

Software Modeling & Analysis

One More Chance

OOPT stage 1000 ver.5

Plan & Elaboration

Team No	Team2
과목	Software Modeling and Analysis
담당교수	JUNBEOM YOO Associate Professor / Ph.D
팀 구성원	201014184 김도윤
	201111367 여승훈
	201111347 김태호
제출일자	2016-06-01

Index

Activity1001. **Define Draft Plan**

Activity1002. **Create Preliminary Investigation Report**

Activity1003. **Define Requirements**

Activity1004. **Record Terms in Glossary**

Activity1005. **Implement Prototype**

Activity1006. **Define Draft System Architecture**

Activity1007. **Define Business Use Case**

Activity1008. **Define Business Concept Model**

Activity1009. **Define System Test Case**

Activity1010. **Refine Plan**

Activity 1001. Define Draft Plan

1. Motivation

소프트웨어의 개발이 활발해짐에 따라 소스코드의 복제, 도용이 증가하고 있다. 따라서 소스코드 복제 여부에 대한 검사 또한 매우 중요해지고 있다. 소스코드가 복제되면 기업, 개인 간의 분쟁이 일어날 수 있다. 따라서 사전에 소프트웨어 유사도 검사가 필요하다. 이것은 학교에서 과제로 진행되는 프로젝트에도 해당되며, 배움에 있어 학생들의 소스코드 복제 및 도용은 옳지 않다. 따라서 Clone Checker 프로그램을 개발하게 되었다.

2. Project Objectives

다수의 C 프로그램을 대상으로 상호 cheating 여부를 정량적으로 판단하고, 해당 내용을 다시 한 번 검토하여 결과를 수정할 수 있는 Clone Checker

- 파일 유사성의 단순 수치화를 통해 코드의 유사성을 직관적으로 판단할 수 있다.
- 각 파일들간의 유사성을 등급으로 표시하여 알려준다.
- 복제 및 도용이 의심되는 코드를 User가 다시 한번 확인하여 판단할 수 있다.

3. Project Scope

'One more chance'는 program이 산출한 결과에 의해 검사대상이 억울한 피해를 받을 수 있는 가능성이 존재한다는 사실에 착안하였다. 이러한 부분에 대해 다시 한 번 확인할 기회를 주는 것을 목표로 한다.

4. Functional requirements

- Load File
- Delete File
- Display File List
- Clone Check
- Check Global Value
- Check Operation
- Calculate Similarity
- Display File Set List
- Show All
- Show Red
- Show Orange
- Show Yellow
- Show Green
- Select File Set
- Display Main File
- Display Sub File
- Display Doubted Main File Operation
- Display Doubted Sub File Operation
- Display Similarity
- User Accept
- User Reject
- Save

5. Non-functional requirements

- 검사시간을 3min/50file 이내로 유지한다.

6. Resource Estimation

6.1 Human efforts (M/M)

3M/3M

6.2 Human resources

3명 - 프로그래머 3명

6.3 Duration

3개월

6.4 Budget

T.F Wage : $30 * 3\text{man} * 3\text{month} = 270$

7. Other Information

- Future Version

향상된 Performance, 한번이라도 Clone Checker 프로그램을 통해 객체화된 File은 정보가 살아있어 다음 Clone Check에 이용된다.

Activity 1002. Create Preliminary Investigation Report

1. Alternative solution

- 1.1 시중에 판매하는 Clone Check Program을 구매한다.
- 1.2 인력을 동원한다.
- 1.3 개발업체에 제작을 의뢰한다.

2. Project justification

- 1) Cost : 2,700,000₩
- 2) Duration : 3 Month
- 3) Risk : OOPT에 대한 이해력 부족. Clone Checker에 대한 지식 부족, 과제 Stack Overflow, 벚꽃사랑. 팀원의 휴학
- 4) Effect : 프로그램 유지 보수가 수월하다. 양산되는 과제물에 대한 Check가 용이하다.
비용 측면에서 저렴하다.

3. Risk & Risk reduction plan

3.1 Risk Management

Risk	Probability	Significance	Weight
Un-Skilled OOPT	3	5	15
Lack of knowledge Clone Checker	2	4	8
Assignment Stack Overflow	5	2	10
Spring Love	1	5	5
Absence	1	10	10

3.2 Risk Reduction Plan

Risk	Reduction Plan
Un-Skilled OOPT	교수님 혹은 강의조교에게 자문을 구한다.
Lack of knowledge Clone Checker	Clone Checker 구동 방식에 대해 공부한다.
Assignment Stack Overflow	Time Sharing 기법을 활용하여 동시에 처리한다.
Spring Love	합숙 프로젝트를 진행한다. (가능성 배제)
Absence	설득한다

4. Market Analysis

- Alphago에 사회적 issue화에 의해 증가할 것으로 전망되는 Computer Engineering 시장.

5. Managerial issue

- 2016년 5월까지 proto-type 개발이 완료되어야 한다.
- 2016년 7월까지 Final version 개발이 완료되어야 한다.

