

# プログラミング授業のための Eclipse Che を用いた 学習支援システムの開発および検証

北 大地<sup>†</sup> 井上 潮<sup>‡</sup>

<sup>†</sup> 東京電機大学大学院 工学研究科 〒120-8551 東京都足立区千住旭町 5 番

<sup>‡</sup> 東京電機大学 工学部 〒120-8551 東京都足立区千住旭町 5 番

E-mail: <sup>†</sup>21kmc04@ms.dendai.ac.jp, <sup>‡</sup>inoueu@mail.dendai.ac.jp

**あらまし** 本研究は、大学におけるプログラミング授業において、プログラミングを苦手とする学生や理解度が十分でない学生の学習を支援することを目的とする。具体的には、Eclipse Che を用いたプログラミング環境において、学生の課題への理解度を向上させるための情報を提供するシステムを提案する。本研究では、このシステムを開発し、実際の授業に導入することで、学生のプログラミングに対する理解度がどのように変化するかについて検証を行った。その結果、プログラミングが苦手、あるいは、理解度が十分でないと思われる学生について一定の理解度向上がみられることが明らかとなった。

**キーワード** プログラミング授業, Eclipse Che, 学習支援システム, Java

## 1. はじめに

情報化やグローバル化による社会の変化を背景として、各教育機関においてプログラミング教育の実施・拡充が行われている。日本においては、平成 29・30・31 年の学習指導要領改訂によりプログラミング教育の拡充が図られた[1]。この改訂を受け、小学校ではプログラミング教育の実施、中学校・高等学校ではプログラミング教育の拡充が順次進められている。

このような背景のもと、我々は、「新たに学生の授業内容および課題への理解度を向上させるための情報を提供するシステムを開発し、プログラミング学習を支援すること」を目的として「プログラミング授業のための Eclipse Che を用いた学習支援システムの検討」を行った[2]。本検討においては、将来的な幅広い教育機関におけるプログラミング教育での支援を目指し、学習支援システムの検討を行った。具体的には、大学におけるプログラミング授業を対象とし、実際に行われているプログラミング授業を受講した学生を対象に行った事前調査によって得られたデータから、「課題実施前に行うクイズ形式の復習を行う機能」「課題実施中に発生するエラー文表現の易化を行う機能」「学習を通じて得られる称号を表示する機能」の 3 つの機能を備えた学習支援システムの提案を行った[2]。

この検討を踏まえ、我々は学習支援システムの開発を行うとともに、実際のプログラミング授業において導入することで検証を行った。本稿では、この学習支援システムの開発および検証の詳細について述べる。

本稿の構成は次の通りである。第 2 章では、関連研究を紹介し、提案手法の方向性を述べる。第 3 章では、前提条件として学習支援システムの構成要素および導入を行ったプログラミング授業について説明する。第

4 章では、開発を行った学習支援システムの詳細について説明する。第 5 章では、学習支援システムを実際の授業に導入した評価実験の内容を述べ、第 6 章では評価実験から得たデータの分析結果について述べる。第 7 章では、本稿全体のまとめと今後の課題について述べる。

## 2. 関連研究

プログラミング授業のための学習支援システムに関する研究は多く行われている。本章では、ダウンロードを必要とする学習支援システムに関する研究と Eclipse Che を用いた学習支援システムに関する研究に着目し、関連研究と本稿との違いについて述べていく。

### 2.1. ダウンロードを必要とする学習支援システムに関する研究

土田ら[3]は、GUI プログラミング授業を対象としたフィードバック支援システムを開発した。このシステムは、ユーザアプリと管理者アプリの 2 つで構成される。このうち、ユーザアプリは学生が取り組んだ課題の採点結果と時系列のログ情報の取得を行う。一方、管理者アプリでは、取得したログ情報の前処理および分析を行う。このシステムにより、採点者は学生の取り組み状況を時系列に沿って把握することができ、システム導入以前と比較してより質の高いフィードバックを行うことが可能となった。しかし、システムのうちユーザアプリは、学生の所有する PC にダウンロードして使用するものであり、使用環境によっては上手く動作しないなどの問題も発生した。また、時系列のログ情報は学生がファイルを保存するタイミングで取得されており、保存回数が少ない場合、十分な量のログ情報を得ることができない場合もあった。

本稿では、学習支援システムをクラウド型統合開発環境(IDE)である Eclipse Che 上に構築することで、利用者の使用環境に依存する点が少ないという点で先行研究と異なっている。

