

開発時のテスト行動に着目したプログラミング支援ツール

橋浦研究室 1165153 柿沼 紀博

1. はじめに

テストはプログラムが仕様に基づいて正常に動作するかを確認する重要な工程である。テストを手動で行う場合、要件を満たす条件及びその結果を頭に入れておく必要があり、正しくテストされたかが担保されづらい。また、テストツールを用いて自動テストを行う場合、別途でテストコードを用意する必要があり、これを書くためにはプログラミングとは別にテストの知識が必要になる。つまりテストは非常に手間のかかる工程なのである。

2. 研究目的

本研究は、プログラミングの中のテスト過程に着目し、テストの新しいアプローチを提案する。現状開発者が手動でテストを行う場合、仕様に当てはまるデータを予め用意し実施を行う。その上で開発者は、テストの度にデータの入力を繰り返す必要が生じ、また修正のたびに結果の変化を捕捉する必要がある。これらに対しツールを用いることで、テスト網羅率の向上を目指すこととした。

3. 提案手法

前述の目的を達成するため、テストの自動記録と再実行機能を持ったツールを開発することにした。

ユーザーが手動によるテストを実行する度に、プログラムに対して入力された値と、最終的なコンソールログ（これらを以下、テストデータと呼称）を自動的に記録する。またそのテストデータからテストを再実行（以下、再テストと呼称）できるようにすることで、繰り返し行われるテストの支援に繋がると考えた。また再テスト時には、テストデータ内のコンソールログと再テスト実行時のコンソールログの記録を差分形式で表示することによって、実行結果の変化を捕捉しやすくすることにした。

4. 実装

本研究では Eclipse のプラグインとして開発することとした。Eclipse 上でプログラムが実行された際に呼ばれるコンソールを監視し、開発者がコンソール上に入力した値を記録する。また実行終了時に、実行結果として最終的なコンソールログを記録する。

記録したデータは図 1 のような表示で Eclipse のビュー上から参照できるようにした。また再テストが実行された際の実行結果の差分表示は、実行終了後に図 2 のようなダイアログ上に描画することにした。

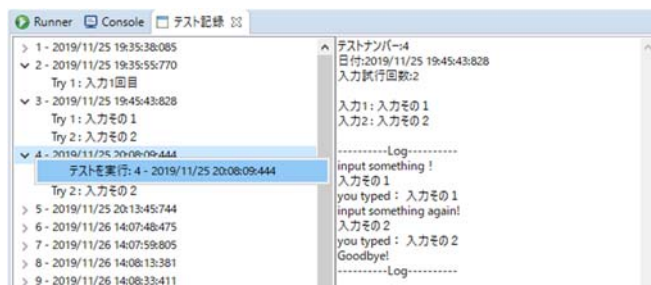


図 1 テスト記録一覧と詳細表示

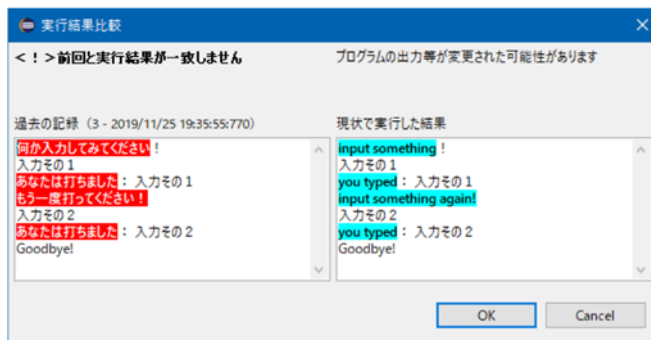


図 2 再テスト時の実行結果差分表示

5. 実験

実験にあたり、以下の RQ を立てた。

1. ツールの支援によりプログラムのテスト網羅率が向上するかどうか
2. ツールによって繰り返し行われるテストの回数が向上するかどうか

実験は、日本工業大学工学部情報工学科に所属する 8 名を被験者として行った。

被験者に実際にプログラミングを行ってもらい、その中のテスト過程で本ツールを使用する場合と、使用しない場合での比較実験を行った。課題は 2 問用意し、それぞれにテスト項目及びその結果を提示し、被験者がそれらを満たしたと挙手した時点で終了とした。

