

実習授業における学生探索を目的とした 自律エージェントシステムの開発

松田 承一^{†1} 冷水 友彦^{†2}
高野 辰之^{†1} 宮川 治^{†2}

大学などで行われる実習授業において教員やアシスタントは教室を巡回しながら課題に取り組む各学生の途中経過や結果に応じて必要な指導を行う。しかし、授業時間内において指導が必要な学生全員を把握することは難しい。また、各教員やアシスタントによる指導の判断基準の違いから、学生は適切な教育を受けられない場合がある。そこで本稿では、自律エージェントの非同期的な動作により学生の探索を行うシステムの開発について述べる。

Development of Autonomous Agent System which Aims to Student Search in Practical Lesson

SHOICHI MATSUDA,^{†1} TOMOHIKO SHIMIZU,^{†2}
TATSUYUKI TAKANO^{†1} and OSAMU MIYAKAWA^{†2}

The practice lesson is performed at the university. In these lessons, teachers and assistants are patrolling the inside of a classroom. They teach to the students who tackle a exercise. However, it is difficult for instruction to grasp all required students in within the time. Moreover, there is a difference in the criterion of judgment of instruction by each teacher or an assistant. Therefore, the student cannot receive suitable education. This paper describes development of the system which looks for a student by asynchronous operation of autonomous agents.

1. はじめに

大学などで行われる実習授業では、学生は教員に指示された課題に取り組む。このとき教員やアシスタント(以下指導役)は教室を巡回しながら、挙手などにより疑問・質問のあることを示す学生や、各学生の課題に対する途中経過や結果などから、必要に応じてそれらの学生へ指導を行う。しかし、挙手を行えない学生が存在することや、授業時間内にすべての学生の詳細な途中経過や結果を把握することは難しく、指導すべき学生を見過ごす場合がある。また、各指導役が持つ指導の判断基準が異なるため統一された基準による学生の探索は難しい。これらの問題から、理解が進んでいないなど特定の条件に合致した学生が存在しても、指導役は実習時間内にこれらの学生の把握を行えていない場合がある。これにより、特に理解に苦しむ学生に対して指導が行われていない場合、その学生の理解度が低下する恐れもある。

近年では、学生の途中経過や結果の情報を収集し、システムによりそれらの情報を様々なかたちで活用する研究が行われている。情報の活用には、ユーザが必要に応じて情報を取得し閲覧や分析を行う場合と、システムが動作を行うにあたって必要な学生の情報や周辺環境を積極的に利用する場合がある。

ユーザがシステムを用いて必要に応じて情報を取得し閲覧や分析を行う研究として、奥井ら¹⁾は、学生の講義中の反応をレスポンスアナライザにより収集し、分析結果による説明方法と教材の改善を試みた。金西ら²⁾は学習者の Web 上の教材の閲覧記録に着目して学習履歴を収集し、教師は教師支援システム LOGEMON によってそれらの閲覧記録を視覚化する研究を行った。瀬下ら³⁾は、中学校における調べ学習などの学習者主導の授業において、調べ学習支援システムの利用による情報収集を行いそれらの情報を定量的に可視化・分析した。

また、システムが動作を行うにあたって必要な学生の情報や周辺環境を積極的に利用する研究として、エージェントを利用したものがいくつか挙げられる。水谷ら⁴⁾はシームレス Web 型学習システムの試作を行った。システムではエージェントを利用することで、学生

^{†1} 東京電機大学大学院
Graduate School of Tokyo Denki University

^{†2} 東京電機大学
Tokyo Denki University

が自宅学習などとして使用する学習支援システムを様々なネットワーク接続状態でもシームレスに利用可能とさせることや、学習支援システムのコンテンツマネージメントを行わせること、レポート提出を様々なネットワーク接続状態においても行うことなどを可能とした。白田⁵⁾は放送教育番組のデジタル化に着目し、双方向機能をもったマルチメディア教育支援システムのモデルについて述べた。学習者の SOS 信号により、エージェントが動作し原因分析や学習ユニットの探索、学習者との対話などを行う。

