

競争型学習を取り入れた初級 C プログラミング演習における モニタリング機能の開発

Development of Monitoring Functions in Introductory C Programming Exercises Based on Competitive Learning

西村 智治, 蔵本 幸司, 富永 浩之
Tomoharu NISHIMURA, Kouji KURAMOTO, Hiroyuki TOMINAGA
香川大学工学部
Faculty of Engineering, Kagawa University
Email: s11g482@stmail.eng.kagawa-u.ac.jp

あらまし：大学情報系での入門的 C 授業に競争型学習を取り入れ、小コンテスト形式でのプログラミング演習を提案している。部分的な仕様に基づく実行テスト系列を用意し、中間目標を提示して段階的な実装を誘導する。開発した支援サーバでは、提出されたソースコードの正誤を自動判定し、進捗状況を即時に通知して、競争意欲を刺激する。教師支援として、問題の解答状況や学生の進捗状況を視覚化するモニタリング機能を実装する。また、学生の積算得点グラフから振舞いパターンを抽出し、指導に役立てる。
キーワード：初級 C 言語演習、プログラミングコンテスト、モニタリング機能、学生の振舞いパターン

1. はじめに

大学の情報系学科では、初年次の C 言語教育を重視している。しかし、理解度が異なる学生が受講しており、進捗状況や演習態度に大きな差が生じやすい。本研究室では、授業中の演習を活性化させるため、初心者が気軽に参加できる小コンテスト形式の演習を提案している(図 1)。また、Web ベースの支援システム tProgrEss を開発し、コンテスト中に教師の指導を支援するモニタリング機能も試作している⁽¹⁾。

2. コンテスト運営サーバ tProgrEss

演習における判定結果の即時通知を重視するため、事前に用意された入出力サンプルによる実行テストを用いて、プログラムの正誤判定を行う(図 2)。照合に成功し、正答と判定されれば、得点を与える。コンテストの実施形態として、教室型、宿題型、試験型のテンプレートを用意する。

コーディングの指針として、複数の予備テストと 1 つの最終テストからなる実行テスト系列を用意する(表 1)。中間目標として予備テストは、仕様を緩めて徐々に完成に近付けさせる。予備テストは、3 つ程度で構成し、プログラムの段階的実装を誘導する。

プログラムの判定結果には、時間調整点と誤答減点を導入した得点ルールを適用する。時間調整点は、提出時期により、通常の配点に加減点を付与する(図 3)。早期期間の提出は、積極性を評価し加点する。通常期間の提出には、加減点はない。延長期間の提出は、多少は許容する代わりに減点する。事後提出は、追試的な措置であり、参加点のみとする。早期と延長における加減点は、提出時期で傾斜させる。これらを設けることで、学生の早期解答を促し、演習の活性化を狙う。一方、誤答減点は、最終テストの誤答に減点を与える。最終テストでは、例外的な入力を含む、網羅的な入出力サンプルを用意する。

誤答減点により、完答への慎重な確認を行わせる。最終テストが誤答や未答で終わった場合は、予備テストの部分点のみを与え、誤答減点を行わない。

コンテスト中の実行テストの結果は、Web 上で公開される。順位表示ページでは、共時的な進捗確認として、教室全体の進捗状況を表示する。提出履歴ページでは、通時的な進捗確認として、個人の提出履歴や解答状況を表示する。

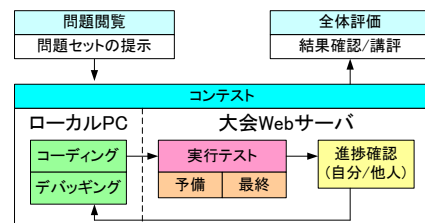


図 1 コンテスト形式の演習の進行

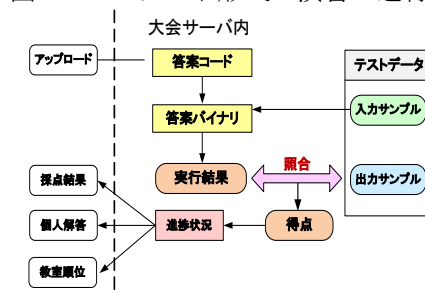


図 2 コンテスト形式の演習と tProgrEss の機能

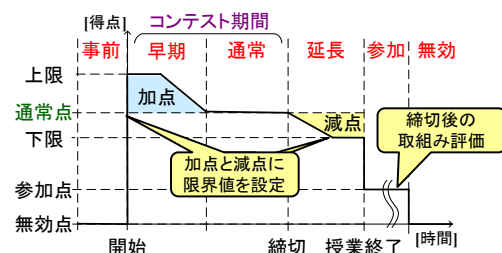


図 3 提出時期と時間調整点

表1 実行テスト系列の予備と最終のテスト

| | 予備テスト | 最終テスト |
|---------|----------------|----------------|
| 入出力サンプル | 部分的な仕様 データを明示 | 完全な仕様 データを隠蔽 |
| 正答の場合 | 解ける所まで取り組める部分点 | 解答時間を差し引いた得点 |
| 誤答の場合 | 減点なしで何度でも提出可能 | 回数による減点を慎重さを促す |

3. コンテスト中の学生の振舞い

コンテスト中の学生の振舞いについて、時系列での積算得点に着目し分析を行う。コンテスト中の学生個人の積算得点の推移を分析し、成績上位陣と下位陣での振舞いの特徴を発見する。