## 2.2. Eclipse Che を用いた学習支援システムに関する研究

杉野ら[4]は、Eclipse Che と Docker を用いたクラウド IDE によるプログラミング演習環境の構築を行った。この研究では、高等教育におけるプログラミング教育にゲームプレイを通じた学習効果を得ようとする教育手法の 1 つであるエデュテイメントのアプローチを用いて、その学習効果について調査を行った。調査にあたり、Eclipse Che と Docker を用いたクラウド IDE によるプログラミング演習環境に、Git による利用者の変更履歴の記録と収集ができるしくみを加えた環境を構築し、実際の授業に導入した。実験を通じて、構築した環境をプログラミング演習環境として提供するには可用性の点で不十分であったものの、発生した問題に対する改善策を提示するとともに、実装できなかったエデュテイメント要素の提案を行い、プログラミング授業におけるクラウド IDE によるプログラミング演習環境の有用性を示した。

Ardimento ら[5]は、Eclipse Che の拡張機能として学生プロファイリングツールを開発した。この研究では、学生がプログラミング課題に取り組む中で発生しやすいエラー（構文違反）を分析し、報告することを目的とした学生プロファイリング手法を提案するとともに、この手法を用いてコーディング活動中の学生に有用なフィードバックを提供することを目的としている。開発した拡張機能を備えたプログラミング演習環境を実際の授業に導入し、実験を行ったところ、提案した手法が学生のコーディング能力を大幅に向上させることが明らかとなった。

これらの関連研究では、Eclipse Che を用いて学習支援システム構築しているという点では本研究と同一である。一方で、利用者である学生のプログラミングに対する理解度向上を目的とし、学生のプログラミングの取り組み方をもとに、複数の機能を一つのシステムとして提供することを目指している点で先行研究と異なっている。

## 3. 前提条件

本章では、学習支援システムの導入を行った授業の詳細と、学習支援システムに利用した Eclipse Che について説明する。

### 3.1. 授業の内容と形態

本研究では、東京電機大学工学部情報通信工学科の 2 年次の学部生を対象とした授業「インターネットプ

ログラミング」のプログラミング課題を対象とする。この授業は Java の基本的なプログラミング技術の習得を目的として全 14 回実施され、授業内容に応じた演習課題が毎回出題される。なお、この授業を受講する学生は 1 年次に必修科目として C 言語の基本的なプログラミング技術の習得を目的とした授業を受講しており、一定のプログラミング技術を習得している。

### 3.2. Kubernetes

Kubernetes とは、コンテナのオーケストレーションツールの一種である。Kubernetes の特徴として負荷分散と高可用性があげられる。Kubernetes はオーケストレーションツールの名称の通り複数のマシンを組み合わせることを前提に開発されている。複数のマシンが協調して作業を行うためには、ある特定のマシンへの負荷の集中を避ける必要がある。そこで、ノード間で負荷を分散させることでマシンの協調性を高めている。また、複数のマシンを使用する都合上、あるマシンに不具合が発生した場合、他のマシンへ影響を及ぼす恐れがある。そのため、事前に正常な動作状況を定義し、問題が発生した場合はその状態へと自動で修正を行うことでシステム全体への影響を最小限に抑えることができる。

### 3.3. Eclipse Che

Eclipse Che とは、ウェブブラウザ上で動作するクラウド型 IDE の一種であり、既存の IDE である Eclipse と同等の機能を有する。現在、Eclipse Che は Kubernetes または OpenShift 上にインストールすることで実行することができる。

Eclipse Che をプログラミング学習支援ツールとして利用することの利点は大きく分けて 2 つある。

1 つ目は、Kubernetes の特徴である「負荷分散」「高可用性」を活かすことができる点である。プログラミング授業においてプログラムを実行するためには学生個人ごとのワークスペースが必要となる。Eclipse Che では、個々のコンテナにこのワークスペースを当てはめることで独立したプログラミング実行環境を構築することができる。プログラミング授業においてはこのコンテナを複数運用することになるため、「負荷分散」や「高可用性」によって快適な学習環境を提供することができる。また、一度構築した環境を移植することが比較的容易であることから、授業規模に応じたマシンの増設にも対応しやすいという利点もある。

2 つ目は、Eclipse Che に拡張機能を搭載することができる点である。Eclipse Che における拡張機能を利用するためには、互換性のある一部の VSCode 拡張機能を利用する方法と標準の IDE である Eclipse Theia の拡張機能として TypeScript 言語を用いて開発する方法の 2 種類がある。このため、各教育現場において求めら

れる拡張機能が異なっても既存の拡張機能を導入する,または,拡張機能を開発することで実現できる.

### 3.4. 定義ファイル

Kubernetes は, 定義ファイル (マニフェストファイル) において登録された内容に従ってコンテナとボリュームがセットになった Pod を作成する. Kubernetes で用いられる定義ファイルは YAML 形式または JSON 形式で記述される.

