
プログラミング言語教育のための 適応型個別学習システムの設計と試作

黒 田 哲 也
陳 漢 雄
大 野 侑 郎

1. はじめに

1980年代後半から、大学初年度の学生に対して、いわゆる「情報リテラシー」教育がほどこされてきた。情報専門学科においては、大学等の情報専門学科における情報処理教育の実態に関する調査研究委員会による「大学の理工系学部情報系学科のためのコンピュータサイエンス教育カリキュラム J97」⁽¹⁾⁽²⁾にみられる統一的なカリキュラムの試みがある。しかしながら、情報リテラシー教育においては、「J97」のような適切なカリキュラムがまだないとみなされる。

また、「情報リテラシー」の定義はいまだ存在せず、言葉だけがひとり歩きしている感は否めない。たとえば、文科系大学で実施されている情報リテラシー教育では、特定のプラットフォームやアプリケーションの操作法・利用法の紹介に固執している例が圧倒的に多い。すなわち、「とりあえずパソコンくらいは使えない」「表計算くらいはそれなりに操作できなければ」といった考え方である³⁾。学生たちは卒業後、職場で多用するプラットフォームの操作法に関する講習会への参加や大学教育で学習しなかったプログラム言語の習得を業務上要請されることが多い。

打開策として、大野は「システム作法」を提唱した⁴⁾。いわゆるリベラル・アーツやラーニング・スキルなどの基本的素養がなければパソコンをはじめとするコンピュータサイエンス（以下、CS）の習得はのぞめないとする理念は、それまでの情報リテラシー教育になかった視点であった。ここで、基本的素養とは、日本語の論文作法、国際交流語作法、論理的抽象力などをさし、論理の基礎にはシステム論とプログラミング作法を置いた。つまり、プログラミング言語の基礎を習得しなければ、CSの習得がのぞめないとし、この習得が基本的素養やシステム論にも多大な影響を及ぼすとしている。以上より、ここでいう「情報リテラシー」は、①タッチタイプ、②日本語の論文作法（テクニカル・ライティング）、③国際交流後作法、④プログラミング基礎、の4項目から成り立つとする。

「情報処理技術者試験」のカリキュラム策定や改訂に参画し、その実施結果や企業における合格者の実態を追跡調査した結果、現行のCS教育カリキュラムで指摘されてきた諸問題は次のようにまとめられる⁽¹⁾⁽²⁾。

- (1)カリキュラム内容が一部過度に詳細すぎ、中核的（コア）カリキュラムが設定できていない。
- (2)用語の理解が勝った皮相的な知識偏重型である。
- (3)学習の学習法（ラーニング・スキル）やコンピュータ利用以前の基本的素養をとまわらない。
- (4)海外にも通用する「分散型高度情報教育システム」がない。

これら課題群は、大学等における情報リテラシー教育・情報処理教育にも一部あてはまる。

そこで、本稿ではとくにプログラミング言語教育システムの構築を提案し、その試行的開発の中間成果について報告する。具体的に、プログラミング言語の習得を可能とする適応型個別学習システムのアウトラインを設計・開発した。そこでは、システム記述言語・スクリプト言語に共通する基本言語概念を「超構文」として設定した。そのうえで、適応型個別学習システムの基本的概念を整理し、プロトタイプを開発した。

2. 適応型個別学習システムの設計

(1) 先行研究

プログラミング言語教育分野で成功した数少ない個別学習アプローチとして、T. Selker による認知科学的学習ツール“Cognitive Adaptive Computer Help: COACH”⁽¹³⁾⁽¹⁴⁾がある。COACH は LISP 言語と UNIX コマンドの習得を飛躍的に向上させた実験結果を報告している。しかしながら、学習内容は特定の言語にとどまり、ツールの動作環境も特定プラットフォームのみを保証しているため、さまざまなプラットフォームの混在する教育環境には対応できない。

プログラミング言語には機械語・アセンブラ言語・コンパイラ言語・インタープリタ言語・プログラムジェネレータなどがあり、目的のプログラムに応じた適正によって使い分けられる。そのため、たとえば大学等で CS を学んだとしても、学生は卒業後に大学教育で習得しなかったシステム記述言語やスクリプト言語を業務上習得することを要請されることが少なくない。このような状況を克服するには、特定言語の個別学習ではなく、手続き型と宣言型に共通する言語概念の基礎を効果的に学習するカリキュラムとツールが必要である。

また、前述の CS 教育カリキュラムにおける諸問題を解決するうえでとくに中核となるのは、分散型高度情報教育システムの構築である。ここで、「分散型高度情報教育システム」として、産業界から望まれている条件は次のようである⁽¹⁾⁽²⁾。

- (1) エージェント指向の適応型・知的教育情報システムである。
- (2) 日本のみならず、環太平洋諸国でもインターネットを通じて習得可能である。
- (3) カリキュラム改訂後のすべての人材類型に対する、英文・和文の対話型テキストを含む。
- (4) カリキュラム改訂後の教育関係者に対する、教育指針・教育方法などの支援機能をもつ。

