成度の高い作品をつくるうえでのモチベーションを高める効果を期待

[6] http://trac.edgewall.org/
[7] http://qwik.jp/

している．相互評価活動を効果的かつスムーズに行うため，出席システムに相互評価機能を追加した．**図 5** は，受講者が各グループ，各発表者に対して点数付けしたり，コメントを記入するためのフォーム画面である．受講者はリストボックスから点数を選択したり，フォームにコメントを入力して Enter を押すと，Ajax 技術をもちいて画面遷移なしで即座にサーバに送信するようになっている．採点基準はあらかじめ，(1) 個人向けかグループ向けか (2) 項目説明 (3) 点数の範囲と初期点数，またはコメント入力ならフォームの文字数，といった各項目を設定しておく．それに加えて学生点と教員点の満点を設定する．具体的には，**図 5** の評価フォームは以下の枠で囲った設定で定義されている．5 点満点の項目 3 つ分の 15 点満点の点数から，学生の平均と教員の平均を計算したあと，個人やグループに配点する際にはそれぞれ満点を 20 点，10 点として正規化している．これにより，評価項目を追加したり，学生：教員の比を変更したりといった修正が容易に行えるようになっている．なお，個人やグループに宛てて記入されたコメントや入力されたスコアは，各受講生が即座に閲覧できるため，発表会の直後にコメントを読んで振り返り作業をさせることも可能である．この相互評価機能は情報 PBL だけでなく，総合システム工学部 2 年次対象の実践プログラミング PBL でも運用しており，発表後には大量の建設的な相互コメントと，非常に妥当な評価結果が得られることを確認している．

```
// で始まる行はコメントです．ただし，行の先頭が//である必要があります．
// | で始まる行は共通プロパティです．
|target|individual
// 学生相互評価の配分点（項目の合計点は 15 ですが，20 に正規化する）
|maxstudent|20
// 教員評価の配分点（項目の合計点は 15 ですが，10 に正規化する）
|maxteacher|10
>u
:530: 明瞭性・自信・態度 (5~0) : 大きな声ではっきりと話していれば 5，視線がずっと下だったり，カンペを読んでいたら 1~0
:txt27: コメント : 個人へのコメントを入力して Enter :
>g
:530: スライドとプレゼンのわかりやすさ (5~0) : スライドがみやすくて，プレゼンがわかりやすく工夫されていたら 5
:530: グループ内の協調／チームワーク (5~0) : メンバーが互いに協力し，高め合い，真摯に取り組んだ様子が伺えれば 5
:txt120: グループへのコメント : 発表・活動を通したグループへのコメントを入力して Enter :
```

4 情報教育を行ううえで重視している点

これまでに述べてきた出席システムの機能を構築するにあたって，筆者がこれまでに考えて，重視してきた点を以下に挙げる．

- 素早いフィードバック：学習に対する意識やモチベーションを維持してもらうため，演習，レポートの結果やスコアといった学習状況をなるべく早く公表し，透明性を高める．また相互評価コメントなどは，すぐに閲覧できるようにする．
- 操作性，視認性の向上：授業支援システム自体が複雑すぎてしまったり，使いにくかったりすると，受講生の学習の妨げになりかねない．受講生には**図 3** で示すように「出席」「演習」「試験」の 3 つのボタンを提示し，やるべきことと操作のミスマッチを防ぐようにしている．相互評価のインタフェースについても，画面遷移を防ぐことで入力や変更を容易にしている．