Eclipse Che においては YAML 形式の定義ファイルである devfile が用いられており, この devfile をワークスペースに読み込むことでプログラミング実行環境を構築できる. なお, この devfile はあらかじめ用意されたものだけでなく, 自身で作成したものも利用できるため, プログラミング実行環境を自分好みにカスタマイズすることもできる.

### 3.5. Eclipse Theia

Eclipse Theia とは, オープンソースの IDE フレームワークであり, Eclipse Che においてはデフォルトの IDE として採用されている. Eclipse Theia は VSCode をもとに TypeScript 言語を用いて実装されていることから, VSCode の拡張機能に関する一部の API と互換性がある. このため, Eclipse Theia の拡張機能を開発する際は既存の VSCode 拡張機能で用いられている技術を活用することで, 開発コストの低減を図ることができる.

## 4. 提案システム

本章では, 提案する学習支援システムの全体像, ユーザーから見える外部設計, および, システムの実装にかかわる内部設計について説明する.

### 4.1. 提案手法

第1章で述べたように, 我々は「プログラミング授業のための Eclipse Che を用いた学習支援システムの検討」[2]において行った事前調査で得られたデータを踏まえ, 学生がプログラミング課題に取り組む工程および各工程で必要となる技能を定義した. この定義に従って各工程において学生の学習を支援する機能を考案した. 加えて, 学生のプログラミング作成課程を記録するため, 管理者向けの「ログ収集機能 (機能 4)」を作成した. 以上の工程・技能・提案手法の関係性を表 1 に示す.

これらの機能を Eclipse Che 上に拡張機能として組み込むことで一つの支援システムとして提供することを目的とする.

表 4.1 本学習支援システムの提案手法

	工程 1	工程 2	工程 3
項目	問題文の整理	プログラムとして書き起こし	プログラムの動作
技能	・計算論的思考	・情報を整理 ・プログラムとして書き起こし	・コンパイルエラーの修正 ・実行時エラーの修正
提案手法	機能 1 クイズ形式の復習を行う機能		機能 2 エラー文表現の易化機能
	機能 3 称号獲得機能		
	機能 4 ログ収集機能		

#### 4.1.1. クイズ形式の復習を行う機能 (機能 1)

この機能は, 計算論的思考およびプログラムの作成を支援することを目的としており, 授業内容の振り返りを行うとともに, 課題に取り組む上で必要な情報を提供する. 具体的には, 講義内容について用語やプログラムの処理などを問う設問と, 提示された課題に取り組む上で必要な数学的知識やアルゴリズムなどを問う設問を想定している. 本システムにおいては, Microsoft Forms のクイズ機能を利用して, 授業ごとに課題に対応したクイズ作成し, 学生に提供する. このクイズには, 各項目に正誤に応じた得点が与えられ, この得点を 3 つ目の機能である称号の一部に反映させる. なお, 学生は期間内であればクイズに何回でも解答することができ, 最終解答をその生徒の得点とする.

#### 4.1.2. エラー文表現の易化機能 (機能 2)

この機能は, Java においてコンパイルエラーやプログラム実行時に発生するエラーのうち, 頻出するエラーについて, 日本語での説明を行うとともに, 一般的な解決方法を提示することで, エラーの解消を支援することを目的とする.

#### 4.1.3. 称号獲得機能 (機能 3)

この機能は, 学習内容に対応した称号を機能 1 の得点および各課題の得点や取り組みに応じて与えるものである. この機能自体は, 学生の理解度向上に直接寄与するものではないものの, ゲーミフィケーションの手法を活用した学生のモチベーション向上を通じて, プログラミングへの理解を深めることを目的としている.

#### 4.1.4. ログ収集機能 (機能 4)

4 つ目は, 学生のプログラミング作成課程を記録する機能である (以下, 機能 4). この機能は, 学生のプ

プログラミングの作成過程を記録することで、管理者(採点者)が学生の学習状況を把握できるようになるとともに適切な助言・指導を行うことを支援することを目的とする。

## 4.2. 提案システムの設計

まず、提案手法に基づいて学習支援システムを実現する方法について考える。提案手法で取り上げた4つの機能を搭載した学習支援システムを開発する方法は、大きく分けて2つ存在する。

1つ目は、2.1節「ダウンロードを必要とする学習支援システムに関する研究」で取り上げたシステムのように、必要な機能を搭載したアプリケーションを構築し、学生にインストールさせる方法である。この方法は、システム開発の自由度が高く、各プログラミング授業の内容に合わせてシステムを構成しやすい特徴がある。しかし、先に述べたようにインストール型のアプリケーションはPC環境に依存する場合も多く、十分なデバッグを行うことが難しい場合、問題が発生する可能性が高くなる。