(5) テキストの学習は、学生生徒のレベルに合わせた個別学習が可能である。

(6) 演習・実験・シミュレーションまわりの教材を多く含む。

(7) 旧来の集団教育より、教育効果が数倍は高い。

以上の観点から、特定の言語とプラットフォームにとどまる先行研究では学習内容や動作環境が不十分かつ好ましくないといえるし、分散型高度情報教育システムの構築も満たしてはいない。

(2) 研究の目的

本稿では、システム記述言語・スクリプト言語教育を支援する適応型個別学習システムの活用を提案する。

適応型個別学習システム（以下、本システム）は、①学習領域すべてに共通の知識フレーム、②ユーザの経験や学習過程の適応情報フレーム、③学習支援のための複数の教示モデル、④これら①-③の関係フレームから構成される適応型エージェントツールとして開発する。これにより、入門レベル・初級レベル・プロフェッショナルレベル・熟達レベルといった学習者それぞれの経験や特性に適應する学習支援型エージェントシステム⁽⁴⁵⁾が可能になる。ここで、適応型エージェント・ツールの特徴を次のように設定する。

(1) コースウェアを開発した教師が独自に、学習領域をシステムに組み込める。

(2) ユーザモデルは、個別のプログラム作成回数やプログラム管制率などの学習経験に適應する。

(3) 教示モデルは、入門・初級・中級・上級の各学習者に複数の学習モデルを提示する。学習者の特性は、超構文の熟達度に区別される。

(4) 学習領域、利用者モデル、教示モデルをまとめる知識・関係フレームをもつ。

(5) 上記学習領域の言語的要素について、支援情報記述やテキストのために、プログラミング言語に共通する個別言語に依存しない知識フレームとして「超言語な構文・意味・語用」を設定する。

(6) 教師の教材開発用に、オーサリングツールをもつ。

上記要件を満たすためには、まず学習領域全般にわたる共通のフレームを抽出する必要がある。そこで、今回は学習領域としてシステム記述言語とスクリプト言語を選択し、両言語共通のフレームとなる「超構文」の基礎研究、ユーザモデル・教示モデルのプロトタイプ開発をおこなった。

(3) 超言語の設定

「超言語」として、計算機科学（CS）の歴史における過去30年間のシステム記述言語（コンパイラ；FORTRAN / COBOL / Pascal / LISP / C / Java）・スクリプト言語（インタプリタ；UNIX shell / BASIC / HTML / JavaScript）の基本的記述が可能となる共通構文を、文字（識別子）、語、表現・式、文、プログラム単位、モジュール・ライブラリ、プログラムの7項目から成り立つと設定した。図. 2-1に、構文論からみた「超構文」の概略を示す。

①文字

文字については、可能な文字集合、語との違い、世界共通文字集合と個別文字集合、禁止文字、文字集合の種類（ISO646 符号, ASCII）の5項目の概念を含む。

②語 (word)

語では、名前、データ、演算子、述語（キーワード）の4項目の概念を種類として区別する。

名前（識別子 Identifier）は、制限と、変数・変数名・関数名・手続き名・型名・クラス名・ラベル名・モジュール名・プログラム名・ファイル名・フィールド名といった種類および表記の3項目の概念を含む。

データは、整数・実数・複素数・分数・定数からなる数値データ（記法として16進・10進・8進・2進）、文字・文字列・文字コードからなる文字データ、配列・リスト・項・オブジェクト・レコード・ファイルからなる構造データ、ポインタ・画像・音声などからなるその他データの4種を区別する。

演算子には、自然言語においては助詞や助動詞に相当することを理解させ、「句」を構成する。

述語（キーワード）には、ある単位の始まり、ある単位の終わり、制御の方式、データ記述、データ型、入出力、組込み関数名、入出力ライブラリの定義をあたえる。

③表現・式

表現・式では、算術式・データ要素式・関係式・論理式・関数式・代入式・条件式・定義式の8種を区別する。

式は何らかの計算を表現し、結果を値としてもつものと定義する。つまり式を値の等しい「語」に置き換える簡約（Reduction）操作をおこなうものとする。

演算子と被演算子の書き方には中置法・前置法・後置法の3方式の記述を含み、式の評価は「内側から・外側から」「左側から・右側から」「片側から・後ろから」の3方式の評価法をあたえることとする。

④文

文は、入出力文・代入文・条件文・ラベル付文・飛越し文・繰返し文・手続き呼出し文・データ領域割り当て文・データ領域解放文・ブロック文・宣言文・注釈文の12種を区別する。

⑤プログラム単位

プログラム単位は、関数の定義・手続きの定義・クラスの定義・抽象データ型の定義のように、ひとまとまりの仕事をする文の集まりと定義する。すなわち、

〈プログラム単位名〉(〈引数〉, ..., 〈引数〉)

