

《研究資料》

「ペアプログラミングによる学生関与の促進」 翻訳\*

**Enhancing Student Engagement through Pair Programming (Japanese Translation)**

内 田 君 子\*\*  
UCHIDA Kimiko

訳者序文

本稿は、コールマン博士 (Sonya Coleman) とニコルス氏 (Eric Nichols) が、2010年8月24日から26日にダーラム大学で開催された情報とコンピュータサイエンス研究センター第11回年次大会 (11<sup>th</sup> Annual Conference of the Subject Centre for Information and Computer Sciences) において発表した Enhancing Student Engagement through Pair Programming<sup>1)</sup> の翻訳である。

第一著者のソニア・コールマン博士は、アルスター大学マギーキャンパス SCIS (School of Computing and Intelligent Systems) の講師である。イメージ処理やコンピュータビジョン、医療画像処理、ロボット工学を研究領域とし、これまで日本においても日立と共同研究などを行っている。そして現在、博士は「ペア学習と評価による単位不認定の改善」の研究プロジェクトに着手している。一方、エリック・ニコルス氏は、同大学の博士課程の学生でナノエレクトロニクスプロジェクトチームの一員であり、感覚性融像に関する研究を主としているがプログラム教育についても興味を持っておりコールマン博士の指導のもと研究を行っている。

本論文に関心を向けたのは、筆者らが現在ペアによる協同学習に着目し、大学の情報基礎教育を対象としたペア学習効果に関する実践研究を行っており、共通の思索と研究成果を多く含んでいるからである。事実、大学初年次のコンピュータ教育にペア学習法を適用し、ペアでの取り組みが実習授業への関与と動機付けを高め、最終的に成績の向上につながることで、学生がペアでの学びと作業を楽しんだことを示唆するなど、本論文が明らかにした内容を支持する調査結果を筆者らも報告している。したがって、筆者らがこれまでに得た知見を吟味する上で、本論文は非常に意味深い資料として位置づけられるものと思われ、その内容について翻訳するに値すると判断した。

要約

ペアプログラミング<sup>2)</sup> は、生産性の向上とプログラミングエラーを減少させるため、産業界でしばしば利用される手法である。ペアで作業する場合、通常プログラム作成に長時間を要するが、でき上がる製品は単独作業の場合よりも結果として概ねより良い品質となる。今回、学生にやる気を与え、出席率を高め、結果として、授業への関与を向上させることを目的として、このコンセプトを大学1年次後期のプログラミングモジュールに導入することによって、教室の中でのペアプログラミングへと拡張することをこの論文により報告する。

\* 2011年9月15日受理

\*\* 名古屋学芸大学短期大学部

本論では、その方法や、実習への参加が増加し、実習評価点が前年度と比較して上昇したことなど、本年度に得られた主要な結果について概述する。

キーワード：ペアプログラミング、1年次モジュール、関与、出席

## 1. はじめに

ペアプログラミングの考え方は、近年米国で、企業ならびに大学の双方へ導入されてきている。産業界では、2人（あるいはそれより多く）のチームで働くことが生産性を高めることは事実として認識されており、ソフトウェア開発者は他のパートナーとともに働くのに作業時間の50%を費やしている。通常は開発時間こそ15%多くかかるものの、ペアプログラミングはデザイン品質、弱点の減少、専門技術の向上、そしてチームコミュニケーションの改善を含め、多くの状況を改善している。これらはすべて、我々の学生にとっても必要なスキルである。

産業と学術双方でペアプログラミングの有効性について多くの研究が行われ、発表されてきた。たとえば、産業界において Arisholm 他（3）は、ヨーロッパで初級、中級、上級の Java コンサルタントの組み合わせを用いたペアプログラミングの統制実験を行った。その結果、初級開発者が最も多くの利益を得たことが示された。彼らは、パートナーの支援によって、より少ない時間で標準よりもうまく仕事を完了することができた。（5）の研究では、適切な組み合わせによってペアプログラミングが非常に効果的であることが明らかとなった<sup>3)</sup>。

