

オンライン大学のプログラミング演習科目における受講状況と成績に関する考察

後藤 幸功¹

1. はじめに

既存の通学制大学では、プログラミング演習の科目は校舎に通学しコンピュータ教室で対面による指導が一般的である。この授業形態は通信制大学でも同様で、プログラミング演習は学生が演習可能な教室に集い、同一環境において、教員が確認しながら演習を行う形式である。しかし、サイバー大学では、学生は一切通学する必要がないこととしているため、プログラミング演習は従来の大学と異なり、リアルタイムに学生の学習状況を把握できず、また演習環境も学生のコンピュータに依存する環境である。このような状況の中で、学生の演習授業に対する、出席状況や演習の進捗状況、成績状況を調べることにより、遠隔かつオンデマンド環境である完全インターネット大学におけるプログラミング演習の課題と今後の可能性について考察する。

学生の出席状況については、学生がいつ授業を視聴しているかについて調査し、演習の進捗状況については課題の提出状況を演習状況と定義して調査した。これらの状況と学生成績の関連性について調査を行った。

2. 遠隔環境におけるプログラム演習

サイバー大学はインターネットを介してオンデマンドに授業を受講できる大学である。授業はビデオを用いた Video on Demand のコンテンツと、テキストと問題を解いて進める Web Based Training のコンテンツを用いて行う。そして、その受講環境はモバイル環境ではスマートフォンのアプリケーションを使用し、PC ではウェブブラウザを使用して提供する。そのため、学生が使用する PC の OS は自由である。

筆者が担当するプログラミング関連の演習授業では、UNIX を用いたサーバを用意し、学生は自分の PC と大学の UNIX サーバを用いて UNIX コンソール上の各種コマンド操作と Web アプリケーションのプログラミングを演習する。演習を行うために、学生は自分の PC にエディタ(TeraPad¹⁾)や端末接続用ソフトウェア(TeraTerm²⁾)、ファイル転送ソフト

¹ サイバー大学 IT 総合学部・教授

ウェア(Cyberduck³⁾)をインストールする。そして、学生は自分の PC でプログラムを記述する。記述したプログラムを大学が提供する演習用のサーバにアップロードし、さらにサーバ側で Web アプリケーションの設定を行う。そののち、プログラムの動作を確認しながら演習を行う。

学生はサーバにプログラムをアップロードしたのち、サーバ上でプログラムの動作確認を行うため、エラーが発生するとプログラミング経験の少ない学生はそのエラーがプログラムの記述に起因するエラーであるか、サーバ側の設定が起因するエラーであるかを判断することができず、エラー対応が分からない場合がある。学生のエラー対応に関する質問を受け付けるために、授業運営において質問のための掲示板を準備するとともに教員と TA へのメールで質問できる環境を提供している。

3. プログラミング演習科目の運用上の課題

この節では、オンデマンドの遠隔授業におけるプログラミング演習科目の授業運営上の課題について述べる。

3.1. 演習環境の構築

筆者が担当する演習授業では、学生にエディタ、ファイル転送ソフトウェア、UNIX サーバへ接続するための TeraTerm をインストールさせる。この段階において、ソフトウェアを初めてインストールする学生や、PC の操作、ブラウザを用いてファイルをダウンロードする操作が分からない学生もいる。そのため、何人かの学生はこの最初の段階で演習を放棄してしまうものもいると考えられる。実際にこの段階において、質問の多くはブラウザでファイルをダウンロードしたがその先の操作の画面が授業で示している内容と異なるために混乱してわからないというものであった。

このように演習環境を学生個人の PC で構築することは遠隔授業においては難しいことであるといえる。そのため、この解決策としては以下の2つがあげられる。

案1. 学生の演習環境を大学が準備し、リモートデスクトップで提供する

案2. 受講条件として、PC の操作やソフトウェアの基本操作を習得していることを試験によって把握し、受講生を絞り込む

である。

案1は、学生にとっても教員にとっても授業の進行がしやすく、また、教員も学生の進

オンライン大学のプログラミング演習科目における受講状況と成績の関係に関する考察

捗状況を把握しやすい点が利点である。しかし、大学側の設備負担が大きい点が問題である。通信制の学校では、学生数も多く、また同時利用者数も把握しにくいいため、仮想デスクトップ環境の台数を見積もることが難しい。

