

プログラムの制御構造とそのイメージ形成について

On a Relation between Control Structure and Imagening in Programming

小池 慎一[†], 山住 富也^{††}, 長谷川 聡^{††}

Shin-ichi KOIKE, Tomiya YAMAZUMI, Satoshi HASEGAWA

Abstract We suggest a method of measuring programming capability. On learning programming, the students make the skill of programming to their mind as the abstractly image of them. Then, we can take the their capability by measuring a imagening capability.

The method is composed two kinds of test. The first is : To the students, showing a fragment of source code, let them select the image figure corresponding to it (saying image test). The second is : Let them trace a fragment of source code and answer the result of the effect of tracing (saying trace test). The latter is the same of ordinary achievement test. Here, the students have just finished the step of elementary control structure.

We did the tests for the different two groups of students. As the result, the correlation between the both test was enough strong. This shows that the image test is effective to measure the capability of programming.

1. はじめに

プログラミング学習において、その到達度は個人差がきわめて大きい。学期末になって通常クラスの1割程度のものは、課題をこなしているが、下位の2, 3割の学生は、1次元配列内データの総和を求めるプログラムさえ書けない状態で、ほとんど目標に達していない。

教師の立場から見れば、後者の学生は一步一步理解しながら先に進む努力をしないなどの学習態度が出来てないことを指摘したい。しかし、この点は教え方の問題でもあるので、ここでは言及しない。

別の観点から見ると、プログラミングには、ある具体的な処理手順を約束された文法に従って、効率よく書き換えていくと言う側面がある。これは次のように考えられる。すなわち、車の運転免許を取得したばかりの人間の運転がぎこちないのに対して、慣れたドライバは、狭い交差点に右折車がいるような複雑な状況においても、的確な判断とハンドル操作を意識に上らないでやってのける。言い換えれば、

プログラミングには与えられた処理を無意識的にプログラムの言葉に置き換える能力が必要とされるということである。

我々は、その能力がイメージを介して把握出来るのではないかと考えた。そこで、プログラミング学習の初期に学ぶ、分岐や繰り返しなどの制御構造を持つプログラムを学習者がどの程度自分のものとしているかを、イメージを介して測定することを提案する。¹⁻⁴⁾

調査の対象とする学生は、はじめてプログラミングを学ぶ受講者とする。そして、およそ制御構造の学習を終えた時点でテストした。制御構造を表現するプログラム・ソースコードの断片と、それらのイメージを示す図とを与え、前者に対応する後者の図を選択させる方法で、正しい図を選択できた得点を求めた(イメージテスト)。対照テストとして、制御構造を持つプログラムの断片をトレースさせ(トレーステスト)、実行された場合の結果の変数の値を答えさせた。

その結果、イメージテストとトレーステストとの間には、0.52~0.64の相関が認められた。すなわち、プログラムの制御構造について学習者は何らかのイ

[†] 愛知工業大学計算センター (豊田市)

^{††} 名古屋文理短期大学情報処理科 (稲沢市)

イメージを形成していると考えられる。

今後、イメージを強化するような訓練をすれば学習効果が向上するのか、逆にイメージテストは学習効果を表わすのかについて検討を要する。

2. 調査方法

2.1 イメージテスト

はじめに、制御構造のイメージ図を作成した。

制御構造には、分岐と繰り返し、さらには多分岐、繰り返しには前判定と後判定などいろいろなパターンがある。それらを図1.1に示す。さらに、分岐してから繰り返しがあったり、繰り返しの中の分岐、入れ子になった繰り返しなど、制御構造にはいろいろなパターンがある。それらの例を図1.2に示す。

学習者は、制御構造を含むソースリストの断片を見て、対応するイメージを選択する。

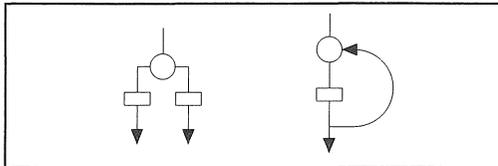


図1.1. イメージ図(分岐, 繰り返し)

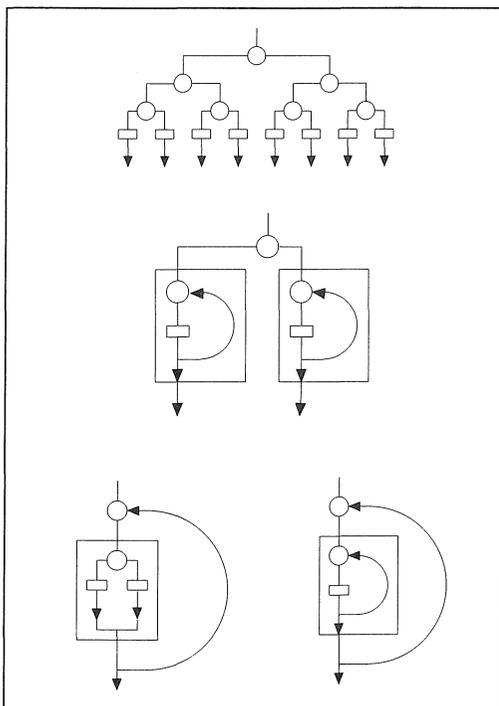


図1.2. イメージ図

(分岐と繰り返しの組合せ, 入れ子)