教育界においてもペアプログラミングが導入されているが、主にイギリスではなくむしろアメリカにおいてである。Braught 他（4）の研究は、学生がペアプログラミングを用いるとよりうまくコースを終了できる傾向にあり、また個々のプログラマとしても強化されることを明らかにした<sup>4)</sup>。（6）（7）はまた、ペアプログラミングの導入が学生の関与と自信を向上させ（特に女子学生や比較的成績が劣る学生において）、その結果プログラミングスキルが高まることを報告した<sup>5)</sup>。これにより、さらに多くの学生が、プログラミングモジュールを無事修了し、その結果として初年度の単位不認定の改善につながるようになる<sup>6)</sup>。

学生の単位不認定の改善と深い学習<sup>7)</sup>は、英国全土における継続的な課題である。本論では、モジュールにペアプログラミングが導入されることにより、アルゴリズムプログラミングⅡの授業において学生の関与が向上した初期の兆候について明らかにする。学生群の能力差のため授業改善の推進は単純でないことは確かだが、私たちはこのモジュールと他のモジュールの2つのグループにおける整合性を確立するための統制基準として、このモジュールの結果を2年以上にわたり分析している。2009年のモジュール出席に相当する今日までの出席記録を比較、対比することによって、ペアでの取り組みが学生の実習授業への出席を促すことを明らかにする。また、全体的に2009年に得られた実習評価成績よりも上昇し<sup>8)</sup>、結果として関与と動機付けを高め、最終的にモジュール結果の改善につながることを示す。

## 2. 方法

これまでのアルゴリズムプログラミングⅡモジュールでは、実習授業と実習試験への出席が悪かった。多くの場合、これらの学生は教科の学習課題に失敗し、結果としてプログラミングモジュールにおける高い不合格率が示されてきたのである。ペアプログラミングを導入することによって、すなわち、その中で学生たちはペアで活動しペアで評価されることによってモジュールにより多く関与し、互いに学びあい、モジュールの実習や評価に関する場面に参加しようとする事への動機

づけが高まると思われる。同様にプログラミングモジュールの合格率を上げ、より深い学びを支援し、モジュールの終了までに知識豊富なより優れたプログラマを育成すると思われる。

ペアプログラミングは、次の方法で実施された。実習と実習評価を行うために、学生はペアで作業を行った。モジュールの過程で、ランダムに選択したパートナーが多くの問題を指導できるよう、学生は自分のパートナーを選ぶことができた。

学生はそのペアで実習を最後まで行い、学期中の指定された時期にそれぞれのペアごとに実習評価が行われた。しかし、それでも評価は次のように個別で行われた。

- a) 個別成績を得るため、双方の学生は試験時に出席していなければならなかった。
- b) 双方の学生は実習について個別に質問されたため、学生は自分自身がどの程度理解しているかを示さなければならなかった。

この方法で評価を行うことによって、学生はパートナーの成果評定の影響を受けることはなかった。

### 3. 結果

モジュールの最終時に、ペアプログラミングの成果について評価が行われた。この最初の評価は、二つに分かれる。まず、科目の実習試験成績と実習授業への出席の全体的改善を評価するため、詳細な統計分析が昨年度と比較することによって行われた。さらに、学生のペアプログラミングに関する意見の収集と分析のため、アンケート調査が行われた。本研究の授業は、Ⅱ期の Semester で行われたため、1年生はすでにⅠ期 Semester でペアプログラミングを用いずに1つのプログラミングモジュールを終えている。そのため、彼らは双方の学習スタイルの利点と欠点を比較し対比することができると思われる。

