

# SV2016 1차 시스템 테스트 보고서

201214262 라가영      201313250 서지혁

May 26, 2016

# Contents

<b>Contents</b>	<b>2</b>
<b>1 Abstract</b>	<b>3</b>
<b>2 Specification Review</b>	<b>3</b>
2.1 Functional Requirements . . . . .	3
2.2 Non-Functional Requirements . . . . .	3
2.3 Performance Requirements . . . . .	3
2.4 Operating Environments . . . . .	3
2.5 Interface Requirements . . . . .	4
2.6 Other Requirements . . . . .	4
<b>3 Brute-Force Testing</b>	<b>5</b>
3.1 Use Case Testing . . . . .	5
<b>4 Category-Partition Testing</b>	<b>6</b>
4.1 Test Specification Language . . . . .	6
4.2 Test Frame Generation . . . . .	7
4.3 Test Results . . . . .	23
<b>5 Pairwise Testing</b>	<b>25</b>
5.1 Test Modelling . . . . .	25
5.2 Test Case Generation . . . . .	25
5.3 Test Results . . . . .	26
<b>6 Miscellaneous</b>	<b>27</b>
<b>References</b>	<b>27</b>

## 1 Abstract

이 문서는 2016년 봄 건국대학교 소프트웨어 검증[1] 과목의 과제로 작성되었다.

이 보고서에서는 소프트웨어 모델링 및 분석[2] 과목의 과제로 개발되는, “다수의 C 프로그램을 대상으로 상호 cheating 여부를 정량적으로 판단하고, 해당 내용을 OOO 하게 알려주는 프로그램”을 검증한다. Clone Checking 예제로는 프로그래밍 프로젝트[3]의 과제로 제출되는 코드를 사용한다.

## 2 Specification Review

SMA T1의 *OOPT Stage 1000, Plan and Elaboration* 보고서의 4번째 판을 기준으로 요구 사항 명세 검토를 진행하였다.

### 2.1 Functional Requirements

요구 사항 항목 “R.1, Select Folder, 비교할 소스 코드 파일들이 들어있는 폴더를 선택한다.” 에서 “소스 코드”와 “소스 코드 파일”에 대한 정의가 불분명하다. 소스 코드는 어떤 프로그래밍 언어로 작성되어야 하는가? 해당 프로그래밍 언어의 명세 상으로 유효한, 즉 컴파일 가능한 코드여야 하는가? 소스 코드 파일의 크기에는 어떤 제한이 있는가?

요구 사항 항목 “R.4, Calculate Sync-Rate, 분석하여 가공한 정보를 비교하여 일치율을 계산한다” 의 가공된 정보의 비교 방식과 일치율의 계산 기준이 명시되어 있지 않다. R.4.1, R.4.2, R.4.3, R.4.4, R.4.5 도 같은 문제가 있다.

### 2.2 Non-Functional Requirements

- “A. 설명이 필요 없을 정도의 조작하기 쉬운 인터페이스” 에서 주관적인 어휘가 다수 사용되었으며, 기준이 불분명하다.
- “B. 한눈에 이해 가능한 결과 출력” 상동.
- “C. 다른 언어의 소스 파일도 비교 가능하도록 하는 확장성” 에서 “다른” 의 기준이 되는 언어가 명세되어 있지 않다.

### 2.3 Performance Requirements

“A. 소스 코드 파일 분석 작업은 5초 이내로 수행되어야 한다” 고 명세되어 있는데, 분석 대상이 되는 코드의 최대량 또는 기준량이 명시되어 있지 않았으며, “B. 기준 소스 코드 파일 변경 시 2초 이내로 비교 결과가 변경되어야 한다.” 역시 같은 문제가 있다.

### 2.4 Operating Environments

하나의 바이너리로 여러 운영 체제에서 실행될 수 있는 Java 프로그래밍 언어를 채택하였는데 “Microsoft Windows 7 이상” 의 실행 환경만을 지원한다.