〈単位の開始〉〈文の列〉〈単位の終了〉

のように記述されるひとかたまりをいい、必要な処理は「値を返す」「制御構造を決める」「継承関係を定義する」などを指す。

文の列だけからなるプログラム単位をブロックともいう。

⑥モジュール・ライブラリ

モジュールは複数のプログラム単位の集まりとし、宣言部（仕様部）と手続き部（本体部）から

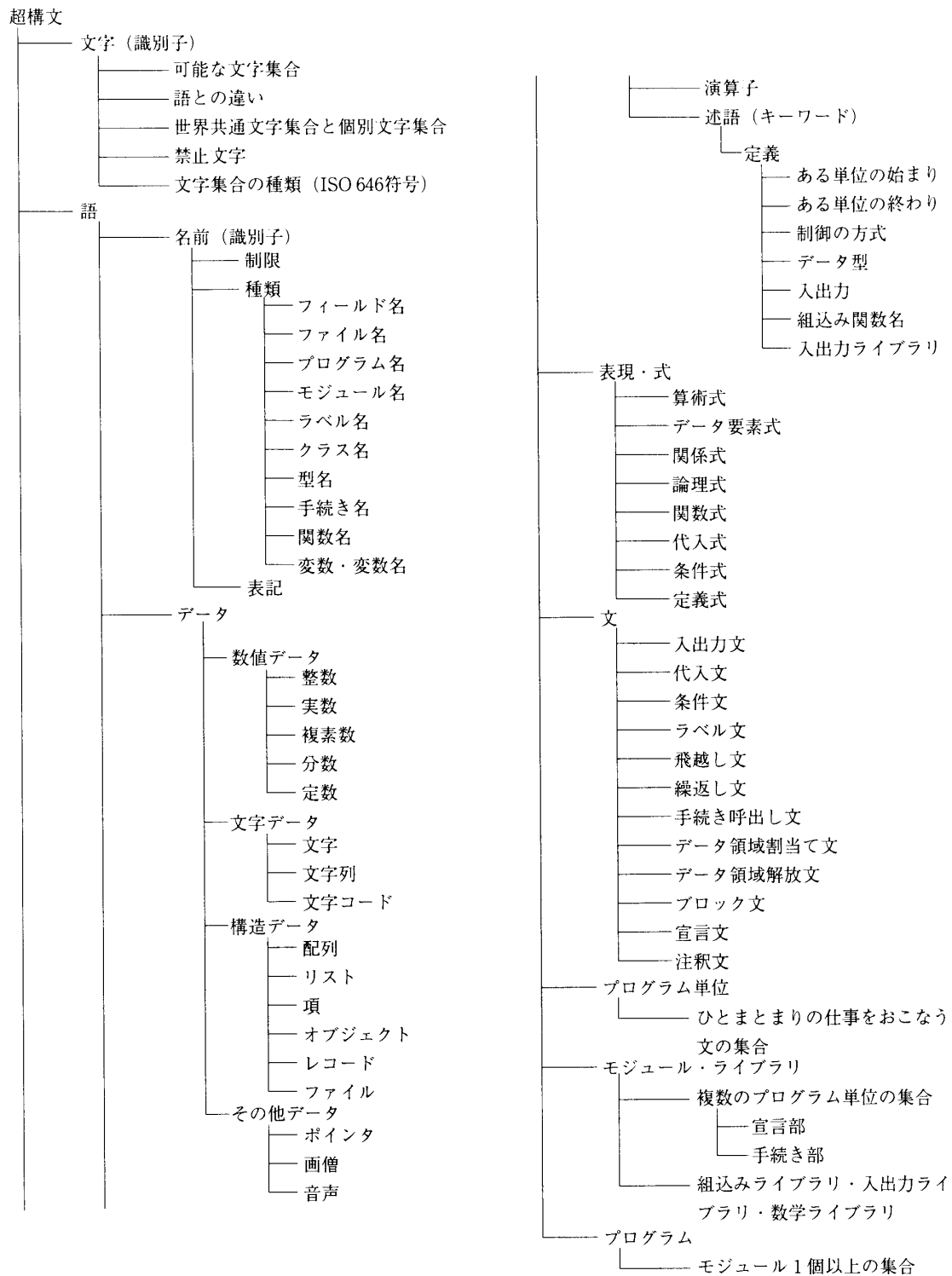


図. 2-1 構文論からみた「超構文」の概略

なると定義する。

モジュールの内容は、基本的に

〈データ・関数・手続きの情報を記述した注釈文〉

〈データ宣言文の列〉

〈関数・手続きの宣言文の列〉

〈関数・手続き定義のプログラム単位の列〉

のように記述されるものとする。

ライブラリは、組込みライブラリ・入出力ライブラリ・数学ライブラリからなるものとする。

⑦プログラム

プログラムは、モジュールの1個以上の集まりとする。

⑧インタープリタとコンパイラ

通訳系・解釈系の2処理系を定義し、区別する。

(4) システムの概要

本システムで扱う適応型エージェントツールを特徴づける基本的特徴は次の通りである。

- (1) コマンドや関数の一般的な記述を扱う。ただし、できる限り禁欲的に提示する。
- (2) 過去の使用例も含めた、実例・演習課題・ケーススタディを扱う。
- (3) 「超構文」に基づく構文を提示する。
- (4) 適切なタイミングで情報を提示する。
- (5) トピックスを情報として提示する。
- (6) レベル別提示スタイルを採用する。
- (7) 文脈センシティブな支援やフォーカスの変更など、支援のレベルを柔軟に切替え可能とする。