#### 3.1 統計的分析

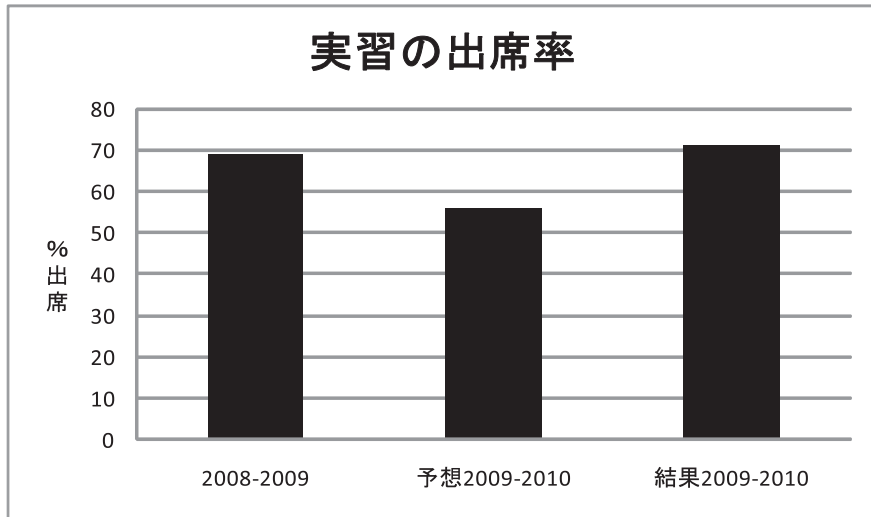
統計分析を詳細に行うため、まず出席が改善されたかを確認する目的で出席数を2008～2009年度と同じアルゴリズムプログラミングⅡのモジュールと比較した。

加えて、出席についての改善が特定群に依存していなかったことを確認するため、出席の概要を得る目的で2009～2010年度Ⅰ期の数学モジュールの出席も分析した。

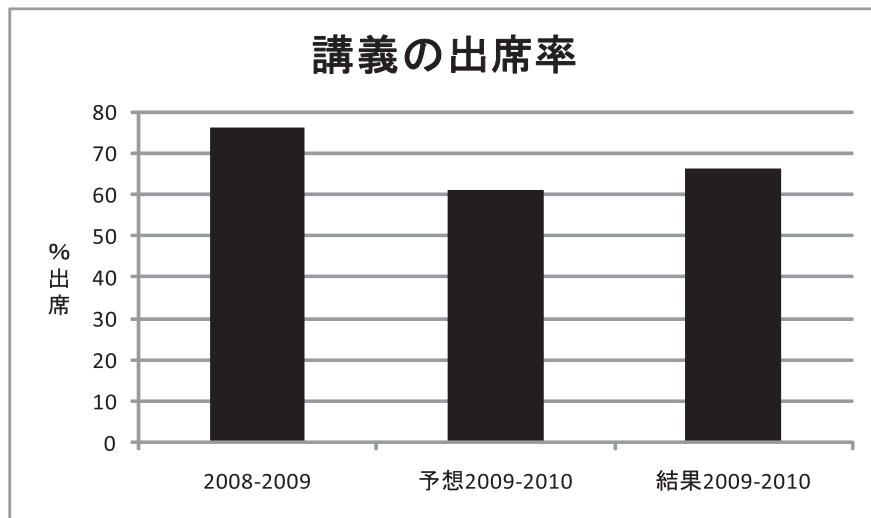
目的とする結果を得るため、学生数×週数をⅠ期モジュールの出席総数とした。次に、その学期の毎週の出席を合計し、その値を出席総数で割り出席率を算出した。これは今年度と昨年度のモジュールについて行われた。これらⅠ期の出席記録は、現在のアルゴリズムプログラミングⅡのモジュールに期待される出席率を算出するため、前年度のアルゴリズムプログラミングⅡモジュールの出席記録と比較された。同じ方法が、実習成績の結果を予想するためにも用いられた。この方法を用いる利点は、差が数量化されることと、異なる年に同じ学生の出席や試験、クラステストに適用できることである。そしてこの方法で、学生の能力の違いではなくペア授業による結果の違いのみが残り、ペア手法を適用しなかった学生の改善あるいは低下の量が見出されることになる。