案2は、受講前に学生の知識および操作スキルが統一されているため、授業内において操作に関するスキル差が生じない点が利点である。しかし、操作能力を遠隔環境でテストすることが難しい点が欠点である。

案1と案2の折衷案として、学生は最低限ウェブブラウザを操作できることを前提として、できるだけプログラムをインストールする必要がないように、ブラウザ上で動くファイル転送および端末エミュレータ、エディタのツールを大学が提供できるようにすることがよいと考察する。

3.2. エラー時の対応

この説で述べる「エラー」とは学生が作成したプログラムを動作させたときにおこるプログラミングミスによるエラーを示す。

エラーが発生すると、プログラム経験の少ない学生はその対応が分からないため、教員へ質問する必要がでてくる。この段階で、質問ができる学生と質問ができない学生にわかれる。質問ができる学生は教員へ Q&A 掲示板かメールで質問を行う。しかし、質問ができない学生は自分でインターネットなどを使用して検索するが、正しく検索できないためにエラーを解決できず、そのまま授業を放棄することがある。これまでの授業経験では、自分で調べたがわからない状態となる学生が質問をしている可能性はたいへん少ない。4学期分の質問では、自分で調べてわからなかったという理由で質問をしてきた学生は1学期中1~2人程度であった。この状況から、エラーが出たときに質問できる学生のみが受講を最後まで行う結果となっている。

授業の運用上、学生がエラーを起こしているかどうかは把握できない。この点がオンデマンドによる遠隔授業と教室で行われるリアルタイム授業の違いであり、オンデマンドによる遠隔授業では学生の状況を把握することができないため学生を助けることはできない。よって、オンデマンドによる遠隔授業では、学生の自主性と質問ができるスキルが必要であると考えられる。

4. 受講状況の推移

このような演習環境で授業を行う状況における受講生の受講状況について調査を行う。受講状況として、出席状況および演習状況（課題の提出状況）の推移を調査する。ただし、本学における出席とは、VoDもしくはWBTのコンテンツを視聴後に毎回実施される小テストの受験を完了すること、もしくはレポート課題の提出を完了したことを示す。筆者の

授業におけるレポート課題とは、プログラムを作成して実行した結果を提出することである。

ここでは、筆者が2014年度秋学期（2014年10月～2015年2月に開講）と2015年度春学期（2015年4月～8月に開講）に担当した演習授業の出席状況について調査する。担当する授業は全15回で構成され、そのうち1回、8回、12回、15回にそれぞれプログラムを作成する課題を課している。2014年度秋学期の履修者は121名、2015年度春学期の履修者は85名である。