## 2.5 Interface Requirements

“B.4, 종료 버튼, 화면 하단에 위치하여 클릭 시 프로그램이 종료된다”, “C. 그 외의 메시지 출력은 대화상자(Dialog) 활용” 이라고 명세되어 있는데, 종료 버튼과 대화상자가 없다.

## 2.6 Other Requirements

“A. 비교 결과(일치율, 유사항목)를 한 눈에 알아볼 수 있도록 비교 결과 출력 화면이 직관적이어야 한다.” 에서 직관성의 판단 기준이 명시되어 있지 않다.

# 3 Brute-Force Testing

## 3.1 Use Case Testing

- 메인 인터페이스 창을 최소화 하였다 복구시키니 사용자 인터페이스가 육안으로 확인할 수 없을 정도로 작아지며 원상 복구되지 않는다.
- 명세서 상 비교 결과 일치율 85% 이상인 파일은 클론으로 판정하고, 이 결과를 한 눈에 알아볼 수 있도록 직관적이어야 하는데 클론 판정 유무가 인터페이스에 드러나지 않는다.
- `/* comment */` 스타일의 주석을 사용한 코드를 분석하였을 때 클론 체커가 멈춘다.
- 세미콜론을 사용해 코드 여러 줄을 합쳤을 때 “Use Case: Divide Code Line” 기능이 작동하지 않는다.

## 4 Category-Partition Testing

### 4.1 Test Specification Language

Ostrand와 Balcers의 1988 CACM 논문[4]에서 Category-Partition Testing을 위해 정의한 도메인 특정 언어인 TSL(Test Specification Language)을 사용해 테스트 카테고리화 선택지를 명세하였다.

Listing 1: TSL로 작성된 테스트 명세

```
Line :
  Lines :
    Zero . [single]
    Positive .

Annotation :
  Comments :
    Zero .
    Positive . [property ContainsComment]
  Style :
    Single-line . [if ContainsComment]
    Multi-line . [if ContainsComment] [property MultiLineComment]
  Matching :
    True . [if MultiLineComment]
    False . [if MultiLineComment]

Function :
  Functions :
    Zero .
    Positive .

Variable :
  Variables :
    Zero .
    Positive .

Preprocessor :
  Includes :
    Zero .
    Positive .
  Defines :
    Zero .
    Positive .
```

## 4.2 Test Frame Generation

Category-Partition 테스트 명세를 TSL로 작성하고, tslgenerator[5]를 사용해 테스트 프레임 생성을 자동화 하였다.

Listing 2: TSL을 통해 생성된 테스트 프레임

```
Test Case 1                <single>
  Lines : Zero

Test Case 2                (Key = 2.1.0.0.1.1.1.1.)
  Lines      : Positive
  Comments   : Zero
  Style      : <n/a>
  Matching   : <n/a>
  Functions  : Zero
  Variables  : Zero
  Includes   : Zero
  Defines    : Zero

Test Case 3                (Key = 2.1.0.0.1.1.1.2.)
  Lines      : Positive
  Comments   : Zero
  Style      : <n/a>
  Matching   : <n/a>
  Functions  : Zero
  Variables  : Zero
  Includes   : Zero
  Defines    : Positive

Test Case 4                (Key = 2.1.0.0.1.1.2.1.)
  Lines      : Positive
  Comments   : Zero
  Style      : <n/a>
  Matching   : <n/a>
  Functions  : Zero
  Variables  : Zero
  Includes   : Positive
  Defines    : Zero

Test Case 5                (Key = 2.1.0.0.1.1.2.2.)
  Lines      : Positive
```

Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Zero  
Variables : Zero  
Includes : Positive  
Defines : Positive

Test Case 6 (Key = 2.1.0.0.1.2.1.1.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Zero  
Variables : Positive  
Includes : Zero  
Defines : Zero

Test Case 7 (Key = 2.1.0.0.1.2.1.2.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Zero  
Variables : Positive  
Includes : Zero  
Defines : Positive

Test Case 8 (Key = 2.1.0.0.1.2.2.1.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Zero  
Variables : Positive  
Includes : Positive  
Defines : Zero

Test Case 9 (Key = 2.1.0.0.1.2.2.2.)