本稿では実習授業において学生探索を目的とした自律エージェントシステムの開発を行う。システムでは従来各指導役の行っていた学生の探索を自律エージェントが収集された情報を積極的に利用することで行う。エージェントが探索行動を自律的に行うことにより、従来のシステムには難しい、指導役への能動的な情報の配信やシステムの役割の局所化が可能となる。これにより、教員の行う探索行動により近い情報の探索を行うことが可能であると考えられる。本システムにより、指導役は自ら学生間を探索せずとも統一された基準でシステムによって探索された学生に指導を行えるなどの利点が得られる。また指導役がより効率的な指導を可能にすることから、ダイナミックな授業改善に取り組める。

本稿では、システムにおける自律エージェントとシステム構成、試験導入とその結果などに関して述べる。

2. 本システムにおける自律エージェント

本システムでは、自律エージェントが学生探索を行う。自律エージェントとは自らが目的を持ち、動作環境の変化などに応じてその目的達成のための動作を自律的に行うエージェントである。また、エージェントは成長性を持つことや複数のエージェント同士が互いに連携して動作することを特徴として持つ。

自律エージェントを利用することで、指導役自身が収集した学生のデータなどから判断を行う必要がなくなることや、情報の探索方法を限定せずに各エージェントによって自由な探索行動ができ、探索の結果となる学生の情報を統一的に扱えることなどが利点として挙げられる。

本章では、エージェントがどのようなモデルであるかについて述べる。

2.1 実習授業における指導役の動作

指導役は課題に取り組む学生間を巡回しながら、理解の進まない学生などの探索を行い指導する。概念図を図 1 に示す。指導役は特定の学生を発見するために複数の判断基準を持つ

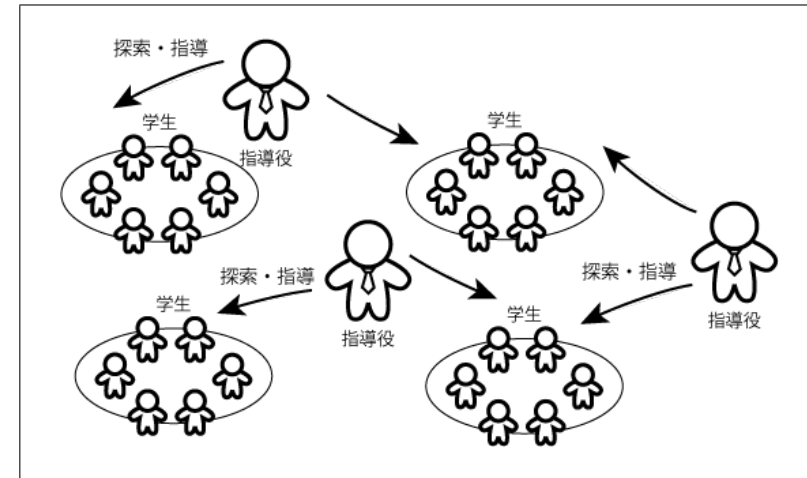


図 1 指導役の動作の概念図

ている。これらの基準の多くは課題の回答や途中経過に関する基準である。しかし、各指導役により判断基準がやや異なる場合がある。

自律エージェントが学生の探索を指導役に代わって行うために、実習授業において指導役がどのような動作を行うかについて適切に理解する必要がある。適切に指導役の動作を理解することにより、エージェントがより適切に学生の探索を行うことが可能である。

2.2 モデル

自律エージェントは主に探索行動を指導役に代わって行う。エージェントは自律的に動作し、それぞれが学生発見のための基準を持ちながら学生から収集した情報を利用して動作する。これにより指導役は自ら学生を探索せずにエージェントの通知を受け取り、必要に応じて指導を行うこととなる。概要図を図 2 に示す。

一人の指導役が持つ判断基準は多く、複雑であるためこれを単独の自律エージェントに行わせることは難しい。このことから自律エージェントと指導役それぞれの探索の異なる点として、システム上で動作する複数の自律エージェントはそれぞれが一つの学生発見のための判断基準を持つ。これにより、各自律エージェントの動作は単純化する。仮に学生が複数の要因を持ち合わせる場合でも、自律エージェント同士の協調によりその要因を網羅的に把握