この2期の授業における出席状況の推移を示したグラフ⁴⁾を図1に示す。

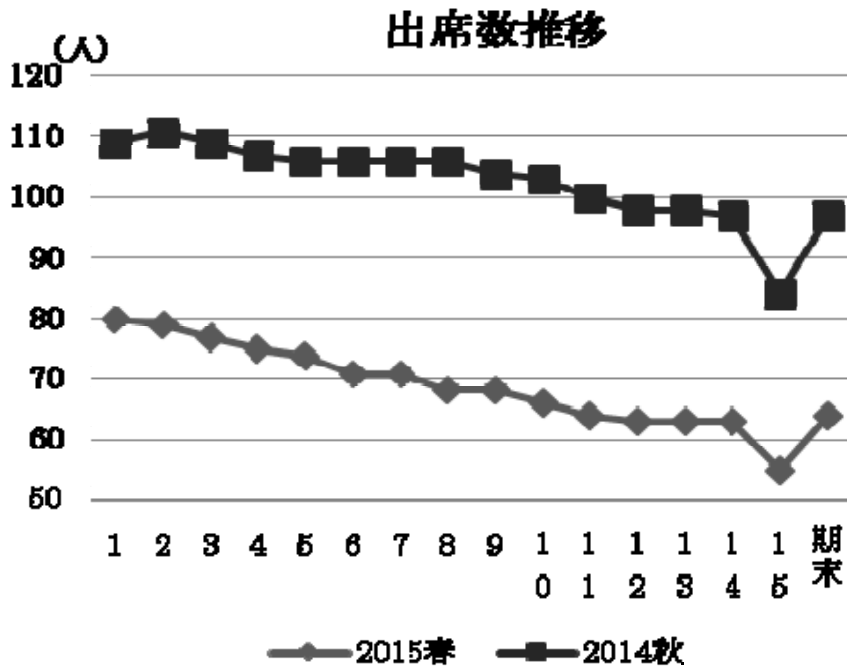


図1 2014年度秋学期と2015年度春学期出席状況の推移

図1のグラフの横軸は授業回と期末試験を示し、縦軸は出席者数を示す。このグラフから第1回から第14回の授業までは少しずつ出席者が減少しているが第15回では出席者数が大きく落ち込んでいる。これは第15回の授業を視聴していないのではなく、本学の出席ルールに従って課題を提出していないために欠席者が多い。この授業では、第1回と第15回にレポートを提出する運用である。第1回のレポートは遅刻提出をしても提出しており、またその課題も演習環境を構築できたことを報告するレポートとなっているため、提出者が多い。また、第1回目の課題については、未提出者に対して早い段階から教員やTAがメールや電話連絡を行い、提出を促している。その成果もあり提出者が多いと考えられる。しかし、第15回の課題は作成したプログラムの提出課題であり、また、その動作確認までを示すレポートである。このような課題では、提出者が著しく低くなるのがこの

グラフから見て取れる。

5. 受講時間および受講曜日別受講生の分布

学生が受講している時刻と受講曜日の変化について調査した。

プログラミング演習では、学生がプログラミングや UNIX の操作を行う演習中に疑問や問題が発生したとき、できるだけ早く教員は学生からの質問に対応できることで、演習がスムーズに進むことが期待できる。プログラミングや操作に関する演習では、一般的に受講生自身で対応できないエラーや操作ミスが発生した場合、そこで学習が止まってしまう。この状態でメールや掲示板を用いて質問を行ったとき、即時に教員や TA が対応できれば、学習がスムーズに進むことが期待できる。もしも、教員からの返信が 10 時間以上後であった場合、学生の学習意欲も減ると考えられる。

そこで、受講生が主にどの時間帯に学習しているかを把握することにより、学生からの質問を受けられやすいように効率の良い教員の指導待ち時間を決定することができると考えた。この理由により、学生の学習時刻と曜日を 2016 年度春学期のデータをもとに調査した。学習時刻別の受講生数の分布を図 2 に示し、曜日別の受講生数の分布を図 3 に示す。