課題は 24 時間表記を 12 時間表記に変換する時間計算と、入力された数値を紙幣貨幣毎に何枚かを算出する金種計算問題の 2 問を用意した。なお、被験者それぞれが解く問題の順番とツール使用の有無は無作為に決定した。

6. 評価

今回の実験の評価指標として以下を用いた。

1. テスト記録から算出したテスト網羅率
 2. テスト記録から算出したテスト回数
- なお、それぞれの定義は以下の通りとしている。
- A) テスト網羅率 … ステートメントカバレッジ (C0)で算出

B) テスト回数 … テスト項目毎のテスト回数と、テストが累計で何回行われたかの回数

7. 実験結果

実験結果を示すに当たり、検定で用いる有意水準は0.05とする。

7.1. テスト網羅率

まず実験において算出されたテスト網羅率を表1に示す。

表 1 ツール毎のテスト網羅率

| | ツール毎のテストカバレッジ(C0) | | | | | | | | 平均 |
|-------|-------------------|-------|-------|-------|-------|-------|-------|-------|-------|
| ツールあり | 96.4% | 97.4% | 97.8% | 97.2% | 97.4% | 98.9% | 95.9% | 98.8% | 97.5% |
| ツールなし | 85.3% | 98.4% | 95.7% | 95.0% | 88.9% | 93.8% | 97.9% | 77.3% | 91.5% |

これをF検定にかけた結果を表2に示す。帰無仮説 H_0 を「ツールによってテスト網羅率の分散に差がない」、対立仮説 H_1 を「ツールによってテスト網羅率の分散に差がある」とした。 $P < 0.05$ より帰無仮説を棄却、ツールによって分散に差があることが確認された。

表 2 テスト網羅率のF検定結果

| | ツールなし | ツールあり |
|----------------------|-------------------|-------------|
| 平均 | 91.5375 | 97.475 |
| 分散 | 52.93982143 | 1.087857143 |
| 観測数 | 8 | 8 |
| 自由度 | 7 | 7 |
| 観測された分散比 | 48.66431385 | |
| P(F<=f) 片側 | 2.0741E-05 | |
| F境界値 片側 | 3.78704354 | |

さらに異分散でt検定を行なった結果を表3に示す。帰無仮説 H_0 を「ツールによってテスト網羅率の平均に差はない」、対立仮説 H_1 を「ツールによってテスト網羅率の平均に差がある」とし、検定を行なった。

Pの片側において $P < 0.05$ より帰無仮説を棄却、ツールによる有意差が認められると確認された。

表 3 テスト網羅率の異分散t検定結果

| | ツールなし | ツールあり |
|----------------------|------------------|-----------|
| 平均 | 91.5375 | 97.475 |
| 分散 | 52.939821 | 1.0878571 |
| 観測数 | 8 | 8 |
| 仮説平均との差異 | 0 | |
| 自由度 | 7 | |
| t | -2.284759 | |
| P(T<=t) 片側 | 0.0281201 | |
| t境界値 片側 | 1.8945786 | |
| P(T<=t) 両側 | 0.0562403 | |
| t境界値 両側 | 2.3646243 | |

7.2. テスト回数

テスト網羅率に関連して、テスト回数についても検定を行う。ツール毎のテスト回数を表4に示す。これに対してF検定及びt検定を行なった結果を表5に示す。

表 4 ツール毎のテスト回数

| | ツール毎のテスト回数 | | | | | | | | 平均 |
|-------|------------|----|----|----|----|----|----|----|------|
| ツールあり | 12 | 14 | 38 | 21 | 16 | 37 | 34 | 18 | 23.8 |
| ツールなし | 40 | 31 | 26 | 72 | 14 | 9 | 20 | 33 | 30.6 |

表 5 テスト回数の検定結果

| | | |
|-----|------------|----------------------|
| F検定 | 帰無仮説 H_0 | ツールによってテスト回数に分散の差は無い |
| | 対立仮説 H_1 | ツールによってテスト回数に分散の差がある |
| | 結果 | 分散の差は無い (=等分散) |
| t検定 | 帰無仮説 H_0 | ツールによるテスト回数の差は無い |
| | 対立仮説 H_1 | ツールがない方がテスト回数が多い |
| | 結果 | 回数に差はない |
| | 検定結果 | ツールによるテスト回数の差は無い |

