

資格取得を考慮した専修学校でのプログラミング教育

宮本 行庸[†] 児玉 真悟[‡]

神戸情報大学院大学[†] 神戸電子専門学校[‡]

1 はじめに

プログラミング言語を教育するにあたっては、知識と実装力の双方を向上させるという目的が存在する。実装力については、演習時間を設けて課題製作により評価できる。一方、知識については定期試験により評価する方法が一般的であるが、同一レベルで重複なく問題を作成するのが難しく、また、評価がどうしても学内に留まってしまいがちである。

本稿では、外部団体による資格検定試験をプログラミング教育課程に導入し、知識レベルを測る手法を報告する。外部試験であるため、毎年一定の水準でスキルを測定することが可能となる。また、客観的な外部評価として学生本人の履歴に残すこともできる。同手法を、筆者らが担当する専修学校専門課程（以下、専門学校）IT 学科で実施し、その効用について述べる。

2 専門学校での教育課程

専門学校におけるプログラミング教育[1]では、1年次にC言語の文法等の知識に関する講義と、開発実習等の実装に関する演習の双方を実施する。講義は半期に50分×週5時間×17週を前・後期で、演習は50分×週4時間×17週を後期に設定している。

文法の講義においては、週ごとに簡単な演習課題を設け、演習時間を確保して評価物として提出させている。また、検定試験前には問題演習の時間を2週間程度設定している。これらの時間配分により、半期に講義を実施できる期間は15週間程度となる。

開発実習においては、数名程度のグループにより半期に3つの課題に取り組む。また、中間発表や最終発表を設け、プレゼンテーション力の向上も図る。優秀な作品については、各種プログラミング系コンテストへも応募している。

Programming Education of Computer College Considering the Qualifying Examination

Yukinobu MIYAMOTO[†] and Shingo KODAMA[‡]

[†]Graduate School of Information Technology,

Kobe Institute of Computing

650-0001 Kobe, JAPAN

[‡]College of Computing, Kobe Institute of Computing

650-0002, Kobe, JAPAN

miyamoto@kic.ac.jp

これらの教育課程の中に、外部団体主催の資格検定試験を導入し、文法やプログラムの解読等の知識に関する達成度を測っている。実施時期は前期が7月、後期が1月とし、それぞれの時期に習得すべき受験級を推奨している。内容としては、前期には簡単な入出力、型、変数、演算、制御文、配列まで、後期には配列以降の関数、ポインタ、構造体、ファイル操作、ライブラリ関数など、C言語の基本文法と呼ばれるものすべてを取り扱う。

成績評価については、文法事項の講義は筆記試験、提出物、資格検定試験の合格状況を総合して行う。開発実習においては、作品および発表点を総合して評価する。これらの教育課程の流れを図1に示す。

1年次 :	前期→	後期→
C言語 I	文法 (前半) 検定	—
C言語 II	—	文法 (後半) 検定
C言語 III	—	開発実習

図1 専門学校での教育課程

3 資格試験と教育手法

導入する資格検定試験については、主に専門学校生をターゲットとした民間資格[2]としている。この理由は、専修学校の教育内容をよく反映した試験内容となっていることと、費用が低コストで抑えられることである。多くの学生が受験し、合格認定を取得させるためには、学習内容及び費用の双方の面で可能な限り無理のない範囲に設定することが望ましい。当該資格検定試験は、この要求を十分に満たしている。

前期に受験する試験は、同資格の3級としている。内容は上記の前期学習内容どおり配列まででほとんどの問題が構成されている。また、後期に受験する試験は、同資格の2級を標準としており、内容は3級を含むC言語の文法事項すべてとなっている。いずれの試験も受験時期までには文法内容及び問題演習を終えるようなカリキュラム進行としており、無理のない時間を確保できている。

これらの資格検定試験を受験するにあたり、習熟度別クラス編成を活用し、クラスごとの検定対策授業を展開しているのが本手法の特徴である。クラス編成の基準としては、基本情報技術者試験の模擬試験に相当するレベルの成績を用い、3階層のクラスに再編する。以下では、各階層を上位よりそれぞれ A・B・C クラスと呼称する。