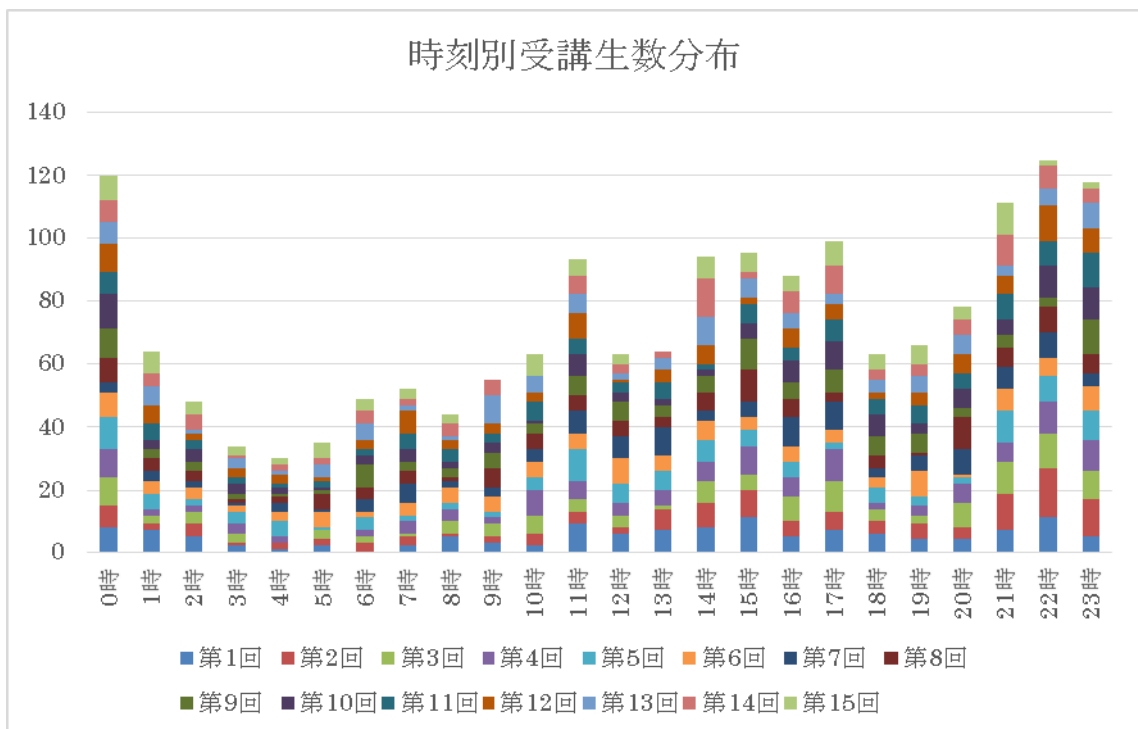


図 2 時刻別受講生数の分布

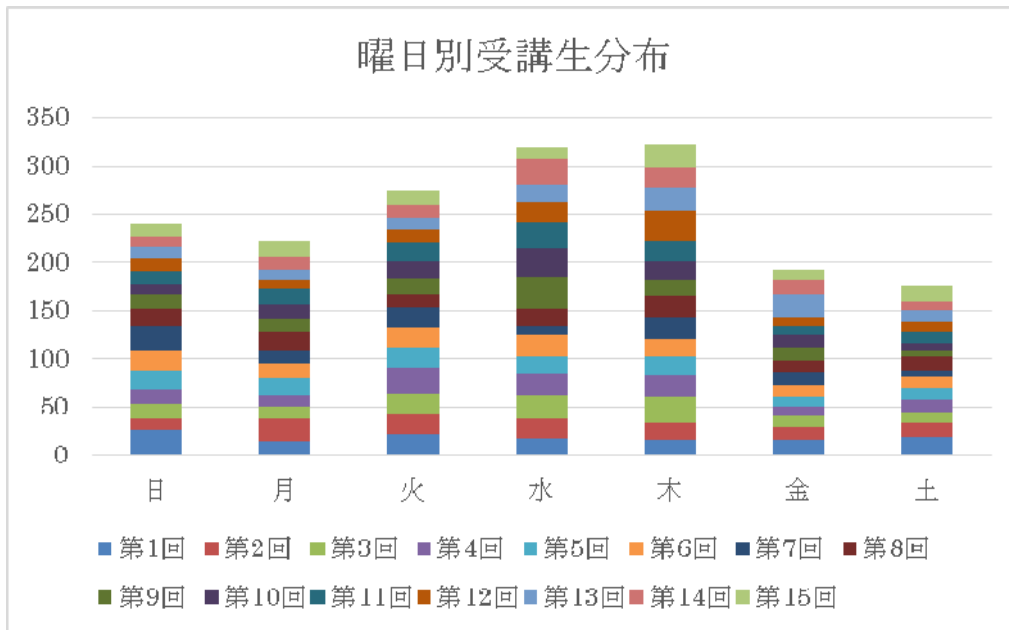


図3 曜日別受講生の分布

図2では、各回の出席者の受講時刻別の累積を示している。2016年度春学期では85名の受講生がおり、各回の受講時間別受講生の分布は多少異なるものの、ほぼ図2のような分布である。図2から見えるように、受講生の多い時間は14時台から17時台と、21時台から0時台である。特に21時台から0時台の受講生数は他の時間と比較して多い。社会人学生の多い本学では、職場からの帰宅後の時間帯に学習する受講パターンに当てはまると推測できる。11時台だけ前後の時間帯と比較して多い理由として、授業の出席認定期間が昼の12時までであるために、駆け込みで受講をしている学生が少なくないと推測できる。演習形態の授業では、スマートフォンやタブレットなどを用いて隙間時間に受講しても、PCを用いて演習を行う必要があり、PCを用いた演習後に課題に着手するために、帰宅後の時間帯である21時以降に学習をしていると考えられる。