3. 適応型エージェント・ツールの試作

(1) 動作環境および開発環境

本システムは、分散型高度情報教育システムの一環となることをめざすから、基本的にあらゆるプラットフォームで動作するシステムとしたい。そこで今回は、Windows95/98を基本ソフトウェア（オペレーティングシステム，OS）とするDOS/V機およびMac OSをOSとするMacintoshという業界シェアのほとんどを占める2種のハードウェア上で作動させるものとした。

ソフトウェアは、インターネット・ブラウザ，米Microsoft社Internet Explorer 4.0以上，米Netscape社Communicator 4.0 (Navigator 4.0) 以上に対応させる。これらは、Windows95/98およびMac OSをOSとするプラットフォーム上で動作することが確認されているフリーウェア（無料提供のアプリケーションソフトウェア）として、もっとも流通している。

ブラウザに対応させることは、基本的にWWW上のホームページ閲覧と同様の操作を実現できる

ことと、WWW 経由で本システムを利用し学習できることを意味する。すなわち、学習者にとっては、その所在にかかわらずインターネットを通しての学習が支援され、また、コースウェアを開発する教師にとっては、HTML の知識があればホームページ制作と同様の手法で独自の学習領域をシステムに組み込むことができる。昨今のインターネット・ブームをみれば、ホームページ制作と同様の手法でコースウェアを構築できることは有利であろう。

開発言語は、JavaScript 1.2, HTML 4.0 を中核とした。

(2) システム仕様

本システムの基本仕様は以下のとおりである。基本仕様と全体構成の関係を図. 3-1 に示す。

①主題フレーム（ドメインの定義）

本システムのドメイン知識を次のように定義した。

ドメイン知識 = 主題フレーム + 適合フレーム + パーザ文法

主題フレームでは、学習対象を文・トークン（キーワード）・概念・規定集合とし、各学習対象フレームは、スロット・ユーティリティ記述・ユーティリティ活用法をもつものとした。

各学習対象は、ユーザレベルごとの事例・記述・構文といった提示情報を選択できる支援知識フレームをもたねばならない。そこで、3つないしは4つの支援スタイルは、図. 3-2 のように設定した。

②適合フレーム（ドメインについてのユーザレコード）

適合フレームは、①経験（回数）、②呼出し回数、③学習（出来）、④学習曲線、⑤ユーザ事例（ユーザの過去の誤りと修正記録）に関する情報を収集し、学習者（ユーザ）への提示に効かせるものとした。

③マルチレベルパーザと支援知識フレーム

文とトークンは構文テーブルで定義される。

文・トークン・基底集合は、関係知識スロットとユーザモデルフレームの支援・関係情報を含むものとする。このうち、関係知識スロットは、学習対象集合を必要知識（スキルドメインの部分リスト）・他の関係知識（学習対象の）をセマンティックネットに関係づける。ユーザフレームモデルは、ユーザごとの学習対象情報を提示し、ユーザによる誤答と正答を支援知識フレームに付加する。

④支援知識

支援知識として、誤りであるけれども結果が良くなったという更新規則、適合ユーザモデルを構成する知識の整合性規則、および提示規則である支援制御規則の3種の基本的規則をもつ。更新規則と整合性規則は、リアルタイムで適合ユーザモデルを更新できるものとした。さらに、提示規則として、基礎未習得規則（IF-THEN 表現の利用）、情報忘却規則（既習得済み知識の提示）、動機づけ情報提示規則（さらに良い方法がある）、詳細化禁止規則、過剰支援禁止規則の5種をあたえる

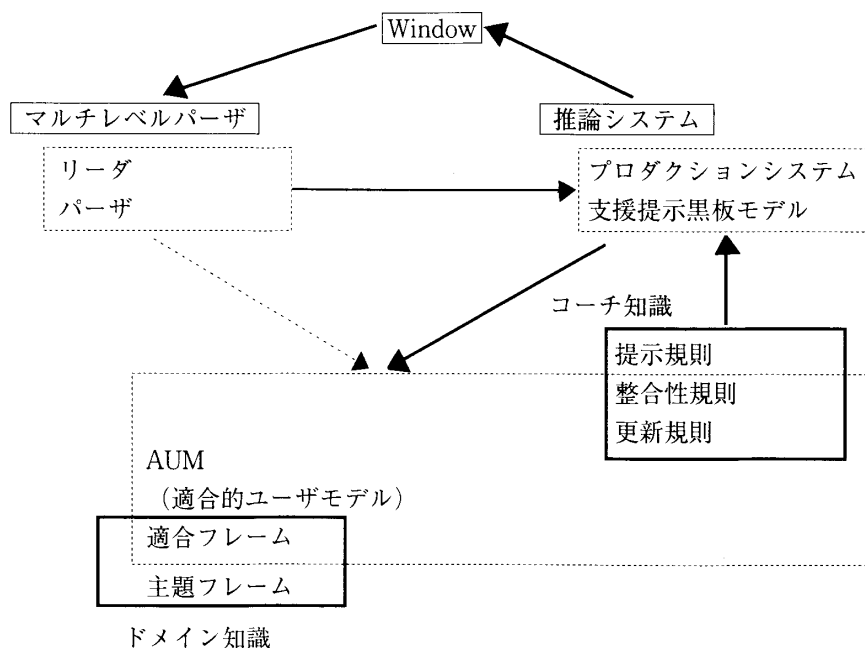


図. 3-1 基本仕様と全体構成の関係

	事例	記述	構文
入門レベル	単純	短く	最小
初級レベル	学習対象の標準的利用法		
プロフェッショナルレベル	詳細な利用法		
熟達レベル	マシンレベルの記述含む		