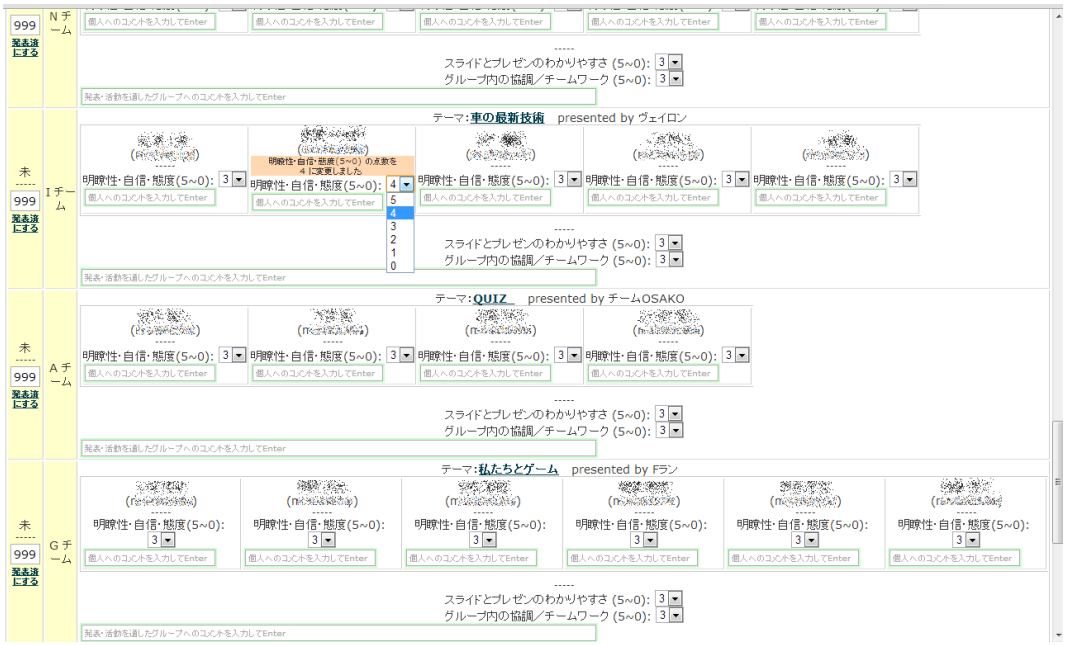

図 5: プレゼンテーションの相互評価画面

- いつでもどこでもアクセスできる：Web システムとして運用しているので，Web ブラウザがあればログインして確認できる．プログラミング環境も Processing.js のように Web ブラウザのみで完結できるようになってきているので，将来的には演習活動のほとんどをブラウザのみで完了できるようにしたい．

これらは，情報技術を活用した教育環境や学習支援システムという点でみれば一般的な観点であり，特別新しいものではない．しかしながら，情報教育を行う教員自身が情報技術を用いた新しい学習環境や支援方法を積極的に取り入れ改善することは，教育の質向上に直接結び付く活動であると筆者は信じている．さらにそのような姿勢を学生に見せることは，情報技術を用いることで利用者にもたらされる教育・社会サービスの具体的な改善や，洗練されたシステムや良く考えられたデザイン等の身近な一事例として想起されることとなるだろう．それによって社会全体がより良い方向に進むための，情報技術の役割や将来の関係性といったことを考え，感じてもらえる効果が期待できると考える．

5 おわりに

本稿では筆者が担当する情報系授業科目の効果的な実施を支える学習支援システムの機能について紹介してきた．情報系科目で直接教える内容は情報倫理であったりプログラミング技法であったり，それを支える論理的思考力である．しかし筆者は，それに加えてもっと高次の姿勢，すなわち現代社会における情報技術のさらなる活用とサービスの深化を模索できる人材を育成したいという観点から，教育というサービス体験者である受講学生にできるだけ優れたサービスを提供することが責務だと感じている．

計算機を用いて講義を行うことによるメリットや教育的効果は，情報系科目に限らず大きいはずである．しかしながら，現状では教室設備の都合上，計算機資源を活用した講義の数はどうしても限られてしまっている．筆者をはじめとする情報セクションの教員はほぼ全ての講義で計算機とネットワーク資源を利用することができるが，これは非常に恵まれた，特別な状況であるといってよい．もし統一的か

つ安定した環境が整備されていなかったら，本稿で説明したようなインタラクティブ性の高い授業や改善活動は実施できなかった．

近年では情報科学センターが無線 LAN 環境を多くの教室に整備している．またスマートフォンを持つ学生の割合も 50%を越え，それに比してソーシャルネットワークサービスを利用する学生の数も増えている．こうした背景から，以前の状況よりは ICT 活用講義に対する障壁は低い．学生，教員，センター等の密な連携により，斬新，魅力的かつ効果的な ICT 学習環境を皆で工夫し，つくりあげていく時代がきていると感じている．

謝辞

本稿で述べた内容のうち，特に情報 PBL におけるプログラミング体験の先行導入や Processing の活用は，工学部の情報教育担当教員を中心に年 2 回実施している「情報教育フォーラム」での議論から生まれたものです．フォーラムでの議論に参加された方全員に感謝の意を表します．