2つ目は、サーバー上に学習支援システムを構築し、学生に提供する方法である。この方法は、PC環境への依存度が低く、学生への必要なプログラミング環境および情報の提供と管理者の情報収集を円滑に行うことができるという利点がある。一方で、従来このような学習支援システムを構築しようとする場合、学習支援システムそのものの開発に加えてサーバーへのアクセスやログイン、ユーザー管理など様々な要素を考慮して開発しなければならず、開発コストが高かった。

しかし、Eclipse Cheはクラウド型IDEとしてこのようなユーザー管理やログインに関する機能があらかじめ搭載されているため、開発コストを抑えることができる。また、サーバーの増強等を行う場合もKubernetesを利用することによってEclipse Che自体の移行やサーバー間の通信構築を比較的容易に行うことができる。

以上のことを踏まえ、本研究においてはKubernetesおよびEclipse Cheを利用し、サーバー上に学習支援システムを実現することとした。

## 4.3. 学習支援システムの概要

Eclipse Cheはクラウド型統合開発環境である。本研究においては、図4.1に示すようにサーバー上にKubernetes環境を構築し、その内部でEclipse Cheを動作させている。そのため、利用者である学生は各個人に割り当てられたコンテナ内のワークスペースにアクセスすることで学習支援システムを利用することができる。

なお、本研究において構成した環境では標準のテキストエディタとしてEclipse Theiaが採用されている。また、授業内で用いるプログラミング言語であるJava

の実行環境を構成している。したがって、学生はワークスペース上でプログラムの作成および実行を行うことができる。

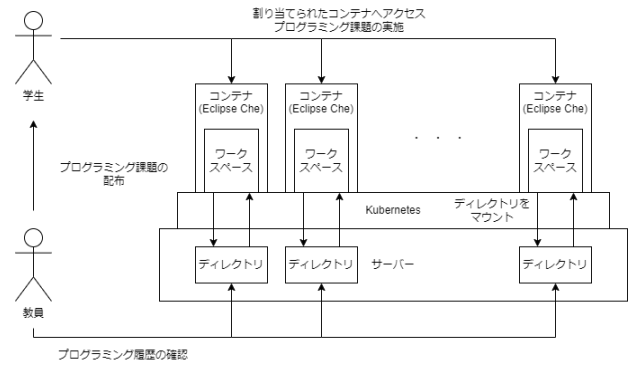


図 4.1 システム概要

## 4.4. 提案システムの実装

### 4.4.1. ハードウェアおよびソフトウェア

今回の実装において使用したハードウェアおよびソフトウェアを表4.2に示す。サーバーについては、マシンを2台用意しKubernetesの機能を用いて協調させることにより、受講者数への対応を行った。また、Eclipse Cheの初期設定としてCPUおよびメモリ使用量にリクエスト(最低限割り振られるリソース)が設定されていたことから、マシンの負荷軽減を目的として各リクエスト値を0として再設定を行った。

表 4.2 使用ハードウェアおよびソフトウェア一覧

	サーバー1 (マスターノード)	サーバー2 (ワーカーノード)
OS	Ubuntu 20.04.4 LTS	
CPU	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-9700 CPU @ 3.00GHz
メモリ	64GB	
Kubernetes	MicroK8s v1.23.5	
Eclipse Che	Eclipse Che 7.45.0	

### 4.4.2. ユーザー管理

Eclipse Cheを個人のサーバー上で運用する場合、ユーザーがワークスペースを利用するためにはOAuth2.0を用いた認証を行わなければならない。今回の実装においては、Microsoftの提供するAzure Active Directoryを使用し、OAuth 2.0認証を行った。これは、本学において全学生にMicrosoft IDが付与されており、Azure Active Directoryを使用することで、Microsoft IDプラットフォームによる個人の識別およびユーザー管理を行うことができるためである。

これにより、学生がMicrosoft IDを用いてログインを行うことで、学生と紐づいたコンテナ(Eclipse Che)が自動的に作成される。

### 4.4.3. プログラミング実行環境

今回の実装においては、対象となる授業において使用する Java 言語のみが搭載されるように設定を行った devfile を作成し、Github を通じて学生に配布を行った。学生は、Eclipse Che にログイン後、指定のフォームに Github の URL を入力することでワークスペースを作成することができる（図 4.2）。