まず、前年度との比較における出席の変化を分析したところ、実習の出席で2.5%の増加が認められた。さらに、同じ群に対してⅡ期データとⅠ期データの関連を調査すると、予想より出席が顕著に高いことがわかった。実際、図1に示すように全体的な実習への出席が予想より27.8%高く、講義への出席は10.35%高くなった。これは、出席率全体としてはまだ著しい改善の余地はあるが、ペアでの取り組みがⅡ期における学生の出席を促進したことを示唆している。

次に、ペアプログラミングが評価結果を改善したかどうかを判断するため、モジュールの全体的な実習評価を分析した。その結果、実習の評価点は前年度と比較して31.2%増加し、予想より



(a)



(b)

図1 2008-2009年の出席率、予想された2009-2010年の出席率、実際の2009-2010年の出席率との関係  
(a) 実習の出席率 (b) 講義の出席率

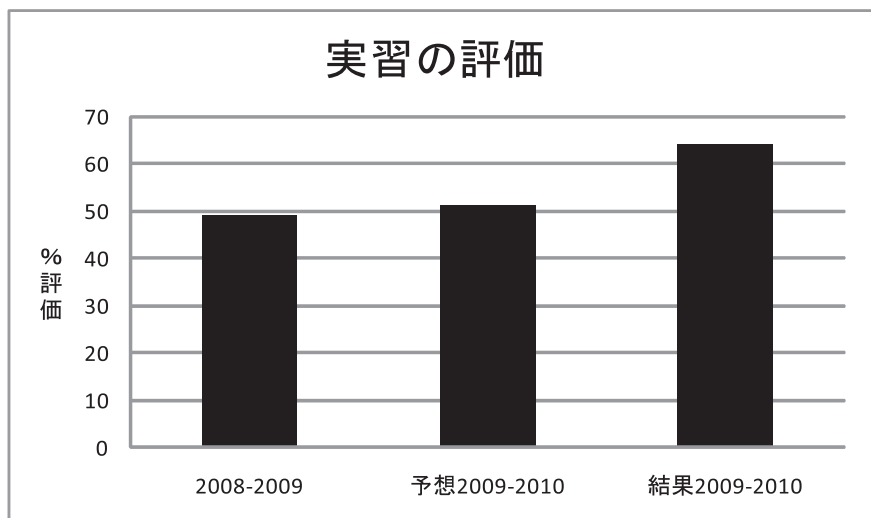


図2 実習評価点の予想と結果

26.2%高くなった。これらの結果を図2に示す。

### 3.2 学生の意見

モジュールの終わりに、ペアプログラミングについて学生の意見を得るためアンケート調査を行った。学生には、この研究計画で用いられたペアプログラミングとI期のプログラミングモジュールで使用された単独プログラミングとを比較、対比をしてもらった。これにより、ペアプログラミングに関する全体的な学生の意見を得ることができた。モジュールを通して各ペア間の大きな意見の衝突を避けるため、学生が自分のパートナーを選択できたことは特に言及すべきである。合計21名が回答を行い、そのアンケートの質問内容と各項目の学生の「はい」の割合を表1に示す。

表1. 学生意見の概要

質 問	「はい」の割合 (%)
あなたはペアで取り組むのが好きですか？	76.2
あなたは独力で取り組むのが好きですか？	28.6
あなたは大きなグループで取り組むのが好きですか？	28.6
あなたはペアでの取り組みがプログラミングの理解に役立ったと思いますか？	57.1
あなたはプログラミングのパートナーから学んだと思いますか？	57.1
あなたはプログラミングのパートナーがあなたから学んだと思いますか？	42.9
あなたとパートナーは与えられたプログラミング実習時間外に一緒に取り組みましたか？	47.6