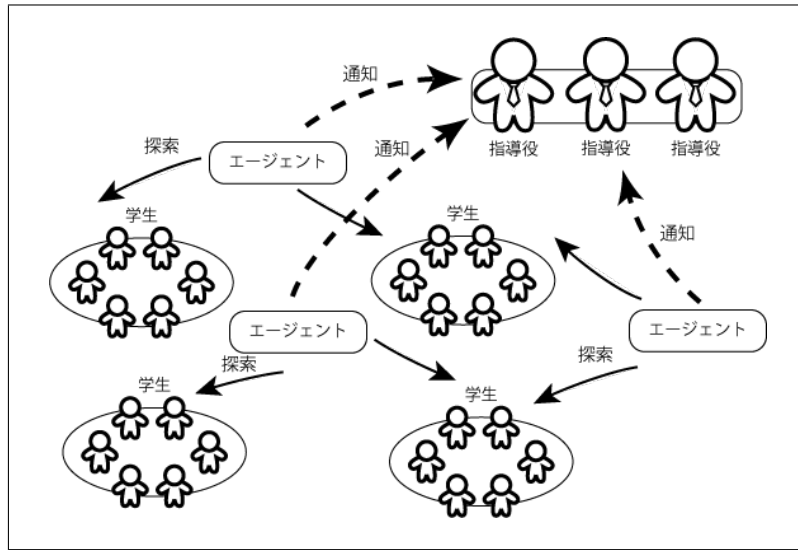


図2 エージェント動作の概念図

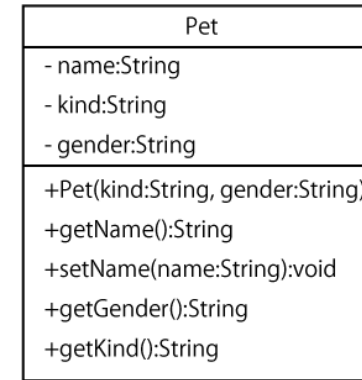


図3 クラス図例

することが可能である。また利点として、新たな判断基準による探索を導入する場合でもその判断基準を持ったエージェントを動作させればよく、それまでに動作していたエージェントには影響を及ぼさずに済むことが挙げられる。

また、エージェントは探索の結果となる情報に対する処理を動的に切り替え可能である。このため、探索の結果となる学生の情報を指導役に配信することやデータベースへ処理を行うなど様々な用途に柔軟に利用することが可能である。

3. 対象とする授業

本学の実習科目である環境プラクティスを対象とする。環境プラクティスはいくつかのコースから構成されており、本システムは特にソフトウェア・データベースコースのうちソフトウェアをテーマとした実習(以下本授業)を対象として開発を行った。

学部2年次を対象とした授業であり、履修人数は30名程度を想定している。教員2名・アシスタント3名の計5名で授業を進行する。

本授業は2つのテーマから構成されており、ソフトウェアをテーマとしてJava言語によるソースコードの最初からの記述であるスクラッチプログラミングを行い、データベースをテーマとしてSQL言語を使用している。

3.1 授業内容

授業ではオブジェクト指向に基づいたプログラミングを行う。複数のクラスやインタフェースを扱うことやデータベースを利用するなどのやや発展的な内容を扱う。

課題はUMLにおけるクラス図を用いて出題される。学生に図3に示すようなクラス図を課題として与え、学生は示されたクラス図に基づいたプログラミングをスクラッチで行う。最終的に目標とする解答を図4に示す。また、このほかに作成したクラスを利用する実行文を記述する必要がある。

課題はいくつかの段階から構成される。はじめに、学生はまずクラス図からメソッドの実装を除いたクラスの骨格部分のみを記述(以下機械的導出)する。次に、メソッドの実装をしてクラスを完成させる。最後に、作成したクラスを利用する実行文を記述し実習課題を終了する。各段階において学生は指導役に確認を受け、次の段階へと進む。

