

モチベーションに着目したプログラミング入門教育

土肥 紳一 宮川 治 今野 紀子

東京電機大学 情報環境学部 情報環境学科

〒270-1382 千葉県印西市武西学園台 2-1200

e-mail: {dohi, miya, konno}@sie.dendai.ac.jp

概要

少子高齢化が進む中、18歳人口の減少によって、大学へ進学する学生は多様化するようになった。新聞紙上では学力低下を懸念する記事が目をひく。e-Learningを活用した復習用の教材を自作しこれを活用する例や、1年目の教育を特に見直し退学者数を減らす試み、歯科系の大学ではキャンパス内に寮を建設しカリキュラム全体の見直しを実施している大学等が紹介されている。これらの試みで共通していると考えられることは、学生のモチベーションをいかに喚起するかという点にある。本研究は、学生のモチベーションに着目したプログラミング入門教育を実施し、授業全体を一つのシステムとしてとらえ、毎回の授業の理解度を調査し、その結果を次の授業へフィードバックする等、モチベーションの向上を目的としたものである。1セメスターを前期、中期、後期に分割し、個々の時点における学生のモチベーションを客観的に測定することによって、それを低下させている要因分析が可能になった。教育効果の客観的な測定は難しいが、受講者のモチベーションに着目した評価尺度は、教授者や教授内容に依存しない客観的な指標としてその効果を示すことができる。さらに要因分析結果から、今後の授業改善策を提案することができる。このように、授業全体を一つのシステムとしてとらえ、学生のモチベーションを向上することを目指した教育の実践は極めて少ない。教育現場における人間と情報環境を探りながらその有効性を示すと共に、このような試みが拡大することを目指す。

1. はじめに

学生のモチベーションを喚起することが重要な社会問題として取り上げられるようになった現在、情報教育を支援するために独自に開発されたシステムが数多く存在し、いくつかの場面で効果を発揮している。しかし、授業を一つのシステムとしてとらえ、時間的な変化の中で授業自体を改善する試みや、授業全体の教育効果を客観的に測定し、その評価を行った例は少ない。本研究では、教授者や教授内容に依存しない客観的な評価尺度として、学生のモチベーションをモニタリングすることに着目した。約2年の歳月をかけ、現場のプログラミング入門教育から得られたデータを基に、標準化した評価尺度を完成した[1]。本論文では、プログラミング入門教育でモチベーションに着目する必要性、評価尺度と分析方法を示すと共に教育の実践から得られた教育効果を述べる。

2. 積み上げ式教育とスパイラル教育

2.1 積み上げ式教育

積み上げ式教育は、基礎から順に断片的な知識(コンテンツピースと呼ぶこととする)を学習し、コンテンツピースを底辺から順に積み上げて行くことによって、一つの分野を修得することを狙った考え方である。これは理路整然としており、周期的に開講される授業形態と調和する。個々のコンテンツピースに着目した教育を続けることによって、高く積み上げることができる。一つのコンテンツピースが学生にとって理解が難しい場合は、コンテンツピースをさらに細分化でき、これらを再帰的に行うことによって学生の知識水準に適応した教育を実現できる。この概念を、図1に示す。

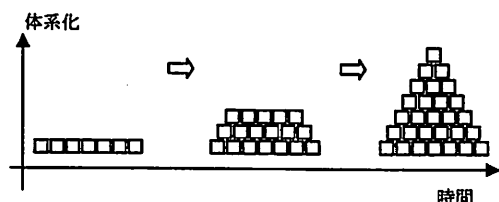


図1 積み上げ式教育 (理想)

Introduction to the Computer Programming Education Focused on Improvement of Students Motivation, S.Dohi, O.Miyakawa, N.Konno
School of Information Environment, Tokyo Denki University.