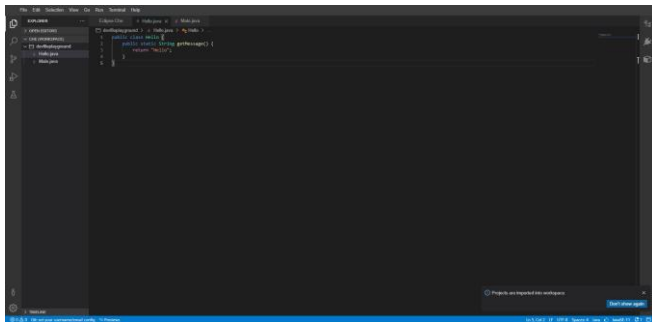


図 4.2 ワークスペース実装イメージ

### 4.4.4. 拡張機能

今回の実装においては、機能 1, 3, 4 について開発を行い、最終的に機能 1, 4 について実装に至った。この内、機能 1 の実装イメージを図 4.3 に、機能 3 の開発イメージを図 4.4 に示す。なお、機能 1 については、クイズ内容を文章で表現することや、正誤判定を行い点数を算出することなど複雑な処理が多く含まれていたことから、Eclipse Che の拡張機能ではなく Microsoft Forms を利用した形式に変更を行った。また、機能 4 については、VSIX パッケージとして Github を通じて取り込む形式とした。

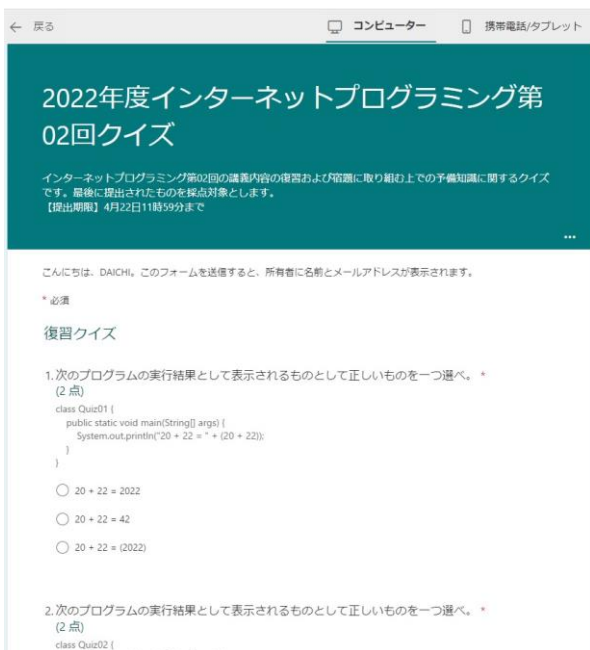


図 4.3 機能 1 実装イメージ



図 4.4 機能 3 開発イメージ

## 5. 評価

本章では、提案した学習支援システムを用いた実験とその評価、考察について述べる。

### 5.1. 実験概要

提案した学習支援システムを対象となる授業に導入し運用を行った。これによって得られた各実験データを分析することで、システムの評価を行う。ここでは、次の 3 点の分析および評価を行う。

1. 学習支援システムの有無と学生の成績との関係
2. プログラム編集履歴を利用した理解度の低い学生の抽出可能性
3. 学習支援システムに関する学生への利便性調査

5.3 節では、1 の学習支援システムの有無が学生に与える影響を検証し、本システムの有効性を確認する。次に 5.4 節では、2 のログ収集機能で得られたプログラム編集履歴に関するデータと学生の成績との関係から成績が伸び悩む可能性のある学生を抽出可能か検証する。最後に、5.5 節で学習支援システムの利便性について学生にアンケートを実施し、改善点を調査する。

### 5.2. 実験データ

本評価において使用する実験データは次の 5 種類である。

#### 5.2.1. 機能 1 におけるクイズ回答および成績

Microsoft Forms を用いた機能 1 におけるクイズは全 14 回の講義中 12 回実施された。各回のクイズは 5 問で構成され、1 問 2 点の計 10 点満点であり、提出期限内であれば何度でも提出可能であった。ただし、最終提出されたクイズの得点を成績として採用した。クイズの内容は、5 問中 3 問は授業内容の復習、2 問はプログラミング課題の内容に関するものであった。設問形式は主に選択肢、並び替え、数値入力 of の 3 つを使用した。

### 5.2.2. 機能 4 におけるプログラミングログ

学習支援システムに搭載した機能 4 によってプログラミングログを収集した。プログラミングログは、学生がワークスペースでプログラムの変更を行った際に保存される仕組みとなっている。このため、学生がプログラミング課題に取り組んだと考えられる各回の授業開始後から課題提出期限までに記録されたログを評価対象として利用した。