Activity 1003. Define Requirements

1. Functional Requirements

Function	Description
Load File	검사할 File을 List에 Load한다.
Delete File	List에 올라와있던 File를 List에서 삭제한다.
Display File List	Load된 File의 List를 보여준다.
Clone Check	Load된 모든 File간의 Clone 여부를 Detect 한다.
Check Global Value	모든 Code간의 Global Value 유사도 검사를 실행한다.
Check Operation	모든 Code간의 Operation 유사도 검사를 실행한다.
Calculate Similarity	모든 Code간의 유사성을 계산한다.
Display File Set List	File Set List를 display한다.
Show All	Clone Check 후의 모든 File Set의 유사성을 표시한다.
Show Red	Clone Check 후 두 파일간 유사성이 70% 이상인 File Set을 표시한다.
Show Orange	Clone Check 후 두 파일간 유사성이 50% 이상, 70% 미만인 File Set을 표시한다
Show Yellow	Clone Check 후 두 파일간 유사성이 30% 이상, 50% 미만인 File Set을 표시한다
Show Green	Clone Check 후 두 파일간 유사성이 30% 미만인 File Set을 표시한다
Select File Set	Clone Check가 끝난 File List상의 File Set을 선택한다.
Display Main File	List에서 선택한 비교항목의 Main File를 display한다.
Display Sub File	List에서 선택한 비교항목의 Sub File를 display한다.
Display Doubted Main File Operation	List에서 선택한 비교항목의 Main File operation 중에서 clone 이 의심되는 operation 부분을 display한다.
Display Doubted Sub File Operation	List에서 선택한 비교항목의 Sub File operation 중에서 clone 이 의심되는 operation 부분을 display한다.
Display Similarity	선택한 두 File의 Similarity를 Progress Bar에 Display한다.
User Accept	User가 선택한 두 File의 Clone이 의심되는 구간을 확인하고 Clone으로 판단한다.
User Reject	User가 선택한 두 File의 Clone이 의심되는 구간을 확인하고 Clone이 아니라고 판단한다.

Save	User가 검토한 정보를 저장하여 File Set List에 적용한다.
------	---

2. Functional Requirements (Categorized Table)

Reference No.	Function	Category
R1.1	Load File	Evident
R1.2	Delete File	Evident
R1.3	Display File List	Hidden
R2	Clone Check	Evident
R2.1	Check Global Value	Hidden
R2.2	Check Operation	Hidden
R2.3	Calculate Similarity	Hidden
R2.4	Display File Set List	Hidden
R3.1	Show All	Evident
R3.2	Show Red	Evident
R3.3	Show Orange	Evident
R3.4	Show Yellow	Evident
R3.5	Show Green	Evident
R4	Select File Set	Evident
R4.1	Display Main File	Hidden
R4.2	Display Sub File	Hidden
R4.3	Display Doubted Main File Operation	Hidden
R4.4	Display Doubted Sub File Operation	Hidden
R4.5	Display Similarity	Hidden
R5.1	User Accept	Evident
R5.2	User Reject	Evident
R6	Save	Evident

3. Performance Requirements

- 검사시간은 3min/50File 이내로 유지되어야 한다.

4. Operating Environment

Source	Case1	Case2	Case3
Operating Software	Win7 – 64bit	Win8 – 64bit	Mac OS – 64bit
CPU	i3-4005U	i5-4200U	i5-4200U
RAM	8GB	4GB	8GB
Library	128SSD	128GB SSD	128GB SSD

Develop Environment: Eclipse LUNA/MARS/KEPLER

5. Interface Requirement

- 유사도 percent에 따른 등급을 보기 좋게 표현해야 한다.
- 유사도 등급 기준에 따라 File Set이 정확하게 분리되어야 한다.
- Clone이 의심되는 부분을 출력하여 User가 판단할 수 있도록 한다.
- User가 유사도 그래프를 확인하여 직관적인 관찰이 가능하도록 한다.

6. Other Requirements

- N/A

Activity 1004. Record Terms in Glossary

Term	Description	Remarks
User	Clone Checker 프로그램을 사용하여 File 여부를 판단하는 객체	
Global Value	File 에서 선언된 Global Value 의 객체	
Operation	File 에서 선언된 Operation 의 객체	
Local Value	File 의 Operation 에서 선언된 Local Value 객체	
Branch	File 에서 선언된 Branch 의 객체	
File	User 가 Load 한 File 객체	
File Set	검사가 끝난 File Set 객체	
Operation Set	File Set 에서 의심되는 Operation Set 객체	
File Manager	File 을 관리하는 객체	
Clone Checker	File 의 유사도를 계산하는 객체	
Main Controller	프로그램을 제어하는 객체	

Activity 1005. Implement Prototype

The interface displays the following components:

- File List:** A vertical list of files (A.c, B.c, C.c, D.c, E.c, F.c) with 'ADD' and 'DEL' buttons.
- Check List:** A list of code ranges with their similarity percentages: A - B (76%), D - G (81%), G - L (91%), N - O (77%), O - Q (77%). Includes color-coded indicators (red, orange, yellow, green) and an 'ALL' button.
- Code G.c:**

```
#include<stdio.h>
void main(){
printf("Hello WorldWn");
print();
}
void print(){
printf("Bye WorldWn");
}
```
- Code L.c:**

```
#include<stdio.h>
int main(){
printf("Hello WorldWn");
printBye();
return 0;
}
int printBye(){
printf("Bye WorldWn");
return 1;
}
```
- print() and printBye() snippets:**

```
void print(){
printf("Bye WorldWn");
}
int printBye(){
printf("Bye WorldWn");
return 1;
}
```
- Progress and Action:** A red progress bar shows 91% completion. Below it are 'Accept', 'Reject', and 'Save' buttons.