3.2 重視している点

本授業におけるプログラミング教育では、学生にスクラッチプログラミングを行わせる。目的の一つは穴埋め方式や選択式での解答の方法と比べて、多くのプログラムを書かせることによりたくさんの失敗を経験させ理解度向上を目指すためである。

```
public class Pet{
    private String name;
    private String kind;
    private String gender;
    public Pet(String kind, String gender){
        this.kind = kind;
        this.gender = gender;
    }
    public String getName(){
        return this.name;
    }
    public void setName(String name){
        this.name = name;
    }
    public String getGender(){
        return this.gender;
    }
    public String getKind(){
        return this.kind;
    }
}
```

図 4 解答ソースコード

学生にスクラッチプログラミングを行わせることにより、指導役は実装に関するものや、コーディングスタイルを守らせるなどの広い範囲での指導を行う必要がある。その中でも表 1 に示すような、基本的な項目に関しては特に重点をおいて指導を行っている。

各指導役は学生のコンピュータの画面を複数の基準において確認しながら指導を行うため、各学生のソースコードを注意深く観察する必要がある。そのため多くの時間を要することから、学生全体を探索するにあたって多大な負担となっている。

表 1 指導のための基本的な判断基準

項目名	説明
コンパイル	コンパイル可能なソースコードか
インデント	正しいインデントがなされているか
機械的導出	仕様に沿ったソースコードが完成しているか
入力方法	ソースコードを適切な入力順で入力しているか
課題の未解決	一定時間以上進んでいない

4. 開発したシステム

本システムは学生探索システムと学生情報収集システムの 2 つから構成される。概要図を図 5 に示す。

システムでは現在のところ、学生情報収集システムとして課題提出・評価システム⁶⁾を利用している。これらのシステムが連携することで指導役の探索をシステムとして行うことが可能となる。

4.1 学生探索システム

学生探索システムは指導役に対して探索結果のプッシュ型の通知を行う。指導役は通知を受け付けるクライアントプログラムとして CUI で作成されたプログラムを動作させておくことで情報の確認が行える。指導役が利用する CUI 画面の外観を図 6 に示す。この画面では通知を受け取る度に自動的にそれらの情報が表示される。プッシュ型の通知を行うことから、指導役はシステムに対して問い合わせをせずに情報を得られる。そのため、指導役の業務への妨げにならないシステムの導入が可能である。

学生探索システムで動作する自律エージェントは学生の探索を行う。自律エージェントは後述する学生情報収集システムに問い合わせを行い、自身の判断基準による学生の探索を行う。探索の結果となる学生の情報は学生探索システムによってあらかじめ指示された処理をする。ここでの処理は、指導役に対するプッシュ型の配信である。

4.2 学生情報収集システム

学生情報収集システムは、学生の課題に対する途中経過や結果を収集・管理する。集められた情報は学生探索システムの探索時に利用される。

現在は、学生情報収集システムとして課題提出・評価システム⁶⁾を利用している。課題提出・評価システムは学生に対してソースコードをスクラッチで記述する課題を提示し、回答