Lines : Positive  
Comments : Zero  
Style : <n/a>

Matching : <n/a>  
Functions : Zero  
Variables : Positive  
Includes : Positive  
Defines : Positive

Test Case 10 (Key = 2.1.0.0.2.1.1.1.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Positive  
Variables : Zero  
Includes : Zero  
Defines : Zero

Test Case 11 (Key = 2.1.0.0.2.1.1.2.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Positive  
Variables : Zero  
Includes : Zero  
Defines : Positive

Test Case 12 (Key = 2.1.0.0.2.1.2.1.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Positive  
Variables : Zero  
Includes : Positive  
Defines : Zero

Test Case 13 (Key = 2.1.0.0.2.1.2.2.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Positive



Variables : Zero  
Includes : Positive  
Defines : Positive

Test Case 14 (Key = 2.1.0.0.2.2.1.1.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Positive  
Variables : Positive  
Includes : Zero  
Defines : Zero

Test Case 15 (Key = 2.1.0.0.2.2.1.2.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Positive  
Variables : Positive  
Includes : Zero  
Defines : Positive

Test Case 16 (Key = 2.1.0.0.2.2.2.1.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Positive  
Variables : Positive  
Includes : Positive  
Defines : Zero

Test Case 17 (Key = 2.1.0.0.2.2.2.2.)

Lines : Positive  
Comments : Zero  
Style : <n/a>  
Matching : <n/a>  
Functions : Positive  
Variables : Positive  
Includes : Positive

Defines	:	Positive
---------	---	----------

  

Test Case 18		(Key = 2.2.1.0.1.1.1.1.)
Lines	:	Positive
Comments	:	Positive
Style	:	Single-line
Matching	:	<n/a>
Functions	:	Zero
Variables	:	Zero
Includes	:	Zero
Defines	:	Zero

  

Test Case 19		(Key = 2.2.1.0.1.1.1.2.)
Lines	:	Positive
Comments	:	Positive
Style	:	Single-line
Matching	:	<n/a>
Functions	:	Zero
Variables	:	Zero
Includes	:	Zero
Defines	:	Positive

  

Test Case 20		(Key = 2.2.1.0.1.1.2.1.)
Lines	:	Positive
Comments	:	Positive
Style	:	Single-line
Matching	:	<n/a>
Functions	:	Zero
Variables	:	Zero
Includes	:	Positive
Defines	:	Zero

  

Test Case 21		(Key = 2.2.1.0.1.1.2.2.)
Lines	:	Positive
Comments	:	Positive
Style	:	Single-line
Matching	:	<n/a>
Functions	:	Zero
Variables	:	Zero
Includes	:	Positive
Defines	:	Positive

Test Case 22 (Key = 2.2.1.0.1.2.1.1.)  
Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Zero  
Variables : Positive  
Includes : Zero  
Defines : Zero

Test Case 23 (Key = 2.2.1.0.1.2.1.2.)  
Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Zero  
Variables : Positive  
Includes : Zero  
Defines : Positive

Test Case 24 (Key = 2.2.1.0.1.2.2.1.)  
Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Zero  
Variables : Positive  
Includes : Positive  
Defines : Zero

Test Case 25 (Key = 2.2.1.0.1.2.2.2.)  
Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Zero  
Variables : Positive  
Includes : Positive  
Defines : Positive

Test Case 26 (Key = 2.2.1.0.2.1.1.1.)

Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Positive  
Variables : Zero  
Includes : Zero  
Defines : Zero

Test Case 27 (Key = 2.2.1.0.2.1.1.2.)

Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Positive  
Variables : Zero  
Includes : Zero  
Defines : Positive

Test Case 28 (Key = 2.2.1.0.2.1.2.1.)

Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Positive  
Variables : Zero  
Includes : Positive  
Defines : Zero

Test Case 29 (Key = 2.2.1.0.2.1.2.2.)

Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Positive  
Variables : Zero  
Includes : Positive  
Defines : Positive

Test Case 30 (Key = 2.2.1.0.2.2.1.1.)

Lines : Positive  
Comments : Positive

Style : Single-line  
Matching : <n/a>  
Functions : Positive  
Variables : Positive  
Includes : Zero  
Defines : Zero

Test Case 31 (Key = 2.2.1.0.2.2.1.2.)

Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Positive  
Variables : Positive  
Includes : Zero  
Defines : Positive

Test Case 32 (Key = 2.2.1.0.2.2.2.1.)

Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Positive  
Variables : Positive  
Includes : Positive  
Defines : Zero

Test Case 33 (Key = 2.2.1.0.2.2.2.2.)

Lines : Positive  
Comments : Positive  
Style : Single-line  
Matching : <n/a>  
Functions : Positive  
Variables : Positive  
Includes : Positive  
Defines : Positive

Test Case 34 (Key = 2.2.2.1.1.1.1.1.)

Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : True

Functions : Zero  
Variables : Zero  
Includes : Zero  
Defines : Zero

Test Case 35 (Key = 2.2.2.1.1.1.1.2.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : True  
Functions : Zero  
Variables : Zero  
Includes : Zero  
Defines : Positive

Test Case 36 (Key = 2.2.2.1.1.1.2.1.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : True  
Functions : Zero  
Variables : Zero  
Includes : Positive  
Defines : Zero

Test Case 37 (Key = 2.2.2.1.1.1.2.2.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : True  
Functions : Zero  
Variables : Zero  
Includes : Positive  
Defines : Positive

Test Case 38 (Key = 2.2.2.1.1.2.1.1.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : True  
Functions : Zero  
Variables : Positive

Includes	:	Zero
Defines	:	Zero

  

Test Case 39		(Key = 2.2.2.1.1.2.1.2.)
Lines	:	Positive
Comments	:	Positive
Style	:	Multi-line
Matching	:	True
Functions	:	Zero
Variables	:	Positive
Includes	:	Zero
Defines	:	Positive

  

Test Case 40		(Key = 2.2.2.1.1.2.2.1.)
Lines	:	Positive
Comments	:	Positive
Style	:	Multi-line
Matching	:	True
Functions	:	Zero
Variables	:	Positive
Includes	:	Positive
Defines	:	Zero

  

Test Case 41		(Key = 2.2.2.1.1.2.2.2.)
Lines	:	Positive
Comments	:	Positive
Style	:	Multi-line
Matching	:	True
Functions	:	Zero
Variables	:	Positive
Includes	:	Positive
Defines	:	Positive

  

Test Case 42		(Key = 2.2.2.1.2.1.1.1.)
Lines	:	Positive
Comments	:	Positive
Style	:	Multi-line
Matching	:	True
Functions	:	Positive
Variables	:	Zero
Includes	:	Zero
Defines	:	Zero





Test Case 47 (Key = 2.2.2.1.2.2.1.2.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : True  
Functions : Positive  
Variables : Positive  
Includes : Zero  
Defines : Positive

Test Case 48 (Key = 2.2.2.1.2.2.2.1.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : True  
Functions : Positive  
Variables : Positive  
Includes : Positive  
Defines : Zero

Test Case 49 (Key = 2.2.2.1.2.2.2.2.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : True  
Functions : Positive  
Variables : Positive  
Includes : Positive  
Defines : Positive

Test Case 50 (Key = 2.2.2.2.1.1.1.1.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Zero  
Variables : Zero  
Includes : Zero  
Defines : Zero