表1に示したように、76.2%の学生は、ペアでの取り組みを楽しんでいた。これらの意見の概要から、ペアでの協力の中に知識の譲渡が生まれたことを示唆している。また、学生が大きなグループよりペアでの取り組みを好んでいることも示された。以上の結果は、評価の方法としてグループ作業を取り入れている他のモジュールに強い影響を与える可能性がある。

## 4. まとめと今後の課題

今日までの分析結果は、ペアプログラミングが学生の出席と実習評価を向上させ、さらに学生がペアでの学びと作業を楽しんだことを示唆している。ただ、今回はモジュールにおける試験の構成要素についてまだ分析が完了していないため、予備的な結果報告に限定されている<sup>9)</sup>。試験後に、前年度と比較して全体的な試験成績が改善したかどうかを確認するための追加の統計分析が行われ、ペアプログラミングはより深い学びと理解へ導くことが明らかになると思われる。しかし、この総合的な改善が果してアルゴリズムプログラミングのモジュールに特有のものか、実は学生の能力が昨年より学問的に勝っていることによるものかを明らかにするため、他のモジュール成績との相互関係をも示さなければならないと思われる。

何人かの学生の口頭による意見では、一方の学生が常にもう一方の学生の知識習得を助けていたと感じており、このペアでは双方向の協力関係がほとんど成立していなかったことを意味する。この理由としては、ペアプログラミングのためのパートナーを学生が自分で選択できたためと考えられる。そのため、2010～2011年度においては、今回同様ペアプログラミングがモジュール評価の重要部分を構成するが、この時のペアの組み合わせは、学生の学力に応じて事前に割り当てていこうと考えている。

## 参考文献

- (1) Cockburn A., Williams, L.: “The Costs and Benefits of Pair Programming” The XP Series, Extreme Programming Examined, pp. 223-243, 2001
- (2) Sanders, D.: “Student Perceptions of the Suitability of Extreme and Pair Programming” Computer Science Education, 2003
- (3) Arisholm, E., et al.: “Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise” IEEE Trans on Software Engineering, Vol. 33, No. 2, Feb 2007
- (4) Braught, G., Eby, L.M., Wahls, T.: “The Effects of Pair-Programming on Individual Programming Skill” SIGCSE, March 2008
- (5) Dyba, T., et al.: “Are Two Heads Better than One? On the Effectiveness of Pair Programming” IEEE Software, Vol. 24, No. 6, pp. 12-15, Dec 2007
- (6) McDowell, C., et al.: “Pair Programming Improves Student Retention, Confidence and Program Quality” Communications of the ACM, Vol. 49, No. 8, August 2006
- (7) Simon, B., Hanks, B.: “First-Year Students’ Impressions of Pair Programming in CS1” ACM Journal on Educational Resources in Computing, Vol. 4, No. 7, Jan 2008