しかし、多様化した学生の様子を観察する限り、本来の積み上げ式教育が意図しているように学生は学習できていない。コンテンツピースを学習する過程で、学生はその場において理解する努力をする。新しく学んだコンテンツピースの知識は、時間の経過にともない徐々に薄れる。この過程で、新しいコンテンツピースが次の授業の中で登場し、過去に知識として修得しているはずのコンテンツピースは、ますます記憶から薄れる。このような結果、本来、基礎として積み上げられなければならない底辺を支えるコンテンツピースそのものが消失し、その上部に位置するコンテンツピースが崩壊する。この様子を、図 2 に示す。最近、教育に関するシンポジウムやフォーラム等で、「木を見て森を見ず」といった指摘が良く聞かれる。学生は、一生懸命勉学に励むものの、森を見ることなく目的を見失い、やる気をなくしてしまっている。

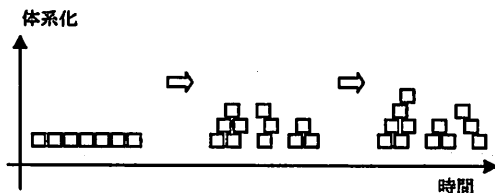


図 2 積み上げ式教育 (現実)

2.2 スパイラル教育

木だけを見せようとする積み上げ式教育では、学生のモチベーションを維持することは困難であり、授業の学習効率が上がらない。学生のモチベーションをいかに維持もしくは向上できるかが重要な課題となる。解決の糸口は、常に森を意識させながら木を見せることである。一見、無謀なことのように聞こえるが、目標を見失わないよう森があることを常に意識させ、そこにたどり着くために必要な木を徐々に見せて行くことにより、モチベーションを維持させる考え方である。最初は森全体の形状は分からないものの、徐々にその形状が明らかになり、授業の進行によって一層鮮明になって行く。この過程を毎回の授業で体験することによって、既に学習した知識の再確認を行うことができる。もし、既に学習した知識が無いと、森の輪郭は永遠に見えて来ない。したがって、森を見るためには、過去の知識を体系化して認知す

る必要がある。このことを毎回の授業で要求する流れを作ることによって、森全体が鮮明に映し出せるようになる。この考え方は、積み上げ式の教育とはまったく逆の発想であり、多様化した学生に合った考え方である。この様子を図 3 に示す。

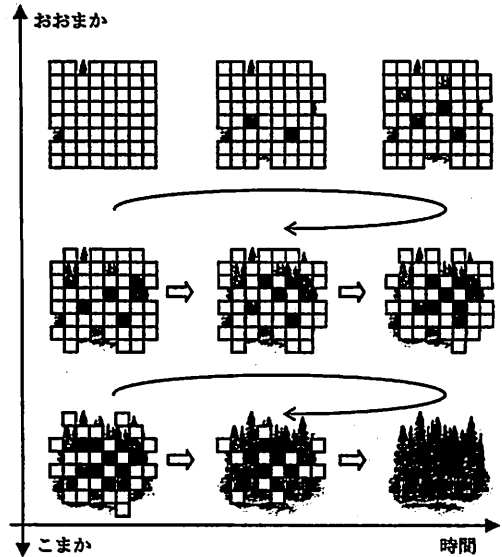


図 3 スパイラル教育

3. モチベーションに着目した授業

3.1 モチベーションの評価尺度

モチベーションに着目した授業を実施するためには、教授した内容に対して学生のモチベーションがどのように推移しているかを客観的に測定できる必要がある。一般的に、授業は教員の経験知に任されて実施されており、学生がどのようにモチベーションを維持しているかを、教授者に依存しないで客観的に測定できなければならない。これを実現するために、ARCS 理論に着目したモチベーションの評価尺度、SIEM アセスメント尺度を完成した。これを、表 1 に示す。

3.2 スパイラル学習

森の見せ方は、徐々にその形状が明らかになる方法である。何をどのように見せて行くかは積み上げ式の教育と大きく異なるため、授業全体の流れを考え直す必要がある。一例であるが、文法中心の流れは、まさに積み上げ式の典型例である。興味の湧かない内容について淡々と説明を受け、

今、必要としない詳細な説明を聞き、一体、何に活用するのか分からないまま授業が完了する。これを避けるため、文法的な説明は必要最小限に止め、必要なことを、必要なときに、必要なだけ教え、スパイラル教育の中で詳細を解き明かして行く。授業は、講義と実習を複合した形で実施する。

