

3S-01 マルチプラットフォーム教育環境における JavaScript を使ったプログラム入門教育

福田 民生 飯倉 道雄 吉岡 亨 樺澤 康夫
日本工業大学

1・背景

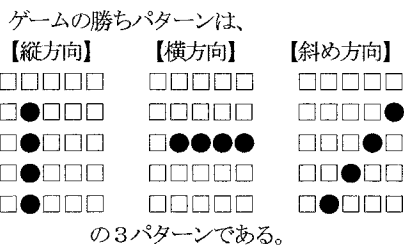
マルチプラットフォーム型情報教育支援環境の構築を行い、その経緯、システムの構成と機能、運用事例の紹介及び、この環境で学習した学生に対して行ったアンケート結果の分析、等を実施している。現在、学生は141台のクライアントで Linux, WindowsNT および Solaris の中から自分の好きなOSを選んで使用することができる。授業では WindowsNT を使った「コンピュータ・リテラシー教育」Linux を使った「情報技術演習」等に利用している。この環境下で実施した『JavaScript を使ったプログラム入門教育』の内容を紹介する。

2・演習の目的

インターネットを利用した情報検索技術が中、高校生にまで浸透した現在、次に要求される技術は“インターネットへの情報提供技術”であろうと考えた。そこでホームページ(以下HP)作成技術の中でダイナミックなHPが容易に作成できる JavaScript を使った『プログラム入門』教育を試みた。具体的には、Hyper Text Markup Language (以下HTML)でHPの基礎学習を行った後に JavaScript を使い“4コマ並べゲーム”プログラムを作成しプログラム入門教育を行った。授業が詰込み形式になり HTML や JavaScript を学生が嫌いになることのないように留意した。

3・“4コマ並べゲーム”の概要

ゲームは、縦5個×横5個の枠目において人間とコンピュータが交互に指し 縦、横、斜め、どの方向にでも先に4個並べた方が勝ちを得るといった単純なものである。



4・授業の進め方

授業は、エディターでプログラムを作成し、ブラウザで表示(確認)しながら進めた。その際、学生が使用するエディターやブラウザは特に指定しないで自分たちの使い慣れたもので実施させた。

(図1 授業の進め方イメージ 参照)

授業の最初の3回でHTMLを使い“会社案内”、“製品一覧表”、“価格表”を作り各々にリンクを張りHTMLの基礎練習を行った。次にJavaScriptを使い簡単だが動きのある例題を体験させてHTMLのみのHPとの違いを強調した。授業は、プログラムの一部(サブルーチン単位)を教材として示し、その通りにコーディングさせ結果を表示(確認)させながら完成させる形で実施した。その際、次の2点を考慮してプログラム内で使用されるオブジェクト(コマンド)を解りやすく体験できる例題を先に説明&体験させた後に、教材のコーディングを行わせた。

- (1)学生に JavaScript の幅広い知識を身につけさせる。
- (2)JavaScript に対する興味を持たせる。

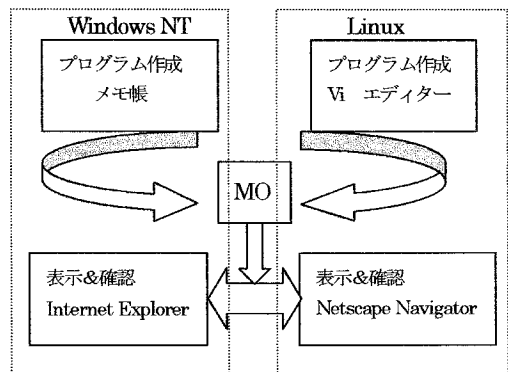


図1 授業の進め方イメージ

Exercise of JavaScript Programming on a
Multi Platform Computer Education Environment
Tamio Fukuda Michio Iikura Tshru Yoshioka
Yasuo Kabasawa Nippon Institute of Technology