図. 3-2 3つの支援スタイル

こととした。

⑤推論システム

適合フレームの制御のための推論システムは、前方プロダクションシステム、黑板モデルの2種を用いる。これは、ドメイン知識とコーチ規則の利用によりゴール・コンフリクトを解消し、適合ユーザモデルに働きかける。

(3) システムの機能と画面設計

本システムは、基本言語概念の表示機能、学習者モデルの判別機能、対話機能の3機能を実現する。

①基本言語概念の表示機能

基本言語概念として設定した「超構文」に基づき、例題・例題の正解・達成度評価・例題のチェックポイント・用語解説などを表示する機能である。

②学習者モデルの判別機能

提示した例題の回答を評価し、学習者モデルに基づいて、学習者（ユーザ）をモデル別に判別する機能である。具体的には、例題の正解例を表示し、学習者自らが確認できる回答評価機能、回答強化の結果により学習者レベルをステップアップさせる学習者モデル判別機能、学習者モデルに合わせた分類評価機能を有する。

③対話機能

学習者のプログラム入力を可能とし、入力されたプログラムを実行するプログラム入力・実行機能と各画面に応じて実行可能な機能を操作ボタンで表示し、ボタンをクリック（押下）によりコマンドを実行するコマンド実行機能の2機能からなる。

以上3機能を実現するため、本システムは、扱う情報の種類により分類される①システム名・目次情報フレーム、②学習者名・学習者モデル情報フレーム、③出力情報フレーム、④ユーザ入力フレームの4フレーム分割表示を基本とする。基本画面フォーマットを、図. 3-3 に示す。

（4）プロトタイプ開発

対象言語を、HTML、JavaScript とし、とくに適合フレームと主題フレームからなるプロトタイプを開発した。

主題フレームに表示する内容は、上記（2）①で説明した3項目からなる学習内容である。同様に、適合フレームに表示する内容は、上記（2）②で紹介した5項目である。各フレームの表示内容とシステム基本仕様の関係詳細図を図. 3-4 に示す。

（5）事例と実行結果

学習者モデルの基本シナリオを、図. 3-5 に示す通り設計した。

学習者モデルの事例として、以下に示す「JavaScript によるテキスト表示のシナリオ」を試作した。

実行の結果、プロトタイプが Windows 95 / 98 上の IE 4.0、NC4.5 および Mac OS 上の NC 4.5 で問題なく動作することが確認できた。現在、ブラウザの Cookie 処理に一部取扱いの異なる個所があるため、学習者モデルの判別機能を両プラットフォーム上で統一させられない。しかしながら、今後、ブラウザの改訂により統一させられることが確認できている。

○「JavaScript によるテキスト表示のシナリオ」の事例：

- (1) テキストエディタを立ち上げさせる。
- (2) ブラウザを立ち上げる。
- (3) スクリプトを入力させる。
- (4) script.htm に文書を保存させる。
- (5) ブラウザに切り替える。

(6) ファイルをまたオープンする。

(7) 表示結果を調べる。

① 表示が正しいとき

② 表示が正しくないとき

質問1 SCRIPT がタグでかこまれているか

HTML 文書に JavaScript コードを入れるときは、いつもスクリプトを<SCRIPT>…</SCRIPT>で囲まなければなりません。

そして、スクリプト言語として、JavaScript をLANGUAGE=“JavaScript”と指定します。

質問2 大文字小文字の区別ができていないか

JavaScript のなかでは、HTML 文書とは違って、すべて大文字と小文字が区別されますので、そのように書く必要があります。

質問3 HTML と区別ができていないか

JavaScript は、HTML のコメント開始タグ

<!— …

を無視します。JavaScript コードの中にコメント行を入れたいときは、

／／

でコメント行を開始します。

質問4 SCRIPT の構文は正しいか

コメントも含めて、スクリプト全体は、HTML のコメントタグ

<!-- …

の中に入っていないければなりません。

質問5 関数 document.write の構文は正しいか

関数 document.write () は、文字列を出力しますが、引数は

静的文字列 Last updated on

評価済み文字列 document.lastModified

ピリオド .