表 1 SIEM アセスメント尺度

F1: 授業構成因子 (因子 1)	
S1: 成功機会度	授業中にできた・わかったという実感がありますか。
S2: 親し度	授業の内容は親しみやすいですか。
S3: 愉楽度	このプログラミングの授業は楽しいと思いませんか。
S4: 理解度	このプログラミングの授業は理解しやすいですか。
S5: 知覚的喚起度	自分が入力したプログラムの動作結果を見るのは楽しいですか。
S6: 意義の明確度	授業の意義や目的がはっきりしていますか。
S7: 好奇心喚起度	授業では好奇心を刺激されますか。
F2: 自覚性因子 (因子 2)	
S8: 将来への有用度	将来に役立つと思いますか。
S9: 向上努力度	もっとプログラミングの勉強を努力しようと思いますか。
S10: 自己コントロール度	授業で学習したことを基にして、自分で工夫し勉強してみようと思いますか。
S11: 自己目標の明確度	自分の到達すべき学習の目標がはっきりしていますか。
F3: 双方向性因子 (因子 3)	
S12: コミュニケーション度	授業中、学生・教員などとのコミュニケーションはありますか。
S13: 所属集団の好意的反応度	教員やクラスのメンバーは好意的ですか。
S14: コンテンツの合致度	演習問題などは授業内容と一致していますか。
F4: 参加性因子 (因子 4)	
S15: 参加意欲度	休まずに出席しようという意欲が起る授業ですか。
S16: 参加積極度	授業での自分の参加態度は積極的ですか。
モチベーション評価項目	
S17: 重要度	プログラミングを学習することは重要だと思いますか。
S18: 現状認知度	現在の時点で、プログラミングの知識・技術は身につけていると思いますか。
S19: 期待度	もっとプログラミングの知識や技術を高めたいと思いますか。

3.3 即時フィードバック

スパイラル教育の概念を支えるためには、毎回の授業で学生の知識を与えるコンテンツピースと教授法が重要になる。どのような授業でも、受講者全員を理解させることは現実的に不可能である。経験値であるが 80%の理解度を目標としている。毎回の授業内容を学生がどの程度理解しているか

を客観的に測定し、その結果を教授者のみならず、学生も相互に共有できることが重要である。調査結果は web にて公開し、次回の授業の冒頭で結果の解説を行う。このことは学生の授業への参加意欲を高め、自分だけが取り残されているのではないかといった疑心暗鬼から開放できる。教授者は、授業内容のどの部分が理解できていないかを知ること、次回の授業に向けた方略を立てられる。

3.4 モデリング学習

授業毎に、新しい知識として認知して欲しい内容がある。プログラムは単なる文字の羅列ではなく、1行1行に意味があり、またプログラムのブロック構造を正確に把握することが必要である。このことを実践するため、授業で使用する例題は、原則として毎回、白紙の状態からプログラムを入力し、適宜、その意味を説明しながら、教員の動作に合わせて学習する。このような授業は、時間的な無駄が多く、能率が悪いと思われがちである。しかし、キー入力速度は徐々に速くなり、繰り返し体験したコンパイルエラーも、その誤りを短時間で発見でき、ブロック構造の大切さが理解できるようになる。授業の中盤以降では、初期の苦勞から開放される。プログラミングに必要な作法を体得しながら、森が少しずつ見えるようになる。

3.5 発見学習

プログラミングの授業で扱う例題は、正確にキー入力を行えば、無事にコンパイル・実行が行える。しかし、ささいな入力ミスは多くのコンパイルエラーを誘発し、初学者にとって何が起っているのかを理解することは難しい。このような失敗は、ある程度想定されるものの、想定外のエラーも沢山発生する。このような失敗は、教授者が学生と同じ目線で原因を探り、一緒に学ぶ姿勢が重要である[2]。こうすることによって、学生はどのように誤りの原因を探れば良いかという、最も本質的なことを理解できる。自分で巻き起こした失敗を自分で解決できたときの満足感を味わうことは、大変有益なことであり、大きな自信につながる。したがって、初学者に対しては、発見学習の機会を多く与えることが重要となる。このような授業は、人手に頼らざるを得ない。十分に経験を積んだTAやSAの支援が必要不可欠である。

