

ソフトウェア開発における詳細設計能力修得支援システムと 被験者の設計結果評価

Software Detail Design Training Support System and Analysis of Learners' Design Results

内田 浩太郎^{*1}, 仲林 清^{*1*2}
Kotaro UCHIDA^{*1}, Kiyoshi NAKABAYASHI^{*1*2}

^{*1}千葉工業大学

^{*1}Chiba Institute of Technology

^{*2}熊本大学

^{*2}Kumamoto University

Email: s1132018ht@s.chibakoudai.jp

あらまし：詳細設計の修得・向上を目的とした支援システムの開発を行い、その有用性を検証する実験を行った。本研究で開発したシステムは3種類のヒントによる補助機能を含めたビジュアルプログラミング環境である。実験では2問の関連性のあるシステム設計の課題を学習者に取り組みさせた。被験者別のヒント利用状況や設計状況を用いて評価を行った。

キーワード：ソフトウェア開発，詳細設計，ビジュアルプログラミング

1. はじめに

ソフトウェア開発における設計工程は重要である⁽¹⁾。より良い設計を行うことにより、ソースコードの質に好影響を及ぼすことが知られている⁽²⁾。またソースコードを実際に製造する前の設計がしっかりとできていない場合、開発の続行ができなくなる恐れがある。それは作るべきシステムのイメージが不明確な状態になってしまうためである。今後のプログラミング教育においても、システムの詳細設計を行う能力(以下、設計能力とする)を修得する必要がある。

従来のプログラミング学習では、この能力を修得することは困難な場合が多い。例えば解説書を用いて学習を行う場合、サンプルを写すだけの作業になってしまい、設計をする必要はない。文法について理解することは可能だが、大規模開発での応用方法がわからないため、設計能力を取得することは難しい。

そこで我々は、与えられた要件定義や外部設計から、プログラムの処理の流れを設計することについて、詳細設計能力として取り扱い、設計能力修得支

援システム(以下、支援システムとする)の開発を行い、それを用いた学習手法の検証を行ってきた⁽³⁾。支援システムを用いながら課題に取り組み、途中適切なヒントによる補助を行うことにより、設計能力修得・向上を目指した。

以降、被験者の設計結果・各ヒントの利用状況・アンケートによる評価について述べる。

2. システム概要

2.1 主要機能

本システムは、PHP・MySQLを用いたWebアプリケーションである。主要な機能として、以下の3種類があげられる。

(1) 構造化プログラミングに必要な要素を用いて処理を組み立て、実行結果を確認する機能

構造化プログラミングでは、逐次・分岐・反復といった基本的な処理の流れや、関数などを用いてプログラミングを行う。これらの要素を利用でき、実際に実行結果を確認できる機能としてVP環境を実装した。図1は開発したシステムのキャプチャである。学習者は、この画面に処理を入力していく。

☐	470.000	if	変数DisCostが0より小さかったら……
☐	480.000	var	変数DisCostに-1を加算したものを代入する
☐	490.000	end	if・while・forのブロックを終了する
☐	500.000	var	変数Costに756を代入する
☐	510.000	var	変数DisCostに108を乗算したものを代入する
☐	520.000	var	変数Costに変数DisCostを加算したものを代入する
☐	525.000	print	変数Costの中身を表示する
☒	530.000	check	変数Costを出力ナンバー0に入れる

図1 編集画面 1

(2) 学習者が容易に正否を確認することが出来る機能

実験を行うにあたって、模範解答を準備するが、同じ課題でも処理の細かい流れは学習者と模範解答で異なる場合がある。そこで入力値とそれに対応した想定される出力値を用意しておき、実行時に結果を判定する機能を実装した。

(3) ヒントによる補助機能

初学者が一から大規模なシステム設計を行うことは困難である。そこで支援システムにヒント機能を実装した。課題を解く上で補助となる情報の提供、または課題の難易度低下を行うことにより、課題への理解を促進させる。ヒント機能の一覧を表 1 に示す。

表 1 ヒント機能一覧

番号	説明
ヒント 1	文章によるヒント
ヒント 2	正しく設計した関数の提示
ヒント 3	穴埋め形式への変更

ヒント 1 では、課題全体の処理工程をおおまかに説明する文章を提示する。与えられた課題から処理工程を細かく分割することができない場合、このヒントを参照することによって一段階処理工程を詳細化したものを把握することが可能になる。