Activity 1006. Define Draft System Architecture



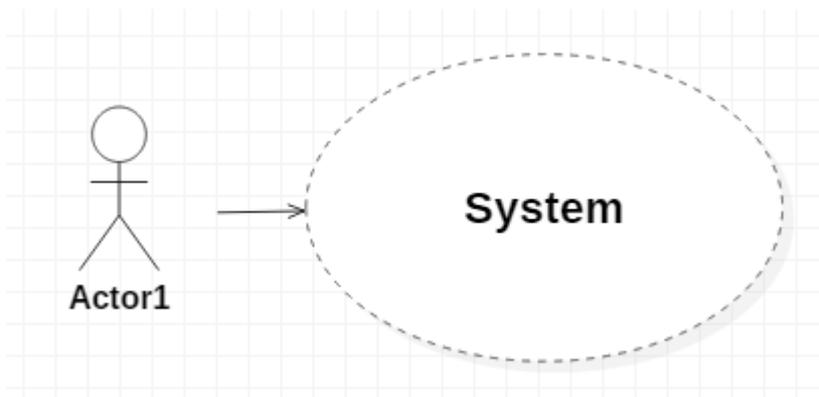
User



System

Activity 1007. Define Business Use Case

1. Define system boundary

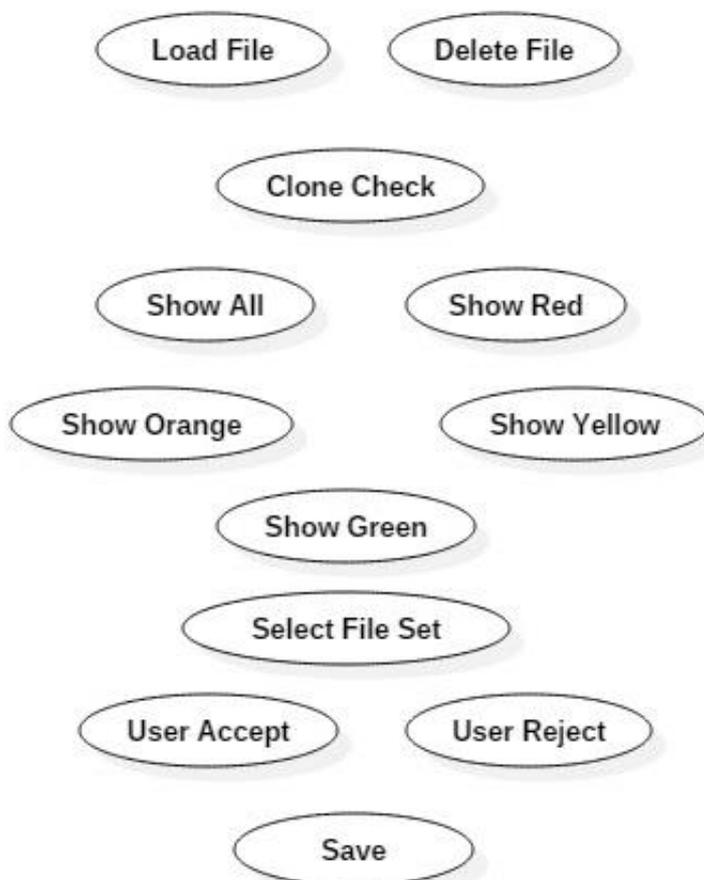


2. Identify and describe actors

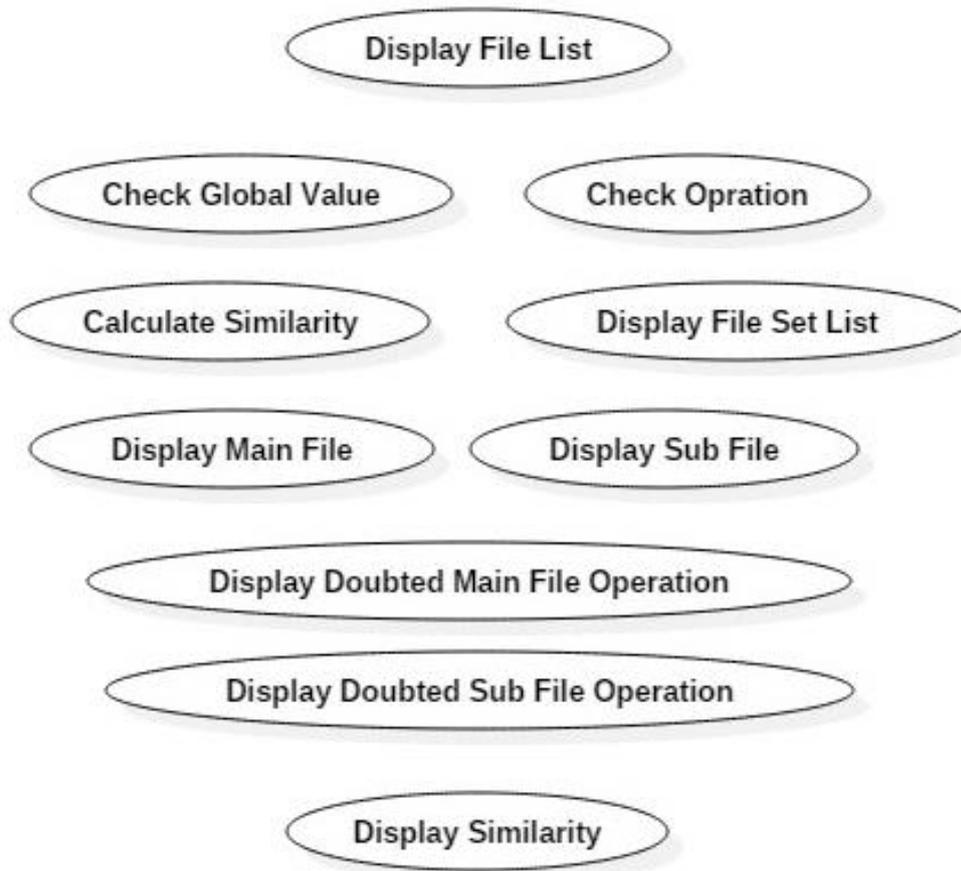
- User : Duplicate Code를 Detect하는 대상.