Test Case 51 (Key = 2.2.2.2.1.1.1.2.)  
Lines : Positive

Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Zero  
Variables : Zero  
Includes : Zero  
Defines : Positive

Test Case 52 (Key = 2.2.2.2.1.1.2.1.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Zero  
Variables : Zero  
Includes : Positive  
Defines : Zero

Test Case 53 (Key = 2.2.2.2.1.1.2.2.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Zero  
Variables : Zero  
Includes : Positive  
Defines : Positive

Test Case 54 (Key = 2.2.2.2.1.2.1.1.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Zero  
Variables : Positive  
Includes : Zero  
Defines : Zero

Test Case 55 (Key = 2.2.2.2.1.2.1.2.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line

Matching : False  
Functions : Zero  
Variables : Positive  
Includes : Zero  
Defines : Positive

Test Case 56 (Key = 2.2.2.2.1.2.2.1.)

Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Zero  
Variables : Positive  
Includes : Positive  
Defines : Zero

Test Case 57 (Key = 2.2.2.2.1.2.2.2.)

Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Zero  
Variables : Positive  
Includes : Positive  
Defines : Positive

Test Case 58 (Key = 2.2.2.2.2.1.1.1.)

Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Positive  
Variables : Zero  
Includes : Zero  
Defines : Zero

Test Case 59 (Key = 2.2.2.2.2.1.1.2.)

Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Positive

Variables : Zero  
Includes : Zero  
Defines : Positive

Test Case 60 (Key = 2.2.2.2.2.1.2.1.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Positive  
Variables : Zero  
Includes : Positive  
Defines : Zero

Test Case 61 (Key = 2.2.2.2.2.1.2.2.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Positive  
Variables : Zero  
Includes : Positive  
Defines : Positive

Test Case 62 (Key = 2.2.2.2.2.2.1.1.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Positive  
Variables : Positive  
Includes : Zero  
Defines : Zero

Test Case 63 (Key = 2.2.2.2.2.2.1.2.)  
Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Positive  
Variables : Positive  
Includes : Zero

Defines : Positive

Test Case 64 (Key = 2.2.2.2.2.2.2.1.)

Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Positive  
Variables : Positive  
Includes : Positive  
Defines : Zero

Test Case 65 (Key = 2.2.2.2.2.2.2.2.)

Lines : Positive  
Comments : Positive  
Style : Multi-line  
Matching : False  
Functions : Positive  
Variables : Positive  
Includes : Positive  
Defines : Positive

### **4.3 Test Results**

다음은 생성된 테스트들의 테스트 결과이다.

#	Key	Result
1	2.1.0.0.1.1.1.1.	Pass
2	2.1.0.0.1.1.1.2.	Pass
3	2.1.0.0.1.1.2.1.	Pass
4	2.1.0.0.1.1.2.2.	Pass
5	2.1.0.0.1.2.1.1.	Pass
6	2.1.0.0.1.2.1.2.	Pass
7	2.1.0.0.1.2.2.1.	Pass
8	2.1.0.0.1.2.2.2.	Pass
9	2.1.0.0.2.1.1.1.	Pass
10	2.1.0.0.2.1.1.2.	Pass
11	2.1.0.0.2.1.2.1.	Pass
12	2.1.0.0.2.1.2.2.	Pass
13	2.1.0.0.2.2.1.1.	Pass
14	2.1.0.0.2.2.1.2.	Pass
15	2.1.0.0.2.2.2.1.	Pass
16	2.1.0.0.2.2.2.2.	Pass
17	2.2.1.0.1.1.1.1.	Pass
18	2.2.1.0.1.1.1.2.	Pass
19	2.2.1.0.1.1.2.1.	Pass
20	2.2.1.0.1.1.2.2.	Pass
21	2.2.1.0.1.2.1.1.	Pass
22	2.2.1.0.1.2.1.2.	Pass
23	2.2.1.0.1.2.2.1.	Pass
24	2.2.1.0.1.2.2.2.	Pass
25	2.2.1.0.2.1.1.1.	Pass
26	2.2.1.0.2.1.1.2.	Pass
27	2.2.1.0.2.1.2.1.	Pass
28	2.2.1.0.2.1.2.2.	Pass
29	2.2.1.0.2.2.1.1.	Pass
30	2.2.1.0.2.2.1.2.	Pass
31	2.2.1.0.2.2.2.1.	Pass
32	2.2.1.0.2.2.2.2.	Pass
33	2.2.2.1.1.1.1.1.	Fail
34	2.2.2.1.1.1.1.2.	Fail
35	2.2.2.1.1.1.2.1.	Fail
36	2.2.2.1.1.1.2.2.	Fail
37	2.2.2.1.1.2.1.1.	Fail
38	2.2.2.1.1.2.1.2.	Fail
39	2.2.2.1.1.2.2.1.	Fail
40	2.2.2.1.1.2.2.2.	Fail