を含みます。

質問6 “+” と “空白” は正しいか

引数の文字列はプラス記号+で結合されます。空白も解釈されます。

質問7 プロパティのための “.” を理解しているか

オブジェクト document のプロパティは、ピリオド以下によって示されます。

この例の正確な解釈は

the LastModified property of the document object

質問8 “updated” を “changed” に変更して結果が正しいかもう一度やり直してください。

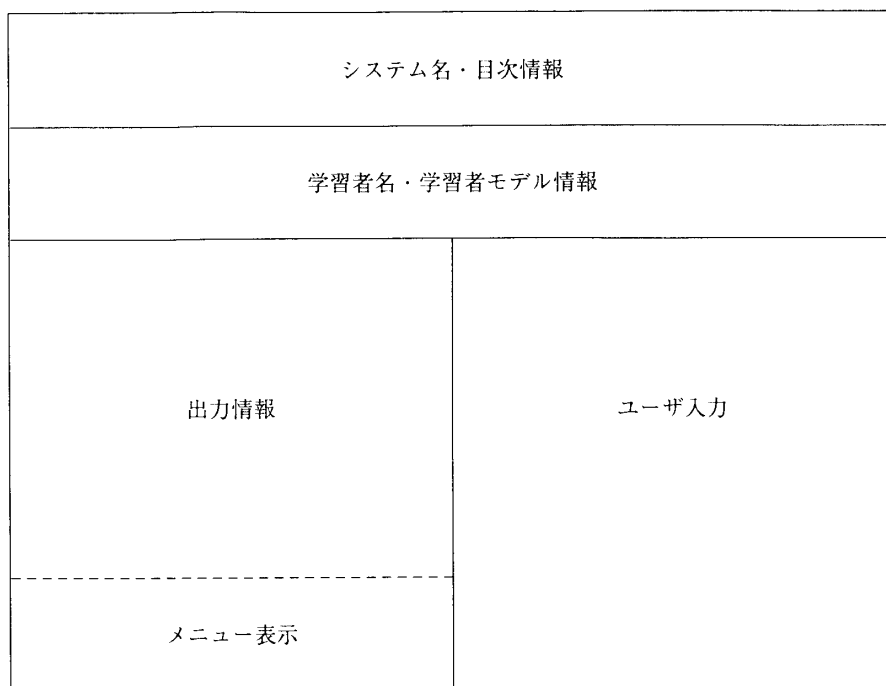


図. 3-3 基本画面フォーマット

| 〈適合フレーム〉 | 〈主題フレーム〉 | | |
|---|--------------------|----------------------------|-----|
| ユーザモデル特性・スロット
経 験
所要時間
達成度
学習曲線
ユーザ記録

→ | 文・トークン
(構文テーブル) | 記 述

概念と基底集合の
リスト | 事 例 |
| 入 門 | | | |
| 初 級 | | | |
| 中 級 | | | |

関係知識
スロット

=必要 (スキルドメインの部分的リスト)
他の関係知識 (学習対象のリスト)