3. Identify use cases

3.1 Actor-Based



3.2 Event-Based



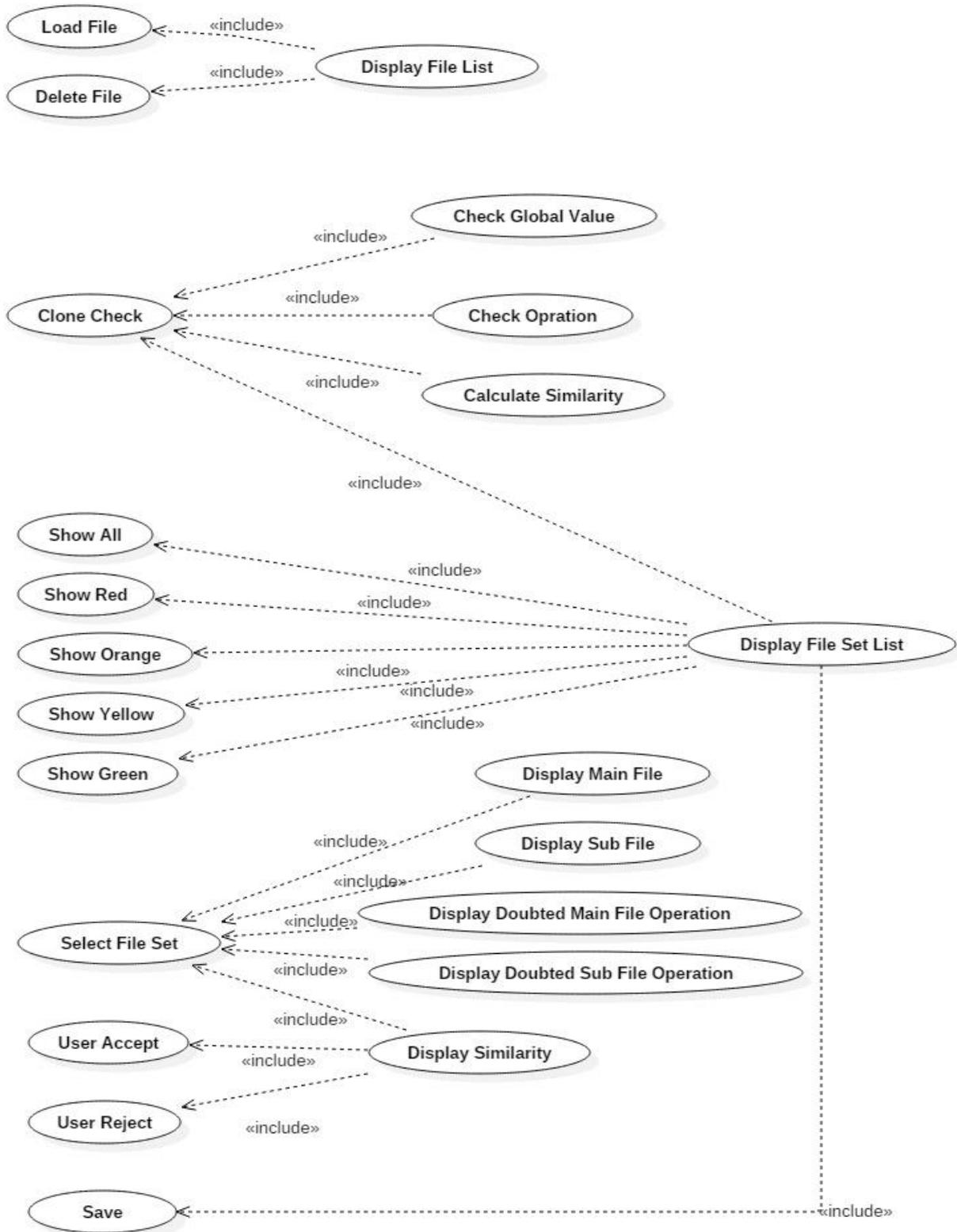
4. Allocate system functions into related use cases

Ref. #.	Function	Use Case Number & Name	Category
R1.1	Load File	1. Load File	
R1.2	Delete File	2. Delete File	
R1.3	Display File List	3. Display File List	
R2	Clone Check	4. Clone Check	
R2.1	Check Global Value	8. Check Global Value	
R2.2	Check Operation	9. Check Operation	
R2.3	Calculate Similarity	10. Calculate Similarity	
R2.4	Display File Set List	11. Display File Set List	
R3.1	Show All	12. Show All	
R3.2	Show Red	13. Show Red	
R3.3	Show Orange	14. Show Orange	
R3.4	Show Yellow	15. Show Yellow	
R3.5	Show Green	16. Show Green	
R4	Select File Set	17. Select File Set	
R4.1	Display Main File	18. Display Main File	
R4.2	Display Sub File	19. Display Sub File	
R4.3	Display Doubted Main File Operation	20. Display Doubted Main File Operation	
R4.4	Display Doubted Sub File Operation	21. Display Doubted Sub File Operation	
R4.5	Display Similarity	22. Display Similarity	
R5.1	User Accept	23. User Accept	
R5.2	User Reject	24. User Reject	
R6	Save	25. Save	