41	2.2.2.1.2.1.1.1.	Fail
42	2.2.2.1.2.1.1.2.	Fail
43	2.2.2.1.2.1.2.1.	Fail
44	2.2.2.1.2.1.2.2.	Fail
45	2.2.2.1.2.2.1.1.	Fail
46	2.2.2.1.2.2.1.2.	Fail
47	2.2.2.1.2.2.2.1.	Fail
48	2.2.2.1.2.2.2.2.	Fail
49	2.2.2.2.1.1.1.1.	Pass
50	2.2.2.2.1.1.1.2.	Pass
51	2.2.2.2.1.1.2.1.	Pass
52	2.2.2.2.1.1.2.2.	Pass
53	2.2.2.2.1.2.1.1.	Pass
54	2.2.2.2.1.2.1.2.	Pass
55	2.2.2.2.1.2.2.1.	Pass
56	2.2.2.2.1.2.2.2.	Pass
57	2.2.2.2.2.1.1.1.	Pass
58	2.2.2.2.2.1.1.2.	Pass
59	2.2.2.2.2.1.2.1.	Pass
60	2.2.2.2.2.1.2.2.	Pass
61	2.2.2.2.2.2.1.1.	Pass
62	2.2.2.2.2.2.1.2.	Pass
63	2.2.2.2.2.2.2.1.	Pass
64	2.2.2.2.2.2.2.2.	Pass

## 5 Pairwise Testing

Pairwise 테스트는 소프트웨어 에러가 대부분 두 개의 파라미터가 잘못 섞여서 발생하는 점에 착안해서 해당 파라미터 쌍에 대해서만 조합해서 테스트하는 방식이다. Category-Partition은 모든 n-pair 에 대한 조합을 exhaustive 하게 생성하며, Pairwise는 튜플에 대해서만 생성한다.

테스트 케이스의 커버리지는 Category-Partition이 높지만 Pairwise는 현실적인 경향에 타협하여 테스트 케이스 갯수 대비 효율이 높다. 조합만 하면 되는 Category-Partition 테스트 케이스 생성기와 다르게 Pairwise 테스트 케이스 생성 도구는 최적화 알고리즘이나 휴리스틱이 중요하다.

### 5.1 Test Modelling

PICT[7]는 Microsoft에서 개발한 Pairwise 테스트 케이스 생성 도구이다. 우리는 PICT가 정의하는 모델 포맷으로 테스트 모델링을 하였다.

Listing 3: PICT 포맷으로 모델링된 테스트 명세

```
Lines: Zero, Positive
Comments: Zero, Positive
Style: Single-line, Multi-line
Matching: True, False
Functions: Zero, Positive
Variables: Zero, Positive
Includes: Zero, Positive
Defines: Zero, Positive
```

### 5.2 Test Case Generation

PICT 모델로부터 테스트 케이스 생성을 하였다.