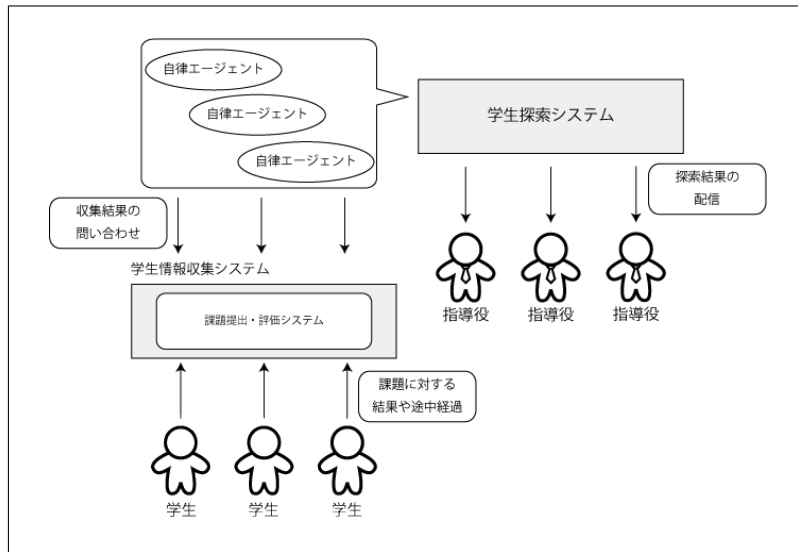


図 5 システムの概要図

の提出を受け付ける。また、回答されたソースコードに対する評価を行う。課題提出・評価システムの評価内容を表 2 に示す。

表 2 課題提出・評価システムの評価項目

項目名	説明
コンパイル	コンパイル可能なソースコードを提出しているか
インデント	正しいインデントを行ったソースコードを提出しているか
機械的導出	クラス図に沿ったソースコードの骨組みが完成しているか

4.3 動作の流れ

システムの動作の流れを図 7 に示す。

学生は学生情報収集システムに課題に対する途中経過や結果を送信する。学生情報収集システムではこれらの情報を管理する。

学生探索システムは自律エージェントに対して情報の探索結果を各指導役に通知することをあらかじめ指示しておく。自律エージェントはそれぞれ非同期に学生情報収集システムに対して問い合わせを行い、返却された情報から探索を行う。探索の結果となる情報が存在した場合、自律エージェントは学生探索システムによって指示された動作である、指導役への通知を行う。探索の結果が存在しない場合は、何の動作も行わない。

通知を受ける指導役はまず学生探索システムに対して登録作業を行う。登録はクライアントプログラムを動作させることで行う。登録を行うことによって探索の結果が通知される。

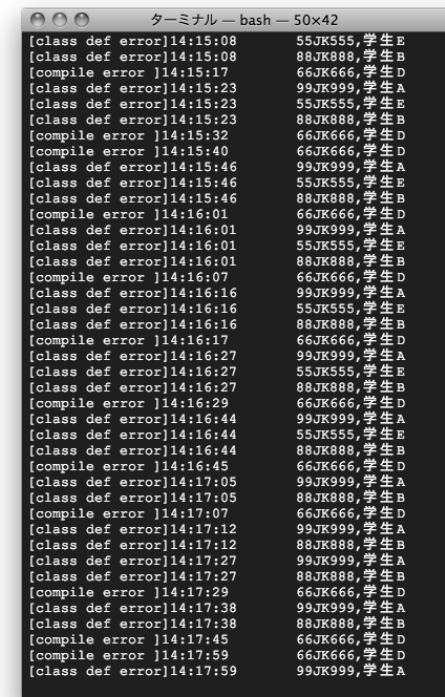


図 6 指導役クライアントプログラムの外観図

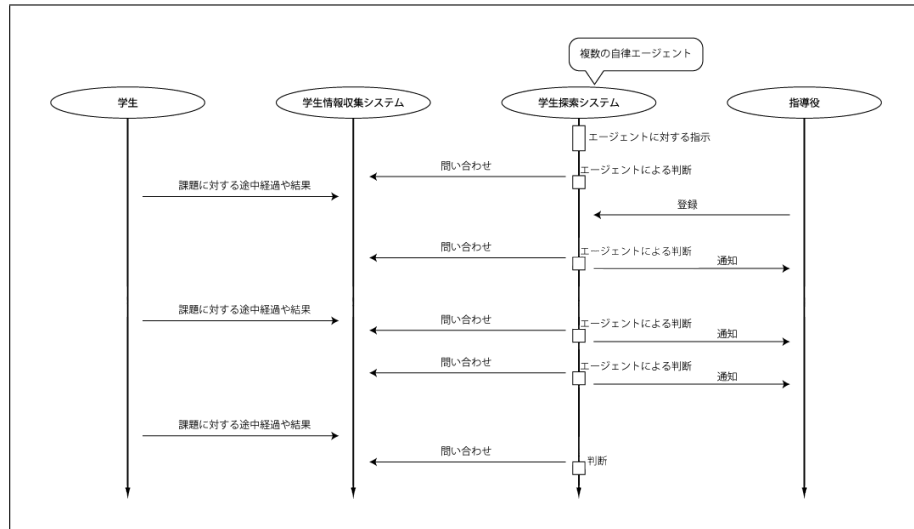


図 7 システムの動作の流れ

通知を受けた指導役はその情報から実際に学生に指導を行うことが可能となる。

4.4 エージェントの実装

本システムの自律エージェントは JavaVM 上で動作するプログラミング言語である Scala 言語を用いて実装している。エージェントのライブラリ階層部分の簡略化したクラス図を図 8 に示す。