2.2 トレーステスト

制御構造の理解度を測るために、プログラム断片として与えられた制御構造を含むコードを1ステップずつトレースして、最終状態における変数の値を答える問題を作成した。解は、選択式でなく、得られた値そのものを記入する。分岐と繰り返しを含むテスト文の例を図2に示す。

【設問】次に示すプログラムの断片について、実行後の変数の値を答えなさい。

[1]

```

a := 0;
b := 0;
c := 0;
for i:=5 to 9 do
  if (i mod 2)=0 then a := a + i
  else
    if (i mod 3)=0 then b := b + i
    else
      c := c + i;
  
```

図2. トレーステストの設問

2.3 調査の内容

調査は対象を変えて2回行った。

【第1回目の調査】

対象 名古屋文理短期大学1年次学生 90名

言語 Pascal

テスト イメージテスト 20問

トレーステスト 30問

時期 受講開始後10週目

制御構造の学習まで修了

【第2回目の調査】

対象 愛知工業大学3年次学生 41名

言語 FORTRAN

テスト イメージテスト 20問

トレーステスト 30問

時期 受講開始後6週目

制御構造の学習まで修了

なお、2回目のテストの内容は、1回目のテストのPascalのソースコードを機械的にFORTRANに翻訳したものである。

3. 調査結果と解析

1回目と2回目のテストの結果は、類似した傾向を示したが、1回目のテストにおいては特徴的な点があった。

3・1 第1回目のテスト結果と解析

(1) 第1回目のイメージテストとトレーステストの得点の散布図を図3.1示す。全体に右肩上がりになっていて、相関がある。相関係数 $r=0.53$, $F=22.8$ となり統計的に有意である。

このデータの内容をさらに検討する。

(2) イメージテストの得点が5点以下の学生について調べてみると、イメージテストの設問のうち、特に、一重の制御構造からなる基本的な設問が出来ていない。そこで、基本的な設問（基本問題）と残りの設問（組合せ問題）との間の得点の関係をプロットしてみると図3.2の結果を得る。相関係数 $r=0.77$, $F=99.0$ になる。基本的な設問が出来ていない学生は、残りの設問がほとんどできていないこと、逆にいえば、出来る学生は、基本的な設問にも正しく答えていることが分かる。

(3) トレーステストの平均点は23.5点であった。しかし、図3.1の丸で囲まれた部分には、平均点以上をとりながら、イメージテストがトレーステストの得点に比べて低いという学生群がある。

このような学生は異質な群を構成すると考え、これらの点を除去して散布図を描くと図3.3のようになる。相関係数 $r=0.68$, $F=56.7$ となり、イメージテストとトレーステストとの相関は高いことが示される。

なお、この群の学生については、次章で考察する。

(4) トレーステストについても、構造が一重の基本的な設問の得点と、複雑に組み合わせられた設問の得点との間の関係を図3.4に示す。相関係数 $r=0.76$, $F=92.7$ となり、十分に高い相関が得られていることが分かる。