JavaScript の場合、バグがいろいろな現象となって出現する為、解決までに予想外の時間を費やすケースが多く、何件かは Teaching Assistant (以下 TA) が掛かり切りで解決した。それらの指導を行った TA の経験を基に学生の間違いやすいポイントを以下に整理した。

- ① 全角, 半角文字の取り違え。 例 ((<< { {
- ② 大文字, 小文字の取り違え。 例 P p O o
HTML では区別しなくても良いと教えたことが悪影響を与えてしまったと思われる。
- ③ 対で使用する記号の閉じ忘れ。
例 () “ ” ‘ ’ { }
- ④ 同じく対で使用すべき閉じタグのミス。
例 </title> </td> </tr> 等
- ⑤ スペルミス
例 value → valeu, text → test, var → vra

最後に、プログラム作成における検査の重要性を認識させる目的で検査成績書作成を体験させた。検査仕様書作成は未だ無理と考えこちらで事前に準備した仕様書を学生に渡して仕様書の指示どりにプログラムが動作するかチェックさせ検査仕様書に記入させた。検査は他人の作ったプログラムに対して行わせ『検査はプログラム作成者以外が客観的に実施すること』『検査仕様書はプログラム作成と同時に準備しておくこと』の大切さを指導した。

5・プログラム構成と使用した例題

プログラム構成と機能及び授業で使おうとしている JavaScript 説明用に準備した例題と“ポイント”を、(図2)にまとめた。

6・今後の課題

(1) <SCRIPT language="JavaScript">

JavaScript 宣言文の下にサブルーチンを追加しながら全体を完成させる今回の進め方の場合、追加した部分にバグがあると <SCRIPT 全体が無視され、今まで正常に動作していた機能も動かない現象が起きた。その結果、何回も JavaScript 宣言文全体を見直さざるを得なくなり予想外の時間を費やす結果になってしまった。

先に整理した学生の間違いやすいポイントを事前に徹底するなど対策が必要である。

(2) 簡単な構文検査ソフトの開発

TA の協力を得たが、限られた時間内で学生のバグを見つける事は困難な時があり授業スケジュールを乱す原因となった。簡単な構文検査ソフトの開発を検討していきたい。

(3) 全体的に、学生には「私は間違っていない！」との思い込みが根強くあり、バグ発見に手間取る傾向が多く見られた。TA や隣の学生の協力を得てバグを探す場合でも、全く他人に頼ってしまっている傾向が多かった。このような学生のモラル改善も今後の課題として対応策を検討したい。

NO	サブルーチン名	機能	説明用例題と“ポイント”
1	JavaScript 基本練習	JavaScript 独特の、動きがありダイナミックな基本機能	<ul style="list-style-type: none"> ・onClick ハンドラー(部屋のランプ オン, オフ “ ” (朝昼晩のあいさつ) ・メッセージを表示して氏名、住所を入力させる。 ・コメント(2種類)の重要性
2	メインルーチン	“樹目”とルールを表示	tr, td タグを使った“樹目”表示
3	対戦開始ルーチン	“先手”か“後手”かを選択させる	<ul style="list-style-type: none"> ・alert メソッド ・confirm メソッド(背景色の変換)
4	“ますめ”初期化ルーチン	次の対戦のために“ますめ”を初期化する	eval メソッド(数字のn進数変換)
5	人間が指すルーチン	人間が指した個所を記録する	<ul style="list-style-type: none"> ・form タグ ・テーブルと配列の定義と使い方
6	コンピュータが指す	下の手順で指す個所を探す ①コンピュータの4並び個所は無い ②人間の “ ” ③受無しの3並び個所は無い ④ランダムに指す	<ul style="list-style-type: none"> ・乱数を使う(小数点以下の処理) ・コンピュータ的考え方(ブルカ爆撃) ・for 文の繰返し(ピラミッド作成)
7	勝敗判定ルーチン	勝敗を判定しメッセージ出力	<ul style="list-style-type: none"> ・コンピュータの勝敗判定方法 ・return コマンド(チャンスは3回)

図2 プログラム構成と機能及び説明用例題