### 5.2.3. プログラミング課題成績

学習支援システムの有無による学生への影響を調べるため、システム導入以前の 2021 年度インターネットプログラミングとシステム導入後の 2022 年度インターネットプログラミングにおける各プログラミング課題成績を利用することとした。これは、両年度ともプログラミング課題の出題内容は一部を除いて同一であることから、比較を行うことができると判断したためである。ただし、2021 年度についてはプログラミング課題のみ 10 点満点の点数が成績となっている一方、2022 年度についてはプログラミング課題を 8 点満点とし、先述のクイズ得点を 0 点は 0 点、2 点から 8 点は 1 点、10 点は 2 点に変換した上で、合計し、最終的に 10 点満点として点数を成績としている。

### 5.2.4. 期末試験成績

学習支援システムの有無による学生への影響を調べるため、システム導入以前の 2021 年度インターネットプログラミングとシステム導入後の 2022 年度インターネットプログラミングにおける期末試験を利用することとした。各年度の期末試験の内容自体は異なるものの、対象授業の内容がどの程度定着しているのかを判断するという目的が同一であることから、これらの成績を利用することとした。

### 5.2.5. アンケート調査

学習支援システムの利便性について調査するため、アンケートを用いた調査を行った。対象授業の期末試験終了後、2 週間の回答期間を設け、37 名から回答を得た。

## 5.3. 学習支援システムの有無と学生の成績との関係の評価

### 5.3.1. 評価方法

提案を行った学習支援システムの有無と学生の成績との関係について評価を行う。学生の成績の比較は、システム導入前の 2021 年度対象授業とシステム導入後の 2022 年度対象授業におけるプログラミング課題成績および期末試験成績を用いて行う。

プログラミング課題成績は、プログラミング課題に取り組んだ上で、成績評価が行われた学生の成績を第 2 回から第 13 回の授業全 12 回の各授業回について平

均したものを用いて比較を行う。また、参考として 2022 年度については、クイズの成績を含めないプログラミング課題の成績(素点)についても比較を行った。なお、各年度によって素点の満点が異なることから、平均点の換算を行う。

期末試験成績は、期末試験を受験し、成績評価が行われた学生の成績について各年度の平均点で比較を行う。なお、各年度によって期末試験の満点が異なることから平均点の換算を行う。

### 5.3.2. 評価結果

プログラミング課題成績の比較結果を表 5.1 に、期末試験成績の比較結果を表 5.2 に示す。なお、表 5.1 における得点の換算は 2022 年度の素点に 1.25 をかけることによって行った。また、表 5.2 における得点の換算は 2022 年度の平均点に 0.7 をかけることによって行った。

表 5.1 各年度の授業回別プログラミング課題成績平均点

授業回	第 02 回	第 03 回	第 04 回	第 05 回	第 06 回	第 07 回
2021 年度	8.3	9.0	8.1	6.9	7.3	8.5
2022 年度	8.6	8.9	8.1	7.9	8.1	9.0
2022 年度換算後	8.8	9.0	8.3	8.0	7.9	9.0

授業回	第 08 回	第 09 回	第 10 回	第 11 回	第 12 回	第 13 回
2021 年度	7.8	7.5	8.6	6.2	5.8	5.7
2022 年度	8.3	8.2	8.6	6.7	8.4	8.2
2022 年度換算後	8.1	8.3	8.9	6.9	8.6	8.4

表 5.2 各年度期末試験の受験者数・満点・平均点一覧

実施年度	2021 年度	2022 年度
受験者数	110 名	106 名
満点	70 点	100 点
平均点	32.8 点	52.1 点
換算後平均点	32.8 点	36.5 点

### 5.3.3. 考察

表 5.1 より、平均点が同じものを除くと、平均点が 2021 年度のほうが高いものが 1 回、2022 年度のほうが高いものが 9 回あることがわかる。以上から、学習支援システムを導入すると学生の成績が向上する傾向にあると言える。

一方、表 5.2 について、各年度の学生の成績を標本とし、2 標本について有効水準を 0.05 として t 検定を行ったところ P 値は 0.051、効果量は 0.27 となった。この結果は各年度の期末試験の成績の間に差があるとはいえないことを示している。したがって、学習支援システムの導入が学生の成績に影響を与えたとは必ずしも言えないことがわかる。



## 5.4. プログラム編集履歴を利用した理解度の低い学生の抽出可能性の評価

### 5.4.1. 評価方法

提案を行った学習システムのうち、機能4のプログラム編集履歴を利用した理解度の低い学生の抽出可能性について評価を行う。

履修登録者112名のうちEclipse Cheを利用した105名から収集することができた。分析においては、収集したログのうち、各学生がそれぞれのプログラミング課題に取り組む間にプログラムの変更を行った回数（プログラムの編集履歴）を利用した。なお、学生の成績との関係を調査するため、取り組んだ各プログラミング課題に対してプログラミングの編集履歴が1回以上存在し、かつ、プログラミング課題に取り組んだ記録が10回以上残っている学生を分析の対象とした。その結果、対象者は35名となった。