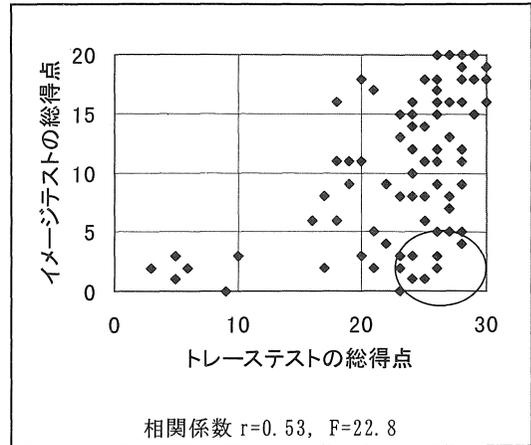


図3.1. トレーステストとイメージテストの得点の散布図

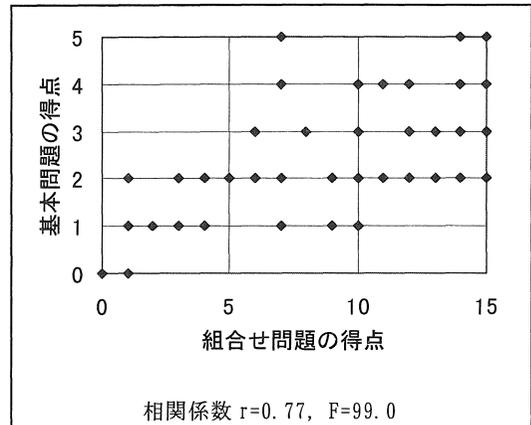


図3.2. イメージテストにおける、基本問題と組合せ問題の得点の散布図

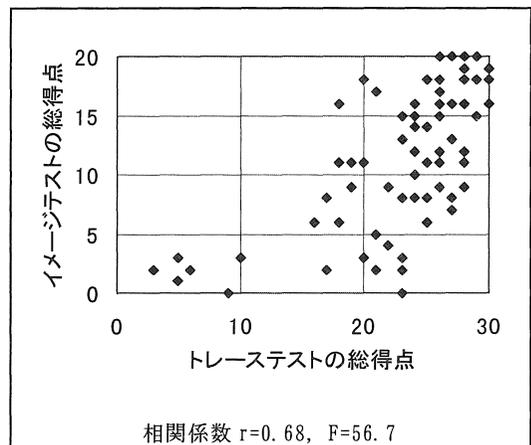


図3.3. トレーステストとイメージテストの得点の散布図

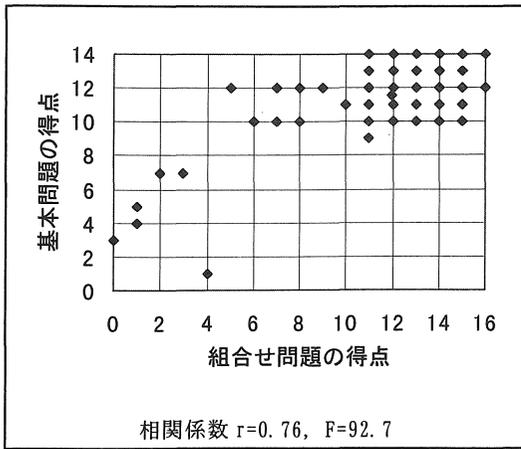


図3.4. トレーステストにおける, 基本問題と組合せ問題の得点の散布図

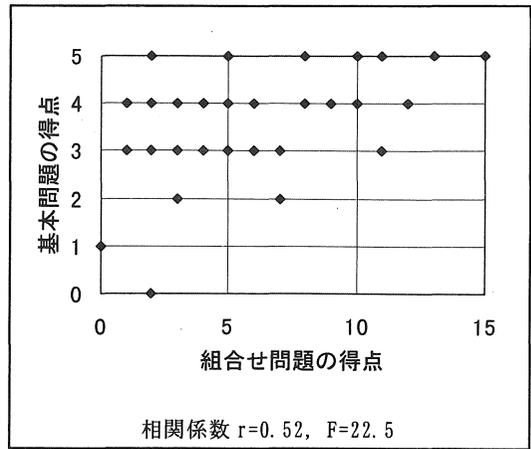


図4.2. イメージテストにおける, 基本問題と組合せ問題の得点の散布図

3・2 第2回目のテスト結果と解析0

(1) イメージテストとトレーステストの得点の散布図を図4.1に示す. 相関係数 $r=0.64$, $F=44.6$ となり, 有意に相関が認められる.

(2) イメージテストの基本的な設問と残りの設問との間の関係を図4.2に示す. 基本的な設問に対する得点と残りの設問に対する得点との間の相関係数 $r=0.52$, $F=22.5$ であり, 十分に高い相関が認められる.

(3) トレース問題についての, 基本的な設問に対する得点と残りの設問に対する得点との間の関係を図4.3に示す. 相関係数 $r=0.76$, $F=90.9$ で, 高い相関が認められる. 基本的な問題の出来ている学生が, その他の設問にもよい得点を上げている事が見られる.