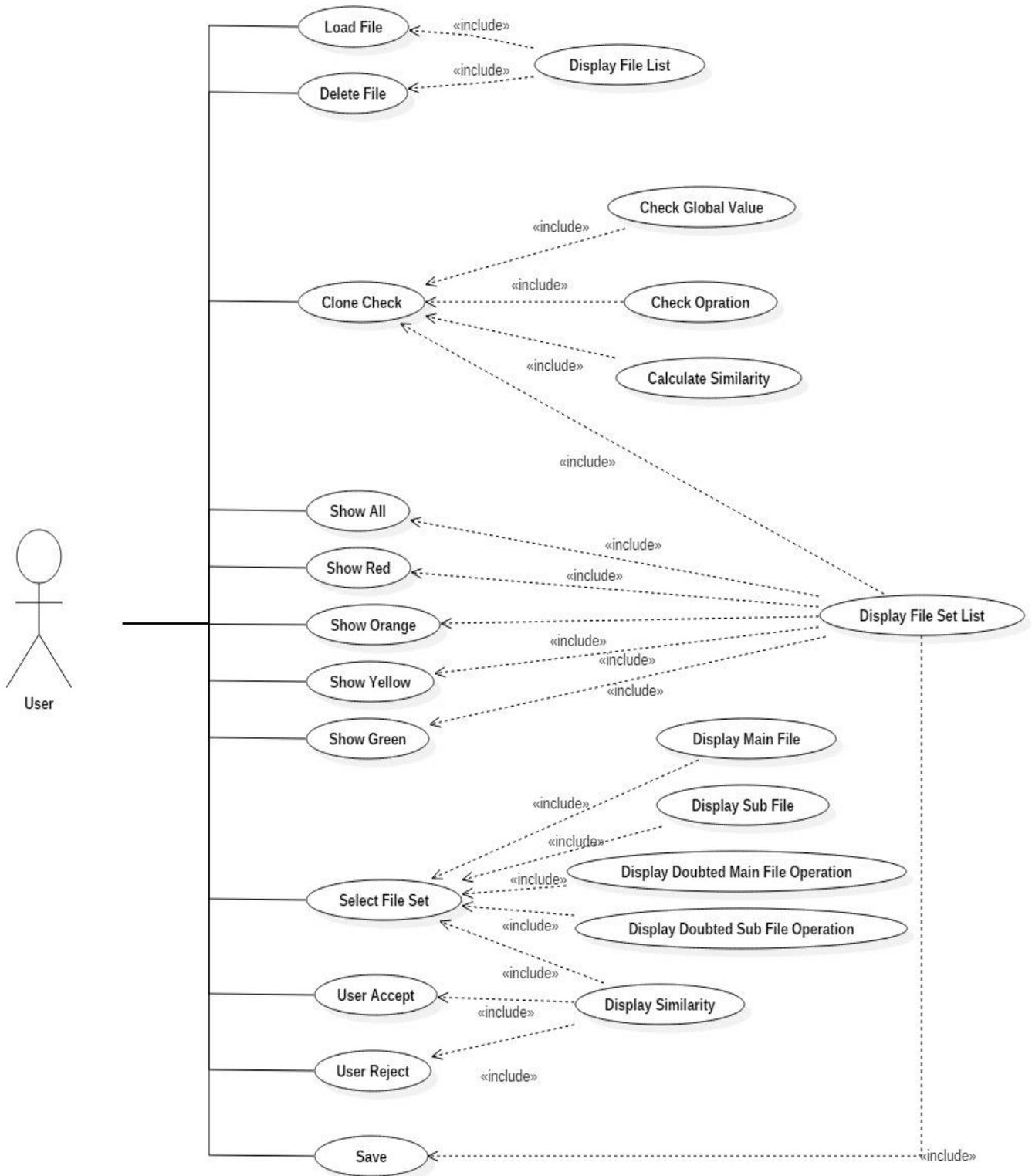
5. Categorize use cases

Ref. #.	Function	Use Case Number & Name	Category
R1.1	Load File	1. Load File	Primary
R1.2	Delete File	2. Delete File	Primary
R1.3	Display File List	3. Display File List	Primary
R2	Clone Check	4. Clone Check	Primary
R2.1	Check Global Value	8. Check Global Value	Primary
R2.2	Check Operation	9. Check Operation	Primary
R2.3	Calculate Similarity	10. Calculate Similarity	Primary
R2.4	Display File Set List	11. Display File Set List	Primary
R3.1	Show All	12. Show All	Primary
R3.2	Show Red	13. Show Red	Primary
R3.3	Show Orange	14. Show Orange	Primary
R3.4	Show Yellow	15. Show Yellow	Primary
R3.5	Show Green	16. Show Green	Primary
R4	Select File Set	17. Select File Set	Primary
R4.1	Display Main File	18. Display Main File	Primary
R4.2	Display Sub File	19. Display Sub File	Primary
R4.3	Display Doubted Main File Operation	20. Display Doubted Main File Operation	Primary
R4.4	Display Doubted Sub File Operation	21. Display Doubted Sub File Operation	Primary
R4.5	Display Similarity	22. Display Similarity	Primary
R5.1	User Accept	23. User Accept	Primary
R5.2	User Reject	24. User Reject	Primary
R6	Save	25. Save	Primary

6. Identify relationships between use cases



7. Draw a use case diagram



8. Describe use cases

Use Case	1. Load File
Actors	User
Description	검사할 File을 List에 Load한다.

Use Case	2. Delete File
Actors	User
Description	List에 올라와있던 File을 List에서 삭제한다.

Use Case	3. Display File List
Actors	None
Description	Load된 File의 List를 보여준다.

Use Case	4. Clone Check
Actors	User
Description	Load된 모든 File간의 Clone 여부를 Detect 한다.

Use Case	5. Check Global Value
Actors	None
Description	모든 Code간의 Global Value 유사도 검사를 실행한다.

Use Case	6. Check Operation
Actors	None
Description	모든 Code간의 Operation 유사도 검사를 실행한다.

Use Case	7. Calculate Similarity
Actors	None
Description	모든 Code간의 유사성을 계산한다.

Use Case	8. Display File Set List
Actors	None
Description	File Set List를 Display한다.

Use Case	9. Show All
Actors	User
Description	Clone Check 후의 모든 File Set의 유사성을 표시한다.