図3では水曜日と木曜日の受講生数が多い。日曜日から木曜日までに受講生数は増え、金曜日と土曜日の受講生は他の曜日よりも少ない。週末の一般的に勤務日ではない土曜日と日曜日に学生は受講していると想定していたが、学生が受講するのは日曜日であることが分かった。一方、水曜日と木曜日の受講生数が多い理由としては、木曜日の昼12時が出席認定期間の終わりであるため、駆け込み受講が多いと推測できる。

6. 単位取得率

この節では各学期の単位取得率を調査し、受講生の成績について考察する。ここでは、2014年度秋学期から2016年度春学期までの成績分布を表でまとめたものを表1に示す。

表1 学期ごとの成績分布

成績	2014年度秋	2015年度春	2015年度秋	2016年度春
A	60	35	53	39
B	13	11	6	5
C	7	11	15	6
D	9	3	6	4
F	32	25	43	31
受講生数	121	85	122	85
合格率	73.6%	70.6%	65.6%	63.5%

表1の成績のAからDまでは合格であり、Fは不合格を示す。Aは評点が90点以上、Bは評点が80点以上90点未満、Cは評点が70点以上80点未満、Dは評点が60点以上70点未満、Fは評点が60点未満である。

この表では、成績がFの学生が約30%と多く見えるが、Fの学生のほとんどは第1回目から一度も受講していない学生である。そのため、第15回目の最後まで受講した学生のほとんどは成績がAからDの合格者である。また、この表からは成績がAの学生が多い。これは最後まで受講することで、成績Aが取れる授業であると推測できる。

成績がBからDの学生は全15回の授業のうち全15回を受講していない学生であった。これは全回を受講していないため、各回の課題を行わないために成績が低くなる。未受講生については、メールによる連絡をして受講を促進しているが、未受講学生はメールを読んでいないために連絡が付かないことが多いことが分かっている。これは筆者である教員が個別に電話をして連絡を付けたときに確認したところ、電話がつながるまで一度もメールを読んでいないことが分かった。2014年度秋学期、2015年度春学期までは未受講生に開講から6週間にうちに6回電話をして受講を促していたが、2015年度秋学期からは6週間のうちに2回と電話の回数を減らしたため、連絡が付かない学生数が増えた。この結果表1からも見えるように、成績Fの学生が増えていることが分かる。

7. 考察とまとめ

本稿では、サイバー大学のサーバと個人の PC を用いたプログラミング演習授業の受講状況と成績について調査し、考察を行った。

PC を用いて実際に作業を行う演習科目では、受講生の学習する曜日や時刻は日曜日から木曜日にむけて受講生が増えており、時刻は 21 時から 0 時台までが多かった。このことから受講生は出席認定期間の最終日に向けて駆け込み受講が多いことが分かった。また、学習時間についても、隙間時間を使用した学習時間の確保をするのではなく、帰宅後に PC に向かってまとめて受講していることが分かった。演習科目では、実務を伴うために座学の講義科目と異なり PC に向かってまとめた時間が必要であることが調査により分かった。

一方、教員は学生の学習意欲を維持するために、多くの学生が学習している 21 時から 0 時台にメールや掲示板による質問を受け付けられ、即時に回答できる学習環境を提供することが望ましいことが分かった。

成績については、全回を受講している学生は成績がよく、成績差がつく原因は未受講回の課題を行わないためであることが分かった。また、年々、一度も受講しない学生が増えてきていることも分かった。一度も受講しない学生に対しては今後の課題として、大学から連絡をつける方法を検討する必要がある。

注および参考文献

- 1) TeraPad 窓の杜ライブラリ, <http://www.forest.impress.co.jp/library/software/terapad/>, 2016 年 1 月
- 2) TeraTerm プロジェクト日本語トップページ, <https://osdn.jp/projects/ttssh2/>, 2016 年 1 月
- 3) Cyberduck, <https://cyberduck.io/index.ja.html?l=ja>, 2016 年 1 月
- 4) 大学のプログラミング演習科目における受講状況と成績の関係に関する考察, 後藤幸功, 中谷祐介, 2015 年度情報処理学会全国大会, 2016 年 3 月