ヒント 2 では、正しく設計した関数を提示する。学習者は提示された関数をパーツとして利用するため、課題の処理が把握しやすくなる。関数を提示することによって課題の重要な部分の理解につながり、全体像を把握しやすくなる。

最後のヒントでは、正答をベースに穴埋め問題へ変更を行う。関数の提示を行っても課題を進めることができない場合、更に処理を見せ、どのようなことを行うべきシステムなのかを把握しやすくする。

これら 3 種類のヒントをもちいて課題への理解を促すことにより、能力向上を目指す。

3. 実験方法と評価

システムの有用性の検証を行う実験を情報系学部 of 大学 3・4 年生計 8 名を対象に行った。本章では実験方法と、設計結果、ヒントの利用状況について述べる。

3.1 実験方法

はじめに、研究の背景・目的・設計の基礎知識と手法例・システムの利用方法に関するレクチャーを行う。システムの利用方法については口頭説明に加え、実際に練習を行い、利用方法を確認してもらう。レクチャー終了後、支援システムを用いてシステム設計に関する 2 つの課題を解かせる。初学者にとってシステム設計は困難であるため、後半の課題は前半の課題の拡張で実現するような構成にし、前半の課題でヒントを用いながら設計の要点をつかませ、

後半の課題で実践させる。実験終了後、アンケートに回答させ、システムログとアンケートの評価終了後、必要に応じて追加のアンケートを行う。

課題は、運送業者の送料計算というテーマに設定した。前半の課題では送料計算に必要な要素を輸送距離とし、後半の課題ではそれに加え荷物サイズを設定した。後半の課題は前半の拡張で解答可能なものであり、これは前半の課題で設計方法の要点を掴ませ、後半で実践させる狙いがある。

3.2 被験者の設計結果

8 人の被験者中、5 人の被験者が課題を完成させることができた。以降この 5 人について述べる。

前半の課題では、5 人中 4 人がヒント機能を全て利用し、1 人はヒント機能を 2 つまでしか利用していない。残りの 4 人は穴埋め部分を埋めることができた。関数のヒントまでしか利用していない被験者 A は、関数を確認した後、その関数がどのような役割になるのか整理した後に、試行錯誤して課題を完成させた。

後半の課題では 5 人全てがヒントを利用していない。1 人はおおよそ模範解答通りになり、3 人は模範解答内の関数とほぼ同じ処理を、関数を利用せずに作成した。残りの 1 人は模範解答で繰り返しをしているところを、繰り返しを利用せずに作成した。

アンケートでは普段の設計に関する意識調査、課題の設計意図確認、ヒント機能の効果及び利用状況、手法全体の検証、能力向上の実感について調査を行った。

ヒント機能に関するアンケートにて、ヒント 1 に図解が欲しいといった意見が上がった。

4. おわりに

後半の課題はヒントを利用していないため、前半よりは設計を行いやすくなったと考えられる。しかし、模範解答と異なる設計を行った被験者が数名いた。必ずしも模範解答通りにならないといけないというわけではないが、設計結果を見ると繰り返しを利用していないなど、無駄な処理が目立つ結果となった。本システムはヒント機能のバリエーションが少ないため、細かい補助ができないという問題点がある。また、バリエーションを増やすなどの改善を行うことにより、被験者の設計にいい影響を与えるだろう。

参考文献

- (1) 鶴保 征城, 駒谷 昇一 : “ずっと受けたかったソフトウェアエンジニアリングの授業 2”, 翔泳社, (2006)
- (2) 工藤 英男, 杉山 裕二, 藤井 護, 鳥居 宏次 : “実験に基づくプログラム設計過程の定量化の試み”, 情報処理学会研究報告ソフトウェア工学, 1988, 第 7 号, pp. 81 – 88 (1988)
- (3) 内田浩太郎, 仲林清 : “ソフトウェア開発における詳細設計能力修得支援システムの開発”, 教育システム情報学会研究報告, Vol.30, No.1, pp.9-14(2015)