これまでの経験からコンテスト中の学生の振舞いは、理解度と積極性の観点により4つのパターンに分類できると考えられる(図4)。パターン1は、授業内容を理解し、コンテストへの参加も積極的な学生である。そのため、開始時から積極的に提出を行い時間調整点での加点を獲得でき、理解度も高いため多くの問題を正答し、最終的な得点は、最も高いものとなる。パターン2は、授業を理解しているが、コンテストへの参加が消極的な学生である。開始時はあまり提出せず、締切直前に提出するので時間調整点で減点されるが、理解度は高いため多くの問題を正答し、終了間際で得点が伸び、2番目に高いものとなる。パターン3は、授業の理解は不足しているが、コンテストには積極的な学生である。開始時から提出を行い、時間調整点での加点をある程度得られるが、簡単な問題しか正答できない。最終的な得点は、3番目となる。パターン4は、授業の理解が不足し、コンテストにも消極的な学生である。一般的にあまり提出を行わず締切直前になって取り組むが、簡単な問題しか正答できず、得点が伸びない。最終的な積算得点は、最も低いものとなる。

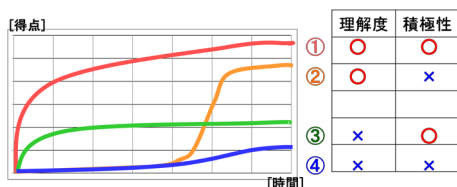


図4 学生の振舞いパターン

4. 運用実験における積算得点の分析

tProgrEss を 2010 年度の情報系学科 2 年次前期の初級 C 演習「プログラミング II」で利用し、運用評価を行った⁽²⁾。受講者は学科 2 年生 64 名である。期末試験において、試験型コンテストを実施した。制限時間は 185 分で 8 問を出題した。出題範囲は、文字列のポインタ処理、構造体と整列算法である。各問は、解答の所要時間が 30 分程度である。配点は、最終テストを 150 点とし、予備 1・2 を、それぞれ 45/105 点とした。合格基準としての目標得点を 400 点とし、完答 3 問よりも半答 7 問を目指すように指導した。なお、8 問目は、この時点で難易度の高い文字列集合の整列問題とした。

結果分析として、時系列での積算得点の分析を行

った。最終的な成績から最上位 17 名を第 1 群、中上位 17 名を第 2 群、中下位 19 名を第 3 群、最下位 11 名を第 4 群とした。開始から 5 分ごとの個人単位の積算得点グラフを集団別にしたものを図 5 に示す。

第 1 群は、60 分程度で合格点に達しており、こちらの想定 1 問 30 分よりも早いペースである。第 2 群は、80 分程度で合格点に達しており、想定通りのペースである。第 3 群は、120 分程度で合格点に達しており、想定よりもやや遅いペースである。第 4 群は、得点に変化がない期間が他の集団に比べて長い。そのため、積極的な予備の提出を促す必要がある。第 3 群の合格点到達が 120 分であるため、コンテスト期間の 3 分の 2 を経過で、予備の着手を助けるヒントを提示するなどが考えられる。また、それぞれの集団における学生の振舞いパターンが、事前に想定したものと同様の振舞いをする事が確認できた。

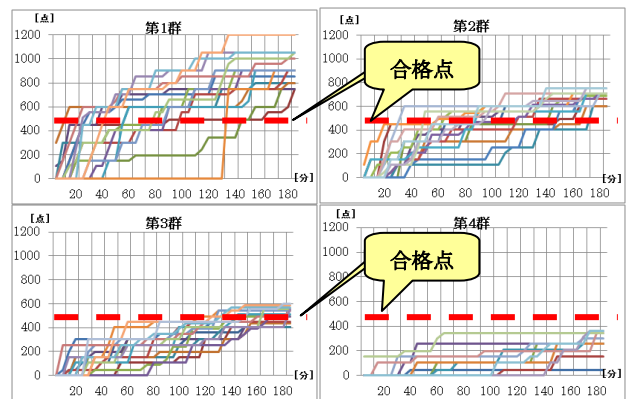


図5 個人単位の積算得点グラフ

5. おわりに

大学情報系の入門 C 授業において、初心者向けの小コンテスト形式でのプログラミング演習を提案し、大会運営の支援サーバを開発している。実行テスト系列と時間調整点を含む得点ルールを採用し、演習への積極的な参加を促す。教育実践として、試験型のコンテストを実施した。学生の積算得点の推移を成績別にグラフにし、各群の特徴を分析した。その結果、上位陣と下位陣での振舞いの特徴を発見することができた。今後の課題として、即時的に積算得点をグラフ化する機能や、成績群に応じた支援機能の開発を行い、学生への助言ができるチュータリング機能の実装も行う。

参考文献

- (1) 西村智治, 川崎慎一郎, 富永浩之:“競争型学習を取り入れた初級 C プログラミング演習における教師支援—モニタリング機能としての解答状況と提出状況の視覚化—”, 信学技報, Vol.110, No.453, pp.163-168, (2011).
- (2) 川崎慎一郎, 西村智治, 富永浩之:“競争型学習を取り入れた入門的 C プログラミング演習—解答状況による学生の振舞いパターンの分析—”, 信学技報, Vol.111, No.141, pp.53-58, (2011).