分析にはまず、各プログラミング課題について各学生が行った変更回数を算出する。次に算出した変更回数を降順に並び替え、10等分にする。そして、上から順に10, 9, ..., 2, 1の階級を付与し、最終的に学生が各階級に何回当てはまったのかと期末試験成績との相関関係について調査を行った。なお、階級の割り当てに際して各階級に等しい人数が割り当てられない場合は、以下の式(1)を用いて値を算出し、値の小数第一位を四捨五入した値と等しい階級に割り当てることで可能な限り均等になるように調整を行った。

$$\frac{10}{x \times y} \quad \dots(1)$$

$x$ : 降順に並び替えを行い下から数えた際の順位

$y$ : その回の対象者数

### 5.4.2. 評価結果

期末試験得点と編集回数との相関係数を表5.3に示す。

表5.3 期末試験得点と編集回数との相関係数

	階級	1	2	3	4	5
成績	全体	0.10	-0.21	<b>-0.32</b>	-0.20	-0.03
	平均以下	-0.06	0.01	-0.15	0.09	-0.17
	平均以上	0.14	-0.12	<b>-0.43</b>	0.12	-0.10
	階級	6	7	8	9	10
成績	全体	0.14	0.21	0.09	0.16	0.08
	平均以下	0.12	0.14	0.26	-0.00	<b>-0.31</b>
	平均以上	0.12	-0.01	-0.30	0.25	0.07

### 5.4.3. 考察

表5.3より、期末試験の得点と編集回数が階級3に当てはまる回数との間に弱い負の相関があることがわかる。また、期末試験の得点が平均点以上の場合、編

集回数が階級3に当てはまる回数との間に負の相関が、期末試験の得点が平均点以下の場合、編集回数が階級10に当てはまる回数との間に弱い負の相関があることがわかる。

よって、階級3と階級10に当てはまる回数が多い学生については、期末試験において成績が伸び悩む傾向にあることから、支援が必要であると考えられる。

## 5.5. 学習支援システムの利便性の評価

### 5.5.1. 評価方法

提案を行った学習支援システムについて、学生に対しアンケート調査を行い、利便性について評価した。

### 5.5.2. 評価結果

実施したアンケート調査のうち、「Eclipse Cheを利用して感じたこと」に対する回答を抜粋したものを表5.4に、「クイズに回答して感じたこと」に対する回答を抜粋したものを表5.5に示す。

表5.4 「Eclipse Cheを利用して感じたこと」回答（抜粋）

回答内容
多数の人が利用しているとアクセスが集中しすぎて利用できないことがあるため、もう少しサーバーを強化した方が良いと思う
利用までの作業が面倒に感じた
面倒な行程が多い。(cookieの削除,ドラッグアンドドロップ等)
(前略)「利用手順書」には様々なパターンの問題解決方法について記載されていたため、(中略)悩むことなく対処することができた。

表5.5 「クイズに回答して感じたこと」回答（抜粋）

回答内容
その回の内容について、問題を通すことで理解が深まったと思う
その日に行った講義の内容において要点がまとまっているように感じ、知識としての定着は円滑に行うことができたように感じる。
良かった点はテストの対策にもなったし、プログラミングのコードや知識の勉強になったこと。
クイズを提出するフォーマットとは別に期間関係なく何回でもできる仕様のものがあればいいなと思った。

### 5.5.3. 考察

表5.4より、Eclipse Cheを利用して感じたこととして、欠点が目立つことがわかる。特に、アクセス集中時の利用のしづらさや利用に際して必要な作業が多い点など、著者がEclipse Cheを実際の授業中に使用した際にも感じた点であり、改善すべき点であると感じる。ただ、コメントとして欠点をあげた学生のほとんどがEclipse Che以外のプログラミング環境を主に使用していた。これは、初期の授業回で特に接続者数が多い場面でEclipse Cheを利用し、上記のような欠点を感じた

ことからプログラミング環境を変更したためであると考えられる。

表 5.5 より、機能 1 のクイズ機能を利用して感じたこととして、利点が目立つことがわかる。特に、授業内容の理解および定着に役立ったと回答した学生が多く、機能 1 としての役割を一定程度果たすことができたと考えられる。