3.6 モチベーションのモニタリング

教育の分析や改善が難しいのは、教員の経験知に任されて授業が実施されているためである。重要なことは、教員の経験知によって行われている授業が、受講者にとって適切に行われているか否かである。教授者の責務は受講者の教育であり、独りよがりではない。また教育効果の測定は、授業内容に依存したり、特定の教員に依存すると、活用の範囲は狭まる。SIEM アセスメント尺度を活用し、授業の前期、中期、後期の3回に分けて、学生のモチベーションを測定する。

3.7 授業への大域的なフィードバック

授業へのフィードバックは、大きく2つある。一つは、3.3節で述べた即時フィードバックである。もう一つは、授業を大きく3分割し、大域的にとらえたフィードバックである。後者は、授業の前期、中期、後期における受講者のモチベーションを測定し、その結果からモチベーションを低下させている要因を分析し、さらにその結果からどのような改善を行えば効果的であるかを提案し、授業へフィードバックするものである。

4. コンテンツについて

コンテンツピースは、スパイラル学習を支える重要な要素である。この内容は、各大学、各学部において最も具象化された部分である。本学部におけるコンテンツピースは、インストラクショナルデザインの考え方を活用している[3,4]。

4.1 科目間の関係

プログラミング教育は、「コンピュータプログラミングA」、「コンピュータプログラミングB」、「オブジェクト指向設計」の順に学習する流れとなっている。学年が無いと各々の科目をどの時点で履修するかは、学生に任されるが、一般的には上記の順番で履修することになる。

4.2 講義ノートの作成方針

講義ノートは、当初から紙での閲覧を考えておらず、ブラウザで閲覧することを前提に作成している。したがって文書量は可能な限り少なくし、閲覧したときの1画面の中で、学ばなければならないことが完結するよう工夫している。3.4節で述べた通り、毎回の授業はモデリング学習を基本と

しており、例題は授業毎に白紙からプログラムを入力する。したがって長いプログラムは避け、「必要なことを、必要な時に、必要なだけ教える」ことに徹した短い例題を用いる。市販の教科書等と比較すると、文法的な記載は少ない。授業では、色々なバリエーションを教授することも多いが、今学ばなければならないことが何かを明確に理解できるよう、単一の例に止めている。

4.3 スパイラル構造の実現

1回目のスパイラルでは、まずやってみることから始める。利用する例題の中には、細部に渡る説明が潜んでいるが、細かいことは必要ない。入力したプログラムを実行し、その反応を体験する。体験した結果から少し違った結果を要求する。そうするためには何を換えれば良いかを試行錯誤し、その結果が正しいこと、もしくは間違っていることを体験を通じて学習する。次の授業では、前回試したことがどのような理屈や決まりにしたがって成り立っているのかを一步踏み込んで説明し、その中に隠されていた文法や作法を習得していく。

この段階で、興味深い学生の反応があった。少し違った結果を表示するためにどうすれば良いかを考えるよう指導すると、何名かの学生は深刻に考え込んでしまった。正解を頭の中だけで得ようとしているようである。一方、思いついたことを素直に実践する学生もいる。一見、安易に試行錯誤を行っているとも見受けられるが、反応を楽しんでいる。この楽しみを体験できることが、学生のモチベーションの向上につながる。

4.4 成功の機会

一度、成功の機会を体験した学生は、その結果に大変満足する。それは、画面に文字列をただ表示するだけのプログラムであっても、自分でやり遂げたという達成感は大変大きい。この体験を重ねることが、スパイラル学習の始まりであり、森を見る一步につながる。パソコンを活用する授業は、注意しないと学生が授業から逸脱することが懸念される。このような現象が発生するのは、授業に集中できていないためである。白紙の状態からプログラムを入力させる意味は大きく、この集中力を維持するためにも効果的である。おまじないと称するソースプログラムを何度も入力することは、