それぞれのエージェントは、AgentBase クラスを継承することでエージェントとして振る舞う。AgentBase クラスは HasGoal クラスと HasRequestConfig クラスの 2 つのクラスをミックスイン継承する。ミックスイン継承によりメソッドを実装とともに継承することが可能である。また AgentBase クラスは Scala 標準の API である Actor クラスを継承する。Actor クラスを継承したクラスのオブジェクトは独立したスレッドで動作可能となる。

各クラスの詳細について述べる。HasGoal クラスは学生を探索した後に行う動作を関数オブジェクトとして動的に保持可能となるトレイトクラスである。関数オブジェクトを追加するための関数やすべての保持する関数オブジェクトを実行するメソッドを定義する。HasRequestConfig クラスは学生情報収集システムに対してリクエストを送る際のパラメー

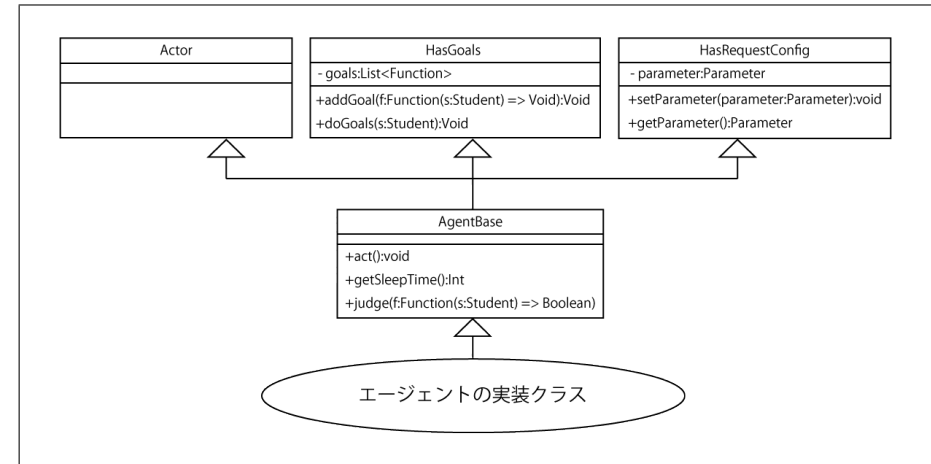


図 8 エージェントのライブラリ階層のクラス図

タを保持可能となるトレイトクラスである。パラメータ設定を変更するためのメソッドを定義している。AgentBase クラスはどのような学生を探索の対象とするかの判断を行うためのメソッドを抽象メソッドとして定義している。各エージェントによりどのように学生の探索を行うかが異なることから抽象メソッドとして定義している。また、エージェントがどの程度の間隔で探索を行うかをミリ秒単位で指定するメソッドも同じく抽象メソッドとして定義している。これらはそれぞれ、GoF デザインパターンにおける Template パターンが適用されており AgentBase クラスのサブクラスではどのようにその抽象メソッドが利用されるのかを気にせず実装を行うことが可能である。

次に現在までに実装した自律エージェントについて述べる。現在は表 3 に示す判断基準を持った自律エージェントがシステム上で動作している。各自律エージェントは一つの判断基準を持ち、学生情報収集システムに問い合わせを行うことでその結果からそれぞれの判断を行う。ここで、課題提出・評価システムは授業に参加している学生の情報を返却し、返却する学生の情報には回答に対するコンパイル・インデント・機械的導出に対する評価結果が含まれる。エージェントは課題提出・評価システムに問い合わせ、授業において収集された学生の情報をすべて取得し含まれる採点結果の中から最新の採点結果を参照し判断基準に

より学生の探索を行う。

表 3 エージェントが持つ判断基準

項目名	説明
コンパイル	コンパイル可能なソースコードを提出しているか
インデント	正しいインデントを行ったソースコードを提出しているか
機械的導出	仕様に沿ったソースコードの骨組みが完成しているか