Use Case	10. Show Red
Actors	User
Description	Clone Check 후 두 파일간 유사성이 70% 이상인 File Set을 표시한다.

Use Case	11. Show Orange
Actors	User
Description	Clone Check 후 두 파일간 유사성이 50% 이상, 70% 미만인 File Set을 표시한다

Use Case	12. Show Yellow
Actors	User
Description	Clone Check 후 두 파일간 유사성이 30% 이상, 50% 미만인 File Set을 표시한다.

Use Case	13. Show Green
Actors	User
Description	Clone Check 후 두 파일간 유사성이 30% 미만인 File Set을 표시한다

Use Case	14. Select File Set
Actors	User
Description	Clone Check가 끝난 File List상의 File Set을 선택한다.

Use Case	15. Display Main File
Actors	None
Description	List에서 선택한 비교항목의 Main File를 display한다.

Use Case	16. Display Sub File
Actors	None
Description	List에서 선택한 비교항목의 Sub File를 display한다.

Use Case	17. Display Doubted Main File Operation
Actors	None
Description	List에서 선택한 비교항목의 Main File operation 중에서 clone이 의심되는 operation 부분을 display한다.

Use Case	18. Display Doubted Sub File Operation
Actors	None
Description	List에서 선택한 비교항목의 Sub File operation 중에서 clone이 의심되는 operation 부분을 display한다.

Use Case	19. Display Similarity
Actors	None
Description	선택한 두 File의 Similarity를 Progress Bar에 display한다.

Use Case	20. User Accept
Actors	User
Description	User가 선택한 두 File의 Clone이 의심되는 구간을 확인하고 Clone으로 판단한다.

Use Case	21. User Reject
Actors	User
Description	User가 선택한 두 File의 Clone이 의심되는 구간을 확인하고 Clone이 아니라고 판단한다.

Use Case	22. Save
Actors	User
Description	User가 검토한 정보를 저장하여 File Set List에 적용한다.

9. Rank use cases

Ref. #.	Function	Use Case Number & Name	Category	Rank
R1.1	Load File	1. Load File	Primary	High
R1.2	Delete File	2. Delete File	Primary	High
R1.3	Display File List	3. Display File List	Primary	High
R2	Clone Check	4. Clone Check	Primary	High
R2.1	Check Global Value	8. Check Global Value	Primary	High
R2.2	Check Operation	9. Check Operation	Primary	High
R2.3	Calculate Similarity	10. Calculate Similarity	Primary	High
R2.4	Display File Set List	11. Display File Set List	Primary	High
R3.1	Show All	12. Show All	Primary	High
R3.2	Show Red	13. Show Red	Primary	High
R3.3	Show Orange	14. Show Orange	Primary	High
R3.4	Show Yellow	15. Show Yellow	Primary	High
R3.5	Show Green	16. Show Green	Primary	High
R4	Select File Set	17. Select File Set	Primary	High
R4.1	Display Main File	18. Display Main File	Primary	High
R4.2	Display Sub File	19. Display Sub File	Primary	High
R4.3	Display Doubted Main File Operation	20. Display Doubted Main File Operation	Primary	High
R4.4	Display Doubted Sub File Operation	21. Display Doubted Sub File Operation	Primary	High
R4.5	Display Similarity	22. Display Similarity	Primary	High
R5.1	User Accept	23. User Accept	Primary	High
R5.2	User Reject	24. User Reject	Primary	High
R6	Save	25. Save	Primary	High

Activity 1008. Define Business Concept Model

Clone Check

통합 검사

File List

간단한 조작

검토 및 검토
결과 반영

Progress Bar

UI

Activity 1009. Define System Test Case

Test Number	Test 항목	Description	Use-Case	System Function
1	Load 버튼 시험	User가 Load한 File이 List에 Load 되는지 확인한다.	1. Load File	R1.1
2	Delete 버튼 시험	User가 Delete한 File이 List에서 Delete되는지 확인한다.	2. Delete File	R1.2
3	Display 시험	User가 Load한 List가 Display 되는지 확인한다.	3. Display File List	R1.3
4	Clone Check 버튼 시험	Load된 모든 File간의 Clone Check 가 수행되는지 확인한다.	4. Clone Check	R2
5	Check Global Value 시험	Code들간의 Global Value 유사도 검사가 수행되는지 확인한다.	8. Check Global Value	R2.1
6	Check Operation 시험	Code들간의 Operation 유사도 검사가 수행되는지 확인한다.	9. Check Operation	R2.2
7	Calculate Similarity 시험	Code간의 유사성이 계산되는지 확인한다.	10. Calculate Similarity	R2.3
8	File Set Display 시험	File Set List들이 Display 되는지 확인한다.	11. Display File Set List	R2.4
9	All 버튼 시험	모든 File Set List가 Display 되는지 확인한다.	12. Show All	R3.1
10	Red 버튼 시험	파일간 유사성이 70% 이상인 File Set이 Display 되는지 확인한다.	13. Show Red	R3.2
11	Orange 버튼 시험	파일간 유사성이 50% 이상, 70% 미만인 File Set이 Display 되는지 확인한다	14. Show Orange	R3.3
12	Yellow 버튼 시험	파일간 유사성이 30% 이상, 50% 미만인 File Set이 Display 되는지 확인한다	15. Show Yellow	R3.4
13	Green 버튼 시험	파일간 유사성이 30% 미만인 File Set이 Display 되는지 확인한다	16. Show Green	R3.5