無意味と考えられるが、何度も入力することによってプログラミングの作法を修得すること、そして一回のコンパイルでエラーが出ないように己と戦うことを実践する。

4.5 スモールステップ

成功の機会を多く体験させることは、重要である。しかし、成功に導くコンテンツピースの設定を誤ると、その敷居が高すぎ、成功に至らない。この様な場合は、スモールステップによってさらに細分化する。適切に細分化された問題を一つずつ解決することによって、当初の目標をクリアできる。このような成功体験が、学生のモチベーションを向上でき、次の授業の楽しみへと導く。スモールステップをどのように設定するかは、学生のレベルや興味に大きく左右される。開講当初の授業では、この辺りの兼ね合いが予測できなかったが、過去5年間の実績を反映した講義ノートの中のコンテンツピースは、毎年入学して来る新生生のレベルに合った内容に収束した。

4.6 画面の配置

ブラウザで閲覧した時、閲覧のしやすさを考慮し、左上にソースプログラム、左下に実行結果、右側に解説を掲載するスタイルを取り入れている。教授者は、左側の画面に着目し、要点を説明しながらプログラムを入力し、その説明を聞きながら学生はプログラムを入力する。口頭で説明した内容で不足するものについては、右側の説明を部分的に説明する。後日、学生が復習を行うときは、右側の説明を閲覧しながら復習する。ブラウザでの閲覧時に、1画面の中で収まるよう配置に工夫し、統一した形式や要点を強調するための配色に注意した。これらの内容は、過去の試行錯誤と、授業で活用した経験から改善したものである。

4.7 当日締め切りの課題

当日の授業が理解できているか否かを知るために、簡単な課題を出題する。授業時間内で解ける程度のものであるが、自力で解決できない場合が出てくる。教員は、TAやSAの支援を仰ぎながら、学習遅滞者の対応にあたる。この時、注意を要することは、決して答えを教えないことである。単刀直入に答えを教えることは、教授者にとってそ

の場を乗り切るには効率の良いことであるが、学習遅滞者にとって何の意味も無い。指導上の重要なことは学習者自身が考えることであり、答えに導く適切なヒントを与えられることが重要である。

5. モチベーションの測定と分析

5.1 モチベーションの測定

モチベーションの測定は、表1に示したS1~S19の項目について、授業の前期、中期、後期の3回、アンケート調査を行う。これらの項目は、授業構成因子、自発性因子、双方向性因子、参加性因子の4つの因子に分類されておりF1~F4で表現する。これらの項目はARCS理論に基づき作成したものである[5]。学生は各々の質問項目に対して5段階のリッカート尺度、「1:まったくそう思わない」、「2:あまりそう思わない」、「3:どちらともいえない」、「4:ややそう思う」、「5:強くそう思う」を回答する。学生iのモチベーションMV_iは、S17_i(重要度)とS19_i(期待度)の積として算出する[6]。したがってMV_iは最小値が1、最大値が25になる。授業のクラス全体のモチベーションMVはMV_i全体の平均として算出する。

5.2 モチベーションの分析

測定したS1_i~S19_iの結果から、4つの因子のF1_i~F4_iを算出する。F1_iはS1_i~S7_iの平均値として算出する。クラス全体のF1は、F1_i全体の平均として算出する。以下同様に、F2_i、F3_i、F4_iはS8_i~S11_i、S12_i~S14_i、S15_i~S16_iの平均値として算出した後、クラス全体のF2~F4を求め、S1~S19、F1~F4、MVの基本統計量を算出する。

6. 実践効果

測定対象とした授業は、2005年度に開講した「コンピュータプログラミングA」である。この授業はプログラミングの入門教育を目的としており、約70%の学生が初学者であった。2学科を対象に4名の教員が授業を担当している。このうち3名の教員のクラスについて測定した。3クラスの授業内容は同じであり、授業進行の同期をとりながら授業を実施した。A先生のクラスは、1学科を対象としたクラスである。一方、B先生とC先生は残りの1学科を学籍番号の奇遇で二分割し、

この2クラスはほぼ同じ母集団と考えられる。代表的なA先生とB先生の基本統計量の変化を表2に、モチベーションの推移を図4に示す。本尺度による分析は、同一内容の授業を異なる教員が担当した場合についても、教員の違いを客観的に分析することができた。さらに、モチベーションの要因分析を行うことによって、各教員に適した授業改善策を提案できることがわかった[7]。なお、紙面の都合で、各教員に示された具体的な授業改善策は省略する。