5. 試験導入と結果

本授業において行った試験導入とその結果、またシステムの導入によって起きた事例について述べる。

2010年10月12日の授業において試験導入を行った。対象人数は33名である。

5.1 試験導入の方法

学生情報収集システムの一つである課題提出・評価システムを利用して、図9に示す互いに関連するBabyクラスとNurserySchoolクラスの2つのクラスを課題として提示した。解答は図10に示す通りである。学生は課題開始時に課題提出・評価システムのクライアントプログラムを起動し、回答できたらクライアントプログラムを利用してそのソースコードをシステムに提出する。

5.2 結 果

表4に試験導入によって得られた探索人数を分単位で集計したデータを示す。注意する点として、ある時刻の検出者数で表されるデータは同一の学生を示す場合がある。これは、システムでは直近の検出者が新たな回答を提出していないことなどに対する配慮を行っていないためである。

現在、システムでは提出結果に対するコンパイル・インデント・機械的導出それぞれの誤答を判断基準とした学生の探索を行ったが、システムの探索により、ほぼ毎分にわたってそれらの判断基準による学生の探索を行えた。

5.3 事 例

これまでは指導役は指導すべき学生を見つけるために教室内を巡回していたが、システムの導入により指導を行う学生のもとへすぐさま向かい指導を行うことができるようになり、

表 4 試験導入時の集計

時刻 (時:分)	合計検出者数	コンパイル検出者数	インデント検出者数	機械的導出検出者数
14:05	6	0	0	6
14:06	6	0	0	6
14:07	10	0	0	10
14:08	6	0	0	6
14:09	6	0	0	6
14:10	9	1	0	8
14:11	17	7	0	10
14:12	12	4	0	8
14:13	13	3	0	10
14:14	13	5	0	8
14:15	12	3	0	9
14:16	17	5	0	12
14:17	14	4	0	10
14:18	9	3	0	6
14:19	7	3	0	4
14:20	9	5	0	4
14:21	3	0	0	3
14:22	5	0	0	5
14:23	5	0	0	5
14:24	6	2	0	4
14:25	3	0	0	3
14:26	5	0	0	5
14:27	5	0	0	5
14:28	7	3	0	4
14:29	6	3	0	3
14:30	9	5	0	4
14:31	5	3	0	2
14:32	3	3	0	0
14:33	5	5	0	0
14:34	4	4	0	0
14:35	1	1	0	0
14:40	7	0	2	5
14:41	12	3	2	7
14:42	3	0	0	3
14:55	2	2	0	0
14:56	2	2	0	0

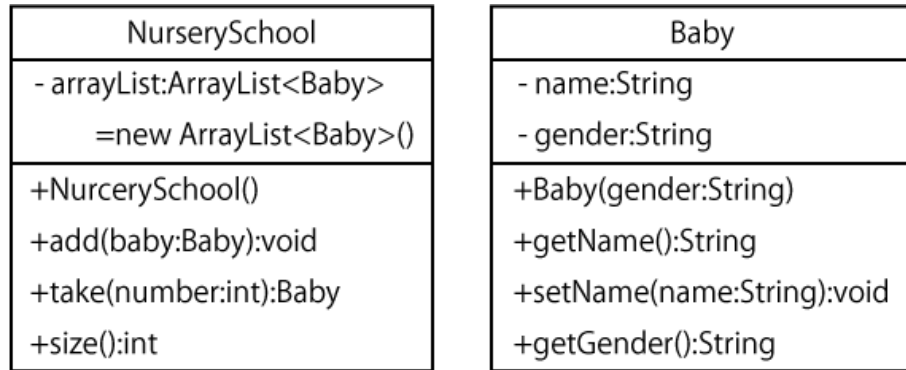


図 9 課題のクラス図

指導役が探索を行わずに指導を行うべき学生の把握が行えた。

しかし、現在のシステムの実装では指導役はコンピュータの画面を通してのみ情報を確認することができるため、情報の確認のためにコンピュータを持ち歩くかその設置場所へ向かう必要があるなど、情報の確認のための煩雑な点が見られた。