8. 分析と考察

実験結果 7.1 において、ツールによってテスト網羅率が向上したと確認された。しかしながら実験結果 7.2 では、テスト回数にはツールによる差が認められなかった。これはテスト回数に変化はないものの、テスト網羅率だけが向上したということである。これについて、実験によって得られたテストデータと提出されたソースコードを分析したが、明確な理由を突き止めることはできなかった。代わりにツールが使用された形跡を探して分析することにした。

先の分析に使用した、本ツールを用いて記録されたテストデータを分析したものの一例を表6に示す。

表 6 被験者 H の課題 B テスト項目①の分析

| 試行 | 備考 |
|----|----------------------------|
| 1 | 繰り返し処理のミス (出力結果が正しくない) |
| 2 | 上に同じ |
| 3 | 紙幣・貨幣の一覧表示に成功 (値は全て0) |
| 4 | 1つ目の項目のみ値の表示成功 (おそらく実行テスト) |
| 5 | 全て表示するも、一部動作不良 (計算ミス) |
| 6 | 全表示に成功 (この時点で要件満了) |
| 7 | 全表示に成功 (おそらく動作チェック) |

課題 B のテスト項目の1つについて、全試行数が7回で、手動による初回実行以外は全てツールによる再テストの試行であった。それぞれの試行についてテストデータを分析してみると、表6の備考欄のような動作結果の推察ができた。

これらについて、ツールの再テスト機能を用いて行われたテストに絞って、ツールの効果が発生したと考えられる箇所を列挙したものを表7に示す。表7の3つ目の項目について例として挙げる。テストデータを見ると、出力値が0~12しか取り得ないものについて「49」や「73」のような値を出力するものがあった。その点の変化が前回とどのように変わったのか、ツールによって確実に捕捉することができたのではないかと考える。

また5つ目の項目について、「AM」と「PM」のわずかな変化の捕捉に、ツールの差分表示が役に立ったの

ではないかと考える。

検定による結果の根拠を示すことはできなかったが、ツールの機能の実行結果の差分表示が使われたと考えられる点と、実際の実行結果の変化を鑑みるに、ツールの有効性は十分にあるのではないかと考える。

表 7 ツールの効果が発生したと思われる箇所

| 項目 | 考えられるツールの効果 |
|---------------|---------------|
| 1 表示項目不足 | 1行分の変化を確認か |
| 2 数値以外の入力でエラー | エラー内容の変化を確認か |
| 3 正常な値が出力されない | 数値の変化を確認か |
| 4 途中で他のテストを実行 | 他の修正による影響を確認か |
| 5 AMとPMの表示ミス | 2文字の変化を確認か |
| 6 繰り返し処理のミス | 1つ1つ修正されたか確認か |
| 7 動作不良 | 余分な変化を確認か |

9. まとめと今後の課題

本研究のツールを用いることでテスト網羅率が向上することが確認されたが、それを裏付ける根拠を示すことができなかった。しかし、ツールによってテストが観測できるようになり、テストが確実に行われているかがわかるようになった。

今後の課題として、ツールによってテスト網羅率が向上したのかを明確に示すため、実験で取得するデータや、実験方法の再検討が必要である。具体的には高山ら[1]の開発したツールのように、プログラムの変更過程を記録できるようにすることが挙げられる。

文 献

- [1] 高山 源貫, 山田 善隆, 橋浦 弘明, 田中 昂文, 樋山 淳雄, 高瀬 浩史 “細粒度履歴を用いたプログラムの変更の方略推測支援ツール,” 電子情報通信学会技術研究報告, Vol.117, No.295, pp.19 - 24, 2017-11-10.
- [2] 蜂巢 吉成, 小林 悟, 吉田 敦, 阿草 清滋, “プログラミング演習におけるテストケース評価システムの提案,” コンピュータソフトウェア, Vol.34, No.4, pp.4_54 - 4_60, 2017
- [3] 浜田 信, 山田 哲靖, 岡本 明, “プログラム変更を考慮したテストカバレッジ率に関する一考察,” 全国大会講演論文集, 第 59 回, ソフトウェア科学・工学, pp.287 - 288, 1999-09-28
- [4] 玉井 哲雄, 三嶋 良武, 松田 茂広, “ソフトウェアのテスト技法,” 共立出版株式会社, 1988