14	File Set Select 시험	Clone Check가 끝난 File List상에서 File Set이. 선택되는지 확인한다.	17. Select File Set	R4
15	Main File Display 시험	List에서 선택한 비교항목의 Main File이 Display 되는지 확인한다.	18. Display Main File	R4.1
16	Sub File Display 시험	List에서 선택한 비교항목의 Sub File Display 되는지 확인한다.	19. Display Sub File	R4.2
17	Doubted Main File Operation Display 시험	List에서 선택한 비교항목의 Main File operation 중에서 clone이 의심되는 operation 부분이 Display 되는지 확인한다.	20. Display Doubted Main File Operation	R4.3
18	Doubted Sub File Operation Display 시험	List에서 선택한 비교항목의 Sub File operation 중에서 clone이 의심되는 operation 부분이 Display 되는지 확인한다.	21. Display Doubted Sub File Operation	R4.4
19	Progress Bar Display 시험	선택한 두 File의 Similarity가 Progress Bar에 Display 되는지 확인한다.	22. Display Similarity	R4.5
20	Accept 버튼 시험	User Accept 명령이 정확하게 전달되는지 확인한다.	23. User Accept	R5.1
21	Reject 버튼 시험	User의 Reject 명령이 정확하게 전달되는지 확인한다.	24. User Reject	R5.2
22	Save 버튼 시험	User의 Save 명령이 정확하게 전달되는지 확인한다.	25. Save	R6

Activity 1010. Refine Plan

1. Project Scope

소스코드의 복제 및 도용으로 인한 문제를 방지하기 위해 소스코드간의 Cloning 여부를 판단한다.

2. Project Objectives

다수의 C 프로그램을 대상으로 상호 cheating 여부를 정량적으로 판단하고, 해당 내용을 User가 검토하여 결과를 수정할 수 있는 Clone Checker를 개발한다.

3. Functional Requirements

- Load File
- Delete File
- Display File List
- Clone Check
- Check Global Value
- Check Operation
- Calculate Similarity
- Display File Set List
- Show All
- Show Red
- Show Orange
- Show Yellow
- Show Green
- Select File Set
- Display Main File
- Display Sub File
- Display Doubted Main File Operation
- Display Doubted Sub File Operation

- Display Similarity
- User Accept
- User Reject
- Save

4. Performance Requirements

- 검사시간을 3min/50File 이내로 유지한다.

5. Operating Environment

Source	Case1	Case2	Case3
Operating Software	Win7 – 64bit	Win8 – 64bit	Mac OS – 64bit
CPU	i3-4005U	i5-4200U	i5-4200U
RAM	8GB	4GB	8GB
Library	128SSD	128GB SSD	128GB SSD

6. User Interface Requirements

- 유사도 percent에 따른 등급을 보기 좋게 표현해야 한다.
- 유사도 등급 기준에 따라 File Set이 정확하게 분리되어야 한다.
- Clone이 의심되는 부분을 출력하여 User가 판단할 수 있도록 한다.
- User가 유사도 그래프를 확인하여 직관적인 관찰이 가능하도록 한다.

7. Other Requirements

- N/A

8. Resources

8.1 Human efforts (M/M)

3M/3M

8.2 Human resources

3명 - 프로그래머 3명

8.3 Duration

3개월

8.4 Budget

₩2,700,000

9. Scheduling

Stage	Phase(00X0)/Activity(000X)	Schedule(Week)													
		1	2	3	4	5	6	7	8	9	10				
1000. Plan & Elaborate	1001. Define Draft Plan	■													
	1002. Create Preliminary Investigation Report	■	■												
	1003. Define Requirements		■												
	1004. Record Terms in Glossary			■											
	1005. Implement Prototype				■										
	1006. Define Business Use Case					■									
	1007. Define Business Concept Model						■								
	1008. Define Draft System Architecture							■							
	1009. Define System Test Case								■						
	1010. Refine Plan									■					
2000. Build	2110. Revise Plan														
	2120. Synchronize Artifacts														
	2130. Analyze														
	2131. Define Essential Use Case														
	2132. Refine Use Case Diagram														
	2133. Refine Conceptual Model														
	2134. Refine Glossary														
	2135. Define System Sequence Diagram														
	2136. Define Operation Contracts														
	2137. Define State Diagrams														
	2140. Design														
	2141. Define Real Use Case														
	2142. Define Reports, UI and Story boards														
	2143. Refine System Architecture														
	2144. Define Interaction Diagram														
	2145. Define Design Class Diagrams														
2150. Construct															
2151. Implement Class & Interface Definition															
2152. Implement Methods															
2153. Implement Windows															
2154. Implement Reports															
2155. Write Test Code															
2160. Test															
2161. Unit Testing															
2162. Integration Testing															
2163. System Testing															
2164. Performance Testing															
2165. Acceptance Testing															
2166. Documentation Testing															

Appendix. Manual

1 Performance Test Environment

- 1.1 CPU : i3-4005U
- 1.2 Memory : 4GB
- 1.3 CPU Rate: 80% 이하
- 1.4 RAM Rate: 50% 이하
- 1.5 CPU temperature: 60°C 이하
- 1.6 Window 7 OS 64-bit
- 1.7 Check 대상 File Size 30KB 이하

2 Calculate Formula

2.1 Calculate File Score

- 2.1.1 선언된 Header 개수 1개당 1점 (stdio.h, stdlib.h, time.h, windows.h, string.h, conio.h)
- 2.1.2 선언된 define 변수 개수 1개당 1점
- 2.1.3 선언된 struct나 enum 개수 1개당 1점
- 2.1.4 선언된 전역변수 개수 1개당 1점 (int, char, long, double, short, float, byte, bool, FILE)
- 2.1.5 선언된 함수의 개수 당 2점 (return type을 비교하기 위함)
- 2.1.6 선언된 함수 내 지역변수 개수 1개당 1점 (int, char, long, double, short, float, byte, bool, FILE)
- 2.1.7 선언된 함수 내 complexity n점 (해당 함수의 complexity가 5면 5점)
- 2.1.8 선언된 함수 내 library 함수 1개당 1점 (scanf, strstr, strcpy, strtok, strcmp, strcat, printf, malloc, sizeof, free, →)
- 2.1.9 함수의 경우 함수만의 점수를 가지고 있다.