また、現在は学生情報収集システムで学生の提出結果に関しての情報のみを収集しているため、途中経過などに対する指導を行うためには、以前と同じく指導役自ら教室を巡回する必要があった。

6. まとめ

実習授業において学生探索を目的とした自律エージェントシステムの開発を行った。

本学のプログラミング実習授業に特化したシステムの開発を行い、システムの試験導入によって自律エージェントシステムによる学生の探索が可能であることを確認した。これにより指導役は自らが学生の探索を行わなくとも、指導を行うべき学生の特定が可能となる。しかし、試験導入により指導役の利用するクライアントプログラムが限定的であることや、学生の途中経過の確認はこれまでと変わらず学生間を巡回する必要があるなど、いくつかの煩雑な点が認められた。

今後は、学生情報収集システムとして学生の課題に対する途中経過を収集するサブシステムの開発を行う。これにより、学生の提出結果や途中経過に応じたよりの確な学生の探索が

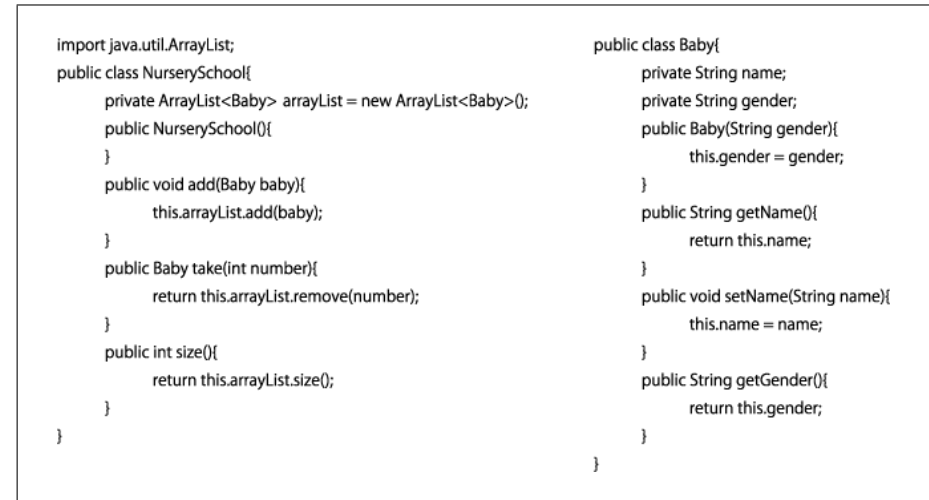


図 10 課題の解答

可能となることが見込まれる。また、エージェント同士の協調動作や成長性などに着目し研究を進めることや、システムの導入によるダイナミックな授業改善に取り組んでゆく。

参考文献

- 1) 奥井善也, 原田史子, 高田秀志, 島川博光: 講義中の反応に基づく説明方法と教材の改善, 情報処理学会誌, Vol.50, No.1, pp.361-371(2009)
- 2) 金西計英, 妻鳥貴彦, 矢野米雄: LOGEMON:Web 教材を使用した授業での教師支援システム, 電子情報数新学会論文誌, Vol.J83-D-1, No.6, pp.658-670(2000)
- 3) 瀬下仁志, 田中明通, 丸山美奈, 鈴木英夫, 高橋時市郎: 学習者主導の学習活動における活動プロセスの可視化・分析, 日本教育工学会論文誌 29(3), pp.359-369,(2006)
- 4) 水谷晃三, 池本悟, 山口大輔, 永井正武: マルチエージェントによるシームレス Web 型学習システムの試作, 電子情報通信学会論文誌, Vol.J90-D, No.3, pp.887-896(2007)
- 5) 白田由香利: 個人進度に合わせた双方向機能をもつマルチメディア教育支援システムのモデル化, 電子情報通信学会技術研究報告. DE, データ工学 101(343), pp.1-8(2001)
- 6) 長谷川伸, 松田承一, 高野辰之, 宮川治: リアルタイム課題提出システムの開発, 日本教育工学会研究報告集 09(5), pp.295-298(2009)