ユーザモデル
フレーム

=ユーザごとの学習対象情報のリスト
ユーザによる誤答と正解は支援知識フレームに付加

図. 3-4 各フレームの表示内容とシステム基本仕様の関係詳細図

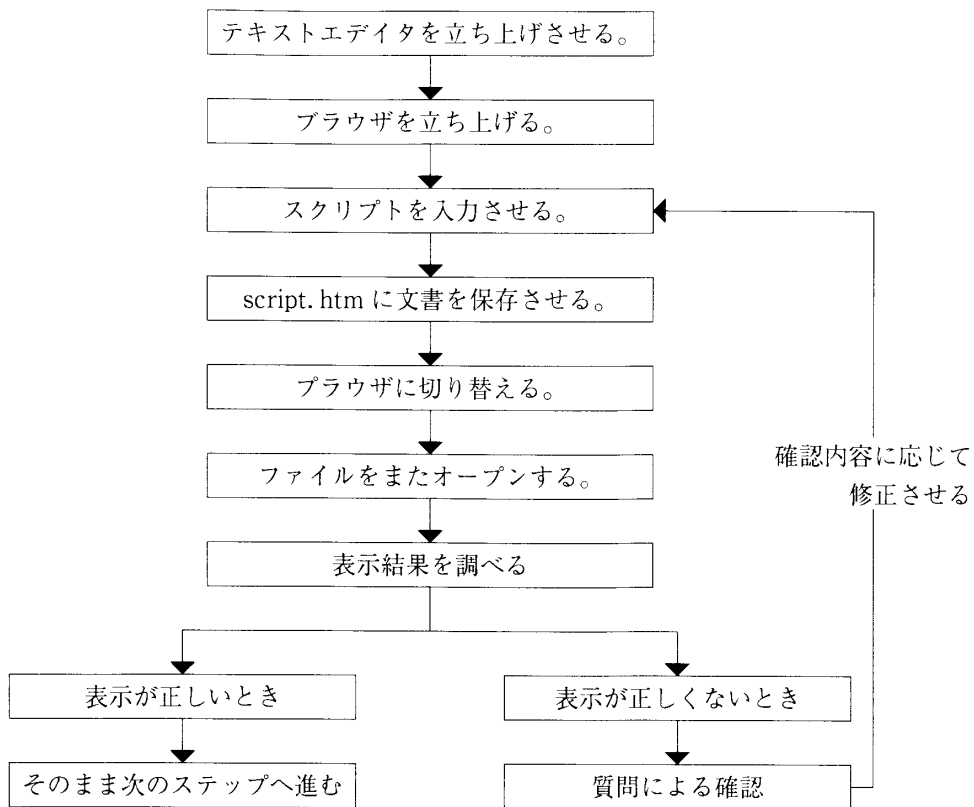


図. 3-5 学習者モデルの基本シナリオ

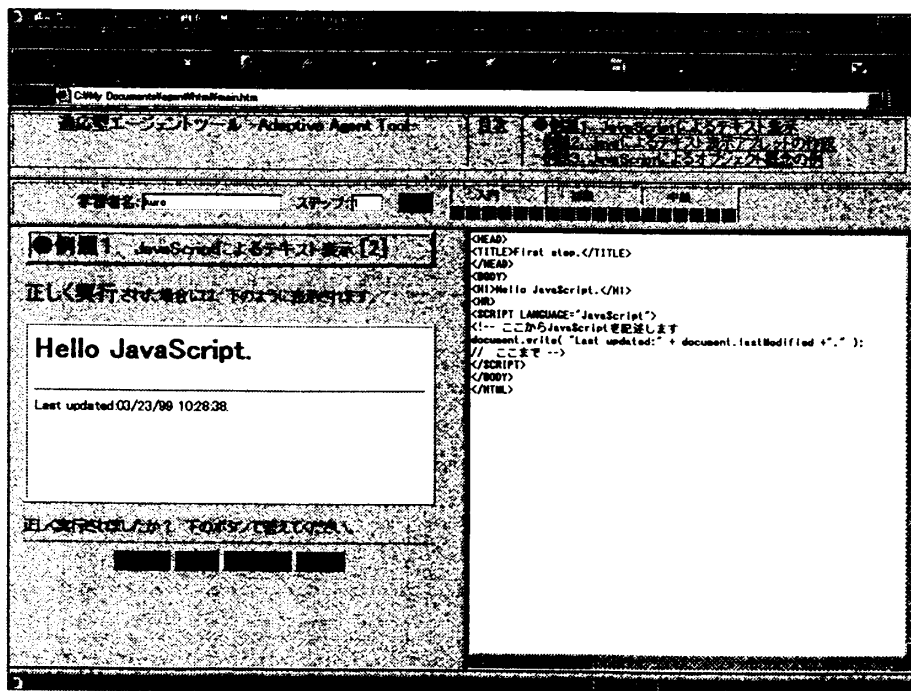


図. 3-6 DOS/V 機上でのプロトタイプ実行画面

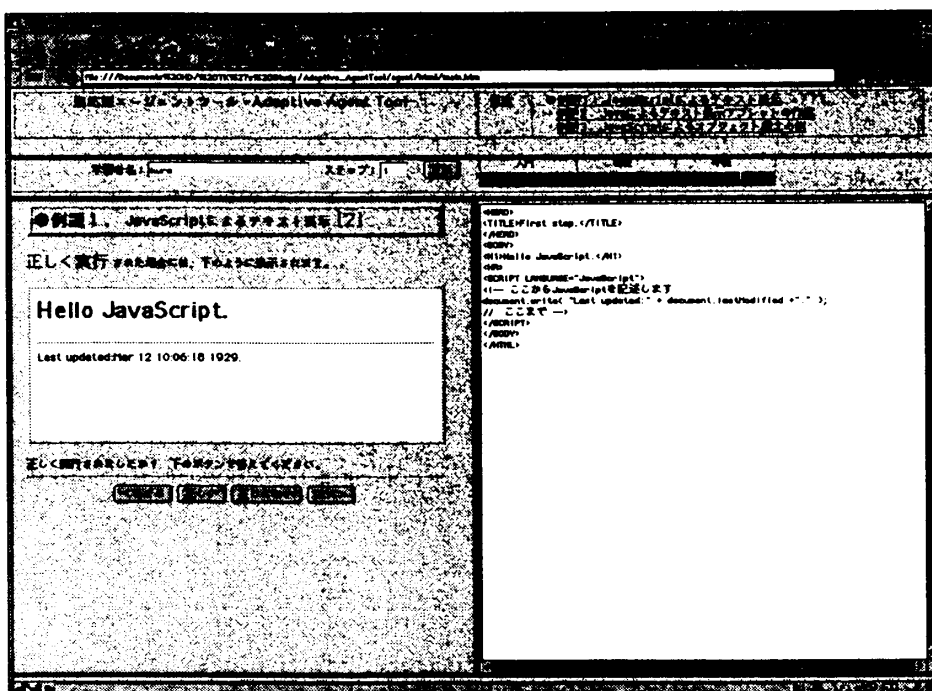


図. 3-7 Macintosh 上でのプロトタイプ実行画面

4. まとめ

本稿では、プログラミング言語教育の支援をめざす適応型個別学習システムの構築について検討した。学習者それぞれのプログラム作成回数やプログラム完成率等の経験度に適応する学習支援型エージェントシステムとして、①学習領域すべてに共通の知識フレーム、②ユーザの経験や学習過程の適応情報フレーム、③学習支援のための複数の教示モデル、④これら①-③の関係フレームから構成されるシステムのアウトラインを設計・開発した。そのうえで、本システムで扱う学習領域のうち、システム記述言語・スクリプト言語の構文論からみた「超構文」を定義する基礎研究をおこなった。