## 6. まとめ

本研究では、大学におけるプログラミング授業においてプログラミングへの理解が十分でないためプログラミングを苦手と感じている学生や成績の伸び悩んでいる学生を対象とした学習支援手法を提案した。具体的には、学生向けとしてプログラミング学習における予習・授業・復習という学習サイクルの定着を目的としたクイズ形式の復習を行う機能、エラー文表現の易化機能、称号獲得機能の 3 つ、および、管理者向けのログ収集機能の計 4 つの機能をクラウド型 IDE である Eclipse Che の拡張機能として搭載したシステムである。このシステムにより、学生に対してはプログラミング学習に役立つ情報を、管理者向けには学生の学習状況に関する情報を提供する。

まず、機能 1 および機能 4 を搭載した学習支援システムについて、システムの有無が学生に与える影響を各プログラミング課題成績と期末試験成績の 2 つの観点から検証した。その結果、プログラミング課題については、システムを導入することによって成績が向上した。一方、期末試験については、システムの導入有無が成績向上に必ずしも寄与しない事が判明した。このことから、現状のシステムでは各授業回に限った知識・技術の短期的な定着には寄与するものの、長期的なプログラミング技能の向上には十分寄与することができないことを示した。

次に、ログ収集機能によって収集した各学生の学習履歴を用いて、成績の伸び悩みが懸念される学生が抽出可能であるか検証を行った。その結果、期末試験成績が平均点以下であった学生の特徴を抽出することができた。このことから、精度は高くないものの、成績が伸び悩む可能性のある学生を抽出することが可能であることを示した。

一方、得られた結果から、今後の課題として 4 つの課題があげられる。

1 つ目は完全な形での提案システム実現である。今回は提案システムのうち一部の機能の実現にとどまった。ただし、機能 1 の利用によって短期的ではあるものの、各授業回の知識・技術の定着が見られたことから、機能 2, 3 を含む完全な形で提案システムを実現し、プログラミング授業内で活用することにより、目標を

達成できると考える。

2 つ目は利用プログラミング環境としての Eclipse Che の改善である。今回のシステム運用を通じて、Eclipse Che をプログラミング環境として利用する際の問題点がいくつか明らかになった。特に、利用障壁ともなる煩雑な操作やアクセス集中による動作遅延など、プログラミングを嫌厭する原因になりかねない事象の解決は喫緊の課題である。

3 つ目は機能 1 の問題作成方法の改善である。今回、機能 1 で用いたクイズの設問は筆者が対象授業の指導教員の助言を受けながら人手で作成したものである。人手で設問を作成する利点として授業内容や学生の学習状況を反映させやすい点が挙げられる。一方で、クイズ作成にかかる負担は大きく、仮に提案システムを他の授業で導入する場合、指導教員に掛かる負担が大きくなりすぎる懸念がある。

4 つ目は機能 4 のログ収集方法の改善である。今回、機能 4 についてはプログラムに変更が生じた時点でワークスペースに存在するすべてのファイルを 1 単位としてログの収集を行っていた。この方法の場合、あるソースコードの変更履歴を閲覧しようとしてもファイルが分散しているため容易に行うことができない。そのため、各ソースコードを中心とした履歴収集方法に変更を行う必要がある。

今後はこれらの問題を解決していくことで、学生に学習サイクル定着が定着し、特にプログラミングへの理解が十分でないためプログラミングを苦手と感じている学生や成績の伸び悩んでいる学生のプログラミングに対する理解が深まることで、学生全体のプログラミング技能向上が実現できるとともに、指導教員側の負担軽減やきめ細やかな学習支援の実施につながると考える。

## 参 考 文 献

- [1] 文部科学省：「教育の情報化に関する手引」について、[https://www.mext.go.jp/a\\_menu/shotou/zyouhou/detail/mext\\_00117.html](https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/mext_00117.html), 2023-01-01 参照
- [2] 北大地, 井上潮：“プログラミング授業のための Eclipse Che を用いた学習支援システムの検討”, 第 14 回データ工学と情報マネジメントに関するフォーラム(DEIM Forum 2022), H41-3
- [3] 土田浩輝, 北大地, 井上潮：“GUI プログラミング授業のためのフィードバック支援システムの検討”, 情報処理学会第 83 回全国大会, 2ZG-05
- [4] 杉野雄大, 新村正明, 岡野浩三, 小形真平：“Eclipse Che と Docker を用いたクラウド IDE によるプログラミング演習環境の構築”, 信学技報(IEICE Technical Report), KBSE2019-57(2020-03)
- [5] Pasquale Ardimento, Mario Luca Bernardi, and Marta Cimitile: “Software Analytics to Support Students in Object-Oriented Programming Tasks: An Empirical Study”, IEEE, DOI:10.1109/ACCESS.2020.3010172, 2020.