2.2 Calculate Similarity Score

2.2.1 선언된 Header 중 같은 Header 선언 1개당 1점

2.2.2 선언된 전역 변수 중 type이 같은 전역 변수 1개당 1점

2.2.3 선언된 함수의 return type이 같으면 2점

2.2.4 선언된 함수 내 선언된 지역 변수 중 type이 같은 지역 변수 1개당 1점

2.2.5 선언된 함수 내 공통으로 사용된 library 개수 1개당 1점

2.2.6 선언된 함수 내 complexity n점

2.2.7 함수의 경우 함수의 유사도가 80% 이상 같은 경우 Score를 측정한다.

2.3 Calculate Similarity

2.3.1 File A와 File B의 비교 가정

2.3.2 File A의 Score가 50점, File B의 Score가 100점이라고 가정

2.3.3 File A와 File B의 유사도 Score가 40점이라고 가정

2.3.4 계산식 : $(\text{File A와 File B의 유사도 Score} / \text{File A와 File B 중 높은 Score}) * 100$

2.3.5 위의 유사도 : $(40 / 100) * 100 = 40\%$

2.4 Example

2.4.1 File A

2.4.1.1 선언된 Header : `stdio.h, stdlib.h, string.h`

2.4.1.2 선언된 struct : `NODE`

2.4.1.3 선언된 전역 변수 : `int a, char c`

2.4.1.4 선언된 함수 : `NODE A(); int main();`

2.4.1.5 `NODE A()` 함수 내 선언된 지역 변수 : `FILE fp, int b`

2.4.1.6 `NODE A()` 함수 complexity : 3

2.4.1.7 `NODE A()` 함수 내 선언된 library 개수 : `scanf` 3번, `strcmp` 4번, → 2번

2.4.1.8 NODE A() 함수의 Score : $2 + 3 + 9 = 14$

2.4.1.9 int main() 함수 내 선언된 지역 변수 : int d, int e, int f

2.4.1.10 int main() 함수 complexity : 40

2.4.1.11 int main() 함수 내 선언된 library 개수 : malloc 5번, sizeof 5번

2.4.1.12 int main() 함수의 Score : $3 + 40 + 10 = 53$

2.4.1.13 FILE A의 Score : $3 + 1 + 2 + 4 + 14 + 53 = 77$

2.4.2 File B

2.4.2.1 선언된 Header : stdio.h, string.h

2.4.2.2 선언된 define : TRUE, FALSE

2.4.2.3 선언된 struct : Node

2.4.2.4 선언된 전역 변수 : int b;

2.4.2.5 선언된 함수 Node B(); void init(); int main();

2.4.2.6 Node B() 함수 내 선언된 지역 변수 : FILE fp;

2.4.2.7 Node B() 함수 complexity : 2

2.4.2.8 Node B() 함수 내 선언된 library 개수 : scanf 3번, strcmp 3번, → 2번

2.4.2.9 Node B 함수의 Score : $1 + 2 + 8 = 11$

2.4.2.10 void init() 함수 complexity : 4

2.4.2.11 void init() 함수 내 선언된 library 개수 : scanf 3번, malloc 3번

2.4.2.12 void init() 함수의 Score : $4 + 6 = 10$

2.4.2.13 int main() 함수 내 선언된 지역 변수 : int c, int d;

2.4.2.14 int main() 함수 complexity : 35

2.4.2.15 int main() 함수 내 선언된 library 개수 : malloc 4번, sizeof 4번

2.4.2.16 int main() 함수의 Score : $2 + 35 + 8 = 45$

2.4.2.17 FILE B의 Score : $2 + 2 + 1 + 1 + 6 + 11 + 10 + 45 = 78$

2.4.3 Calculate Similarity Score

2.4.3.1 같은 Header `stdio.h`, `string.h` 2개

2.4.3.2 선언된 전역 변수 같은 type : `struct`, `int` 2개

2.4.3.3 NODE A() 함수와 Node B() 함수 return type이 `struct`로 같음

2.4.3.4 NODE A() 함수와 Node B() 함수 선언된 지역 변수 같은 type : `FILE` 1개

2.4.3.5 NODE A() 함수와 Node B() 함수 공통으로 사용된 library `scanf` 3개, `strcmp` 3개, → 2개

2.4.3.6 NODE A() 함수와 Node B() 함수 complexity 2점 (2개 중 작은 complexity)

2.4.3.7 NODE A() 함수와 Node B() 함수의 Score : $2 + 1 + 8 + 2 = 13$ (선택)

2.4.3.8 이런 식으로 모든 함수간 계산을 실시

2.4.3.9 NODE A() 함수와 `void init()` 함수의 Score : 6

2.4.3.10 NODE A() 함수와 `int main()` 함수의 Score : 4

2.4.3.11 `int main()` 함수와 Node B() 함수의 Score : 2

2.4.3.12 `int main()` 함수와 `void init()` 함수의 Score : 7

2.4.3.13 `int main()` 함수와 `int main()` 함수의 Score : 47점 (선택)

2.4.4 Calculate Similarity

2.4.4.1 Global Value Score : 4점

2.4.4.2 Operation Score : 60점

2.4.4.3 File A와 File B의 Score : 64점

2.4.4.4 File A와 File B 중 높은 Score : 78

2.4.4.5 Similarity : $64 / 78 * 100 = 82\%$