これら基礎研究をもととして、本システムの基本仕様のうち、とくに主題フレームと適合フレームの実装を実現する適応型エージェントツールを試作した。プロトタイプは、Windows95/98を基本ソフトとするDOS/V機、Mac OSを基本ソフトとするMacintoshの両プラットフォームで動作することが確認できた。

マルチレベルパーザと支援知識フレーム・支援知識・推論システムを実装したうえで、以下の項目について評価することを今後の課題としたい。

- (1) 本システムの本格的実験・評価
- (2) 「超構文」の有用性評価
- (3) 類似システムとの比較・評価。現在のところ、Java、JavaScriptのような通訳系・解釈系の学習を両方支援する類似システムはないので、将来の課題である。

さらに、プログラミング言語の基礎だけでない別の学習領域を選定し、とくに「日本語論文作法」や「国際交流語作法」といった領域まで対象ドメインを拡張することをめざしたい。

5. おわりに

本稿は、日本私立学校振興・共済事業団『平成10年度私立大学等経費補助金特別補助「特色ある教育研究の推進」』課題「プログラム言語学習のための適応型エージェントツール開発・実験」の成果の一部をまとめたものです。ご教授・ご支援・ご協力いただいた関係各位に、ここに深く感謝いたします。

(くろだ・てつや 産業情報学科)
(ちん・かんゆう 産業情報学科)
(おおの・としろう 産業情報学科)

参考文献

- (1)中央情報教育研究所 (1998)『平成9年度国際化に対応した情報処理技術者の育成に関する調査研究報告書』。通商産業省委託研究，日本情報処理開発協会。
- (2)中央情報教育研究所 (1999)『平成10年度国際化に対応した情報処理技術者の育成に関する調査研究報告書』。通商産業省委託研究，日本情報処理開発協会。
- (3)伊藤華子 (1998)『パソコンプログラミング入門以前』。毎日コミュニケーションズ。
- (4)大野尙郎 (1996)『パソコンでつきあう』(情報フロンティアシリーズ第11巻)。共立出版。
- (5)大野尙郎・黒田哲也・反町洋一・陳漢雄・増田文男 (1999)『平成10年度適応エージェントツール開発・実験 研究報告書』。つくば国際大学。
- (6)大野尙郎・黒田哲也・反町洋一・陳漢雄・増田文男 (1999)「プログラム言語学習のための適応型エージェントツールシステム仕様書」，『プログラム言語学習のための適応型エージェントツール設計書』。つくば国際大学。
- (7)大野尙郎・黒田哲也・反町洋一・陳漢雄・増田文男 (1999)「プログラム言語学習のための適応型エージェントツールソフトウェア設計書」，『プログラム言語学習のための適応型エージェントツール設計書』。つくば国際大学。
- (8)大野尙郎・黒田哲也・反町洋一・陳漢雄・増田文男 (1999)「プログラム言語学習のための適応型エージェントツール学習者モデル説明書」，『プログラム言語学習のための適応型エージェントツール設計書』。つくば国際大学。
- (9)大野尙郎・黒田哲也・反町洋一・陳漢雄・増田文男 (1999)「プログラム言語学習のための適応型エージェントツール基本言語概要説明書」，『プログラム言語学習のための適応型エージェントツール設計書』。つくば国際大学。
- (10)大野尙郎・黒田哲也・反町洋一・陳漢雄・増田文男 (1999)『プログラム言語学習のための適応型エージェントツール操作手引書』。つくば国際大学。
- (11)大学等の情報専門学科における情報処理教育の実態に関する調査研究委員会 (1998)『大学等の情報専門学科における情報処理教育の実態に関する調査研究 [文部省委嘱調査研究] 平成9年度

報告書』。情報処理学会。

- (12) 大学等の情報専門学科における情報処理教育の実態に関する調査研究委員会（1999）『大学等の情報専門学科における情報処理教育の実態に関する調査研究 [文部省委嘱調査研究] 平成10年度報告書』。情報処理学会。
- (13) Selker, T. (1994) "Coach: A Teaching Agent that Learns", Communications of ACM, Vol.37, No.7, 92-99.
- (14) Selker, T. (1999) "CoACH - Cognitive Adaptive Computer Help". <http://www.almaden.ibm.com/cs/showtell/ui/coach.html>.
- (15) 山田誠二（1997）『適応エージェント』（認知科学モノグラフ第8巻）。共立出版。

Abstract

The design and implementation of the agent-based learning system for education of programming languages

Tetsuya Kuroda, Hanxiong Chen and Toshiro Ohno

We examined learning systems of the independent adaptative type for supporting the education of programming languages. We set up the learning domain of programming languages, and designed the agent-based learning support system which could be adapted to each learner's expertise and capability.

The system consists of the following frames:

- (1) The knowledge frame common to the all learning domain.
- (2) The adaptative information frame based on the user's expertise and the learning process.
- (3) The multiple instruction model for the learning support.
- (4) The related frame between (1) and (2).

We defined the "meta-syntax" applicable both for the system description languages and the script languages. We developed the prototype of an adaptative type agent tool based on the above basic research.

As a result of the experiment it is confirmed that the prototype functioned well both on Macintosh with MacOS and a DOS/V machine with Windows 95/98.

Keyword: Adaptive Agent Tool, Computer literacy, General Informatic Education