Lines	Comments	Style	Matching	Functions	Variables	Includes	Defines
Positive	Positive	Single-line	False	Positive	Zero	Zero	Zero
Zero	Zero	Multi-line	True	Zero	Positive	Positive	Zero
Zero	Zero	Single-line	False	Zero	Positive	Zero	Positive
Positive	Positive	Multi-line	True	Zero	Zero	Positive	Positive
Zero	Positive	Multi-line	False	Positive	Positive	Positive	Positive
Positive	Zero	Single-line	True	Positive	Positive	Zero	Zero
Zero	Zero	Single-line	True	Zero	Zero	Positive	Positive
Positive	Positive	Multi-line	True	Positive	Zero	Zero	Positive

Table 1: PICT 모델로부터 생성된 테스트 케이스



### 5.3 Test Results

Lines	Comments	Style	Matching	Functions	Variables	Includes	Defines	Result
Positive	Positive	Single-line	False	Positive	Zero	Zero	Zero	Pass
Zero	Zero	Multi-line	True	Zero	Positive	Positive	Zero	Fail
Zero	Zero	Single-line	False	Zero	Positive	Zero	Positive	Pass
Positive	Positive	Multi-line	True	Zero	Zero	Positive	Positive	Pass
Zero	Positive	Multi-line	False	Positive	Positive	Positive	Positive	Fail
Positive	Zero	Single-line	True	Positive	Positive	Zero	Zero	Pass
Zero	Zero	Single-line	True	Zero	Zero	Positive	Positive	Pass
Positive	Positive	Multi-line	True	Positive	Zero	Zero	Positive	Pass

Table 2: Pairwise 테스트 결과

## 6 Miscellaneous

OOPT Stage 1000 보고서 Activity 1005 에 첨부된 스크린샷이 테스트 해당 버전으로 갱신되어 되지 않았으며, SMA에서 진행한 유닛 테스트가 나열된 유스케이스들을 커버하지 못하였다.

또, Category-Partition 테스트 명세로 코드의 컴파일 가능성, 네스팅된 주석, 더 다양한 코드 구조 등의 옵션을 넣고 싶었지만 시스템 테스트 과정에서 테스트가 폭발적으로 많아져 아이디어를 기각할 수 밖에 없었다. 특히 변수 갯수, 함수 갯수 등의 테스트 항목으로  $2^{31}$  등의 특수한 값들을 넣거나 타입 정보도 변화시키고 싶었지만 테스트 갯수가 너무 많아 현 상황에서는 테스트하지 않았다.

## References

- [1] Junbeom Yoo, Eui-Sub Kim, *Software Verification – Introduction to Software Testing & Static Analysis*, <http://dslab.konkuk.ac.kr/Class/2016/16SV/16SV.htm>, 2016.
- [2] Junbeom Yoo, Sejin Jung, *Software Modeling & Analysis*, <http://dslab.konkuk.ac.kr/Class/2016/16SMA/16SMA.htm>, 2016.
- [3] Junbeom Yoo, Jaeyoeb Kim, *Programming Project*, <http://dslab.konkuk.ac.kr/Class/2016/16PP/16PP.htm>, 2016.
- [4] T. J. Ostrand, M. J. Balcer, *The category-partition method for specifying and generating functional tests*, 1988.
- [5] Alex Orso, *TSL Generator for the Category Partition Method*, <https://github.com/alexorso/tslgenerator>
- [6] Jacek Czerwonka, *Pairwise Testing*, <http://www.pairwise.org>
- [7] Microsoft Corporation, *Pairwise Independent Combinatorial Tool*, <https://github.com/Microsoft/pict>
- [8] Jacek Czerwonka, *Pairwise Testing in the Real World: Practical Extensions to Test-Case Scenarios*, <https://msdn.microsoft.com/en-us/library/cc150619.aspx>, 2008.