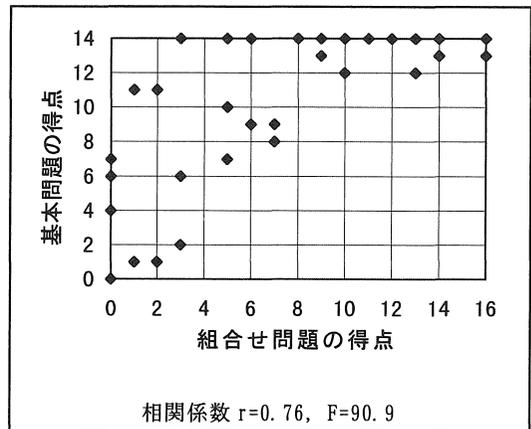


図4.3. トレーステストにおける, 基本問題と組合せ問題の得点の散布図

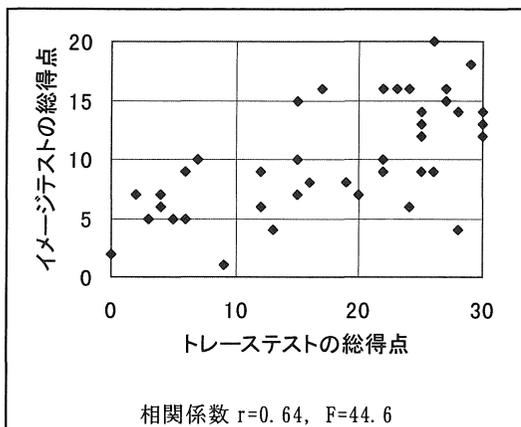


図4.1. トレーステストとイメージテストの得点の散布図

4. 調査結果の考察

上記のデータのついて考察する.

4・1 調査結果3・1について

イメージテストの得点が極めて低いにも関わらず, トレーステストに高得点をあげた学生がいたことについては以下のように考える.

- ・一回目の受講者は, 設問作成者の担当クラスである. したがって, トレーステストの設問形式に慣れていたのではないか.
- ・与えた問題が易しかったので, 設問の解き方を覚えて機械的にトレースすれば出来てしまったと考える. すなわち, 制御構造について, 受講

者なりのイメージ形成が十分なされていないとしても、この程度の設問に対して正しく解答できたということである。

4・2 調査結果 3・2 について

イメージテストとトレーステストの間には高い相関が認められた。

これは、学習者が、制御構造を理解するにつれて、自分のものとして何らかのイメージ形成をしていることを暗示させる。

4・3 一回目と二回目のテストの比較

顕著な相違は、一回目のデータでは、4・1で述べたように、図3・1の右下に見られる特別な群がある。それに対して、二回目のデータでは、イメージテストとトレーステストの得点の間には有意な相関があり、トレーステストが出来てもイメージテストの得点が低いという学生の群は見出せない。

二回目のデータは予想通りのものであることから、一回目のデータにはイメージテストを阻害するかあるいは本質的に異質な学生の群が存在すると考えられる。

4・4 考察

イメージテストとトレーステストは、テストの中に同じようなソースコードの断片を含むが、かなり性格の異なるテストである。前者では、ソースコードを見て論理的にじっくり考えないで直感的に対応するイメージ図を選択させる。後者は、ソースコードを一行づつ丹念に追跡し、条件分岐などにも注意して論理的に正しく理解し、計算をさせるものである。

従来、学習の到達度は、ここで言うトレーステストに類するテスト形式が用いられてきたが、イメージテストも高い相関を持って同様な結果を得ることが確かめられた。

4・5 その他

一回目と二回目の学習者の間の相違は、前者は、必修科目であり後者は選択科目であることである。後者の場合、当初の受講生は60名余りいたが、テスト時には3/4の41名に減じていた。

言い換えれば、選択科目の場合、自分に合わないとか、努力に耐えられないと判断した学生は早い時

期に見きりをつける。しかし、必修科目では、そうはいかない。プログラミングは不得手だと自覚する学生もテストを受けなくてはならない。したがって、図3・1に見られるような、特別な群に属する学生がいたと考えられる。

5. 結論と展望

以上の調査結果より、3・1で除いた群の学生を別にして、多くの学習者は、プログラム内の制御構造について、何らかのイメージを構成していると考ええる。

イメージ形成の進んだ学生は、トレーステストも良い成績を上げているので、イメージテストが学習の効果を測る尺度として利用できる可能性が示された。

さらに、積極的にイメージ形成を強化するような方法が見つければ、学習効果が上げられるかについて検討したい。

ここに言うイメージは、テストに用いた図のこのみならず、学習者の頭の中でなされている抽象化の作用をも意味する。したがって、イメージ図としては、図1・1、1・2に示された処理のフローに近い図形以外に、ソースコードから見ればより抽象化の高いイメージを用いても測定可能ではないかと考える。今後の課題としたい。

参考文献

- 1) 長谷川聡, 山住富也: プログラミング教育における制御構造理解のためのイメージ形成の効果, 情報処理学会第51回全国大会講演論文集, 1-295, (1995)
- 2) 長谷川聡, 山住富也, 小池慎一: プログラム教育における制御構造の理解度とイメージ形成について, 電気関係学会東海支部連合大会講演論文集, 576, p. 288, (1996)
- 3) 長谷川聡, 山住富也, 小池慎一: プログラム教育における制御構造の理解度とイメージ形成について-その2, 電気情報通信学会1997年総合大会講演論文集, D-15-26, (1996)
- 4) 長谷川聡, 山住富也: プログラミング教育と学習者のイメージ形成, 名古屋文理短期大学紀要第22号, pp. 9-14, (1997)