## 訳者注

- 1) Sonya Coleman, Eric Nichols, “Enhancing Student Engagement through Pair Programming” Proceedings 11th Annual Conference of the SCICS, pp. 47-51, 2010  
原文は、<http://www.scribd.com/doc/54590998/Proceedings-11th-Annual-Conference> からダウンロードすることができる。
- 2) ペアプログラミングは Extreme Programming (XP) の主要な要素として注目され、産業界で普及した。XP は、コンピュータソフトウェアの開発手法の一つで、2名のプログラマチームが、アルゴリズム、コード化、テスト上で協働することが求められる。一方のプログラマ (the driver) がコードを作成し、もう一方 (the navigator) が欠陥や弱点をレビューし、両者は定期的に役割交代するというものである。そして、経験の多少に関わらずほとんどすべてのプログラマにとってペアプログラミングが有効であったとの成功例が報告されている。例えば、  
Williams L., Kessler R., Cunningham W., Jeffries R., “Strengthening the Case for Pair Programming” IEEE Software 17, pp. 19-25, 2000
- 3) 本文中で紹介された Tore Dyba 他の文献 (5) において、1998年から2008年までに行われたペアプログラミングと単独プログラミング効果を比較した研究の文献調査結果が報告されている。研究の一覧表は、[www.computer.org/software](http://www.computer.org/software) で、それぞれの実情と背景に関する追記と共に利用することができる。
- 4) Braught 他 (4) の研究は、コンピュータサイエンスを専門としない学生を対象とした点、SAT (Scholastic Assessment Test : combined verbal and math) 得点や担当教員、性別とペアプログラミング効果との関連を検討した点が特徴的である。そして、ペアプログラミングは SAT 得点の低い学生にも有効で、すべての SAT レベルで CS (Computer Science) コース合格率が向上することや、教員の違いによる影響の減少に効果があることを明らかにした。また、統計的有意差は認められなかったものの、ペアプログラミングは女性への影響力が強い (CS コース合格率が男性の3倍) ことも示した。
- 5) Charlie McDowell 他 (6) は、大学の初年次教育に着目し、ペアプログラミングが教育的な手法として有効であることを報告した。その成果は、ペアプログラミングのアプローチが、現在の大学教育が抱える諸問題を解決するための糸口を示唆するものであり、筆者らの実践研究にも大きな影響を与えた。
- 6) 日本の大学教育におけるペアプログラミング研究については、次の文献を参照されたい。  
寺川佳代子, 喜多一: プログラミング教育におけるペア学習の試みⅣ, 情報処理教育研究集会講演論文集, pp. 459-462, 2005  
寺川佳代子, 喜多一: プログラミング教育におけるペア学習の試みⅢ, 第4回情報科学技術フォーラム, pp. 345-346, 2005

寺川佳代子, 喜多一: プログラミング教育におけるペア学習の試み, 第3回情報科学技術フォーラム, pp. 299-300, 2004

7) 原文の Deep Learning は、教育学においてはしばしば Surface Approach (表面的な学習) と対比されて用いられている。すなわち、体験に基づいた問題解決学習は知識の深い理解につながるため Deep Learning と呼ばれる。一方、単なる暗記学習は、知識の表面的な結合作業にすぎない Surface Approach と呼ばれる。一般的には Deep Learning の方がより良い学習法とされている。

8) 個別プログラミング群との比較によるペアプログラミング群の実習成績上昇傾向については、Emilia Mendes 他が2004年に Software Development and Design Computer Science コースでテスト得点、最終試験得点、コース合格率の3項目について調査を実施し、得られた結果の検証実験を2005年に行うことによって、3項目の成績いずれにおいても高い得点が達成されることを証明している。

E. Mendes, et al., "A replicated experiment of pair-programming in a 2nd-year software development and design computer science course" In ITICSE'06, pp. 108-112, New York, NY, 2006

E. Mendes, et al., "Investigating pair-programming in a 2nd-year software development and design computer science course" In ITICSE'05, pp. 296-300, New York, NY, 2005

また、同様に Nagappan 他もコース合格率からペアプログラミングの効果を明らかにし、その合格率改善がCSを専門としない学生の場合は次のコースにも保持される一方で、CSを専攻する学生間には保持されないことを提起している。

N. Nagappan, et al., "Improving the CS1 experience with pair programming" SIGCSE'03, pp. 359-362, New York, NY, 2003

N. Nagappan, et al., "Pair learning: With an eye toward future success" XP/Agile Universe, pp. 185-198, 2003

9) 本論中に今後の研究課題として記載された追加の統計分析については、2011年2月に下記論文において成果が発表された。この内容については、次の機会に翻訳報告を行う予定である。

Sonya A. Coleman, Eric N., "Embedding Inquiry based learning into Programming via Paired Assessment" ITALICS Volume 10 Issue 1, pp. 72-77, 2011 (ISSN1473-7507)