表 2 基本統計量の変化 (2005年度コンプロA)

		A先生			B先生		
		前期	中期	後期	前期	中期	後期
		9月	11月	12月	9月	11月	12月
全体	平均	17.6	17.8	15.5	21.0	21.2	20.7
	標準偏差	0.91	1.01	0.95	0.74	0.64	0.71
	中央	18	20	16	25	25	25
	最頻	25	25	16	25	25	25
	標準偏差	6.57	7.24	6.81	6.12	5.16	5.57
	分散	43.1	52.5	46.3	37.4	26.6	31.0
	尖度	-0.80	-0.56	-0.83	1.40	1.55	1.62
	歪度	-0.59	-0.66	-0.23	-1.62	-1.37	-1.32
	範囲	24	23	23	21	21	24
	最小	1	2	2	4	4	1
	最大	25	25	25	25	25	25
	合計	917	909	789	1426	1377	1264
	上位群	人数	50.0	51.0	37.3	75.0	75.4
20 ≤ MV	平均	23.1	23.8	22.6	24.0	23.7	23.9
中位群	人数	34.6	33.3	37.3	16.2	20.0	27.9
10 ≤ MV < 20	平均	14.7	14.5	14.5	15.4	15.2	14.9
下位群	人数	15.4	15.7	25.5	8.8	4.6	3.3
MV < 10	平均	6.6	5.4	6.4	5.3	6.3	2.5

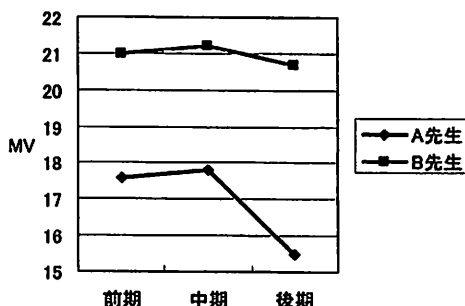


図 4 MV の変化

7. まとめ

モチベーションに着目したプログラミング入門教育は、有効に機能することが明らかになった。現在、分析の自動化を進めているが、モチベーションの要因分析結果を基に授業改善策を提案する部分は人手に頼っている。この問題はエキスパートシステムを開発すれば解決できると考えられる。しかし、人手による授業改善策の提示は、教授者に対して適切な改善策を提案できるメリットも大きい。授業全体を一つのシステムとしてとらえ、受講者の理解度を即時フィードバックし、定期的に受講者のモチベーションを測定しながら授業改善策を適用する試みは、大変成功している。スパイラル教育は、特にオブジェクト指向プログラミングに向いている[8]。今後はこのような考え方が、他の分野でも拡大していくことを期待したい。

本研究は、東京電機大学総合研究所研究Q06J-13およびハイテク・リサーチ・センタープロジェクト重点研究として行っているものである。また、本研究の実施にあたって多くの助言をいただいた、東京電機大学情報環境学部大山実教授、同工学部大井尚一教授に謝意を表する。

参考文献

- 1) 情報環境学部白書, 東京電機大学情報環境学部
- 2) 情報科教育法, 大岩元, 久野靖, 辰己丈夫他, オーム社
- 3) 実践インストラクショナルデザイン, 清水康敬, 東京電機大学出版局
- 4) はじめてのインストラクショナルデザイン, ウォルター ディック, ピアソンアンドエデュケーション
- 5) Keller, J.M., & Suzuki, K. (1988). Use of the ARCS motivation model in courseware design (Chapter 16). In D.H. Jonnasen(Ed.), Instructional designs for microcomputer courseware. Lawrence Erlbaum Associates, USA.
- 6) 上淵 寿, 動機づけ研究の最前線, 北大路書房
- 7) 土肥紳一, 宮川治, 今野紀子, SIEM アセスメント尺度による異なる教員のプログラミング入門教育の分析, 情報処理学会, No4, p377-p378 (2006.3)
- 8) 山田隆太, わかるオブジェクト指向, 技術評論社