本科目担当当初である 2011 年度は、クラスに関わらず全階層で同等の授業進行としていたが、思わしい成果が得られなかったため、翌年度より階層ごとの対策を設定した。A クラスでは、検定試験直前の 2 週間において、可能な限り多くの過去問題を解く演習時間を設定し、解説を端的に行った。B クラスでは、同じ期間に実施する問題数を抑えて、その分を十分な解説の時間に回した。また、C クラスでは応用問題の演習を極力排し、基本問題の繰り返しにより合格点に達するよう目標を設定した。加えて、後期においては C クラスの学生は 3 級未合格者が大半を占めたため、A・B クラスが上記手法により 2 級対策を展開したのに対し、あえて 3 級受験を推奨し、問題演習も 3 級の過去問題を採用した。

さらに、2013 年度より本科目を複数教員で担当し、授業担当者との相性による苦手意識を極力排除するように努めた。希望者には、調整のうえ適切なクラスで受講できるような配慮も実施している。

4 結果と考察

本手法の直近数年での成果を合格者数で示す。図 2 は、2011～2014 年度の C 言語プログラミング能力認定試験の合格者の推移を級別に示している。2014 年度に関しては、原稿執筆時点での半期分のデータとなっており、確定数値は年度末となることを申し添えておく。

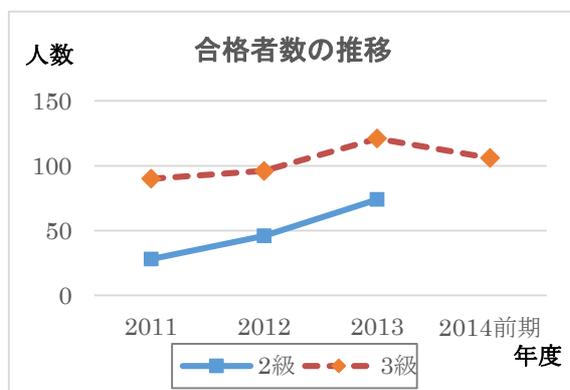


図 2 合格者数の推移

図 2 より、年度を追うごとに合格者数が増加しており、本手法の有効性が確認できる。2013 年度終了時点では、3 級合格者が在籍者の約 8 割、2 級合格者が約 5 割の比率となっている。また、2014 年度に関しては、前期での 2 級受験者はおらず、3 級受験者も 2 年度前の水準に既に達しており、後期での見込み分も合わせると前年度の水準を十分に達成できることが期待される。

年度によって学生在籍数の変動はあるが、3 級合格率については 8 割～9 割の間で落ち着く見通しである。また、2 級合格者に関してはまだ幾分かの上積みが見込める。B クラス階層を可能な限り取りこぼさずに合格に導くことができれば、7 割程度の合格率が達成できると考えられる。

本手法のキーとなるのは、中間階層である B クラスの学生の指導にある。この階層にはいわゆる合格のボーダーラインが存在すると想定でき、全体に対して画一的な指導を行っていた時と同様、B クラス内での理解度が一律でないため、正課外での個別対応が必要とされる場面が存在する。A・C クラスについては、それぞれ指導の下限・上限を設定すればよく、基準を厳密に設定する必要は少ないと考えられる。

5 おわりに

本稿では、プログラミング教育における資格検定試験の導入が、知識レベルの習得に有効であることを述べた。また、検定対策にあたっては習熟度別にクラスを編成し、階層ごとに適した対策を設定することで合格者数の向上が見込めることを示した。

今後の課題としては、現状の手法の中でさらなる合格率の向上を図る工夫を見出すことがあげられる。また、知識レベルの習得が実装力にどのように結びついているかの分析を定量的に測る尺度を検討することも課題である。

参考文献

- [1] 宮本 行庸, 児玉 真悟: 異なる学校種の情報系学科でのプログラミング教育, 情報知識学会誌, Vol. 24, No. 2, pp. 106-111, 2014.
- [2] サーティファイ情報処理能力認定委員会: C 言語プログラミング能力認定試験過去問題集, サーティファイ, 2014.