



세종사이버대학교

02

명세 기반 테스트



SEJONG CYBER UNIVERSITY



윤 동 환
세



2 명세 기반 테스트

☞ 명세 기반 테스트 개요

☑ 특징

- 프로그램의 내부 논리 구조를 참조하지 않고 사용자의 요구사항이 기술된 명세나 설계 정보 등을 이용하여 테스트 케이스 개발
- 컴포넌트 테스트, 통합 테스트, 시스템 테스트 및 인수 테스트 등 전 과정에 걸쳐 사용 가능



2 명세 기반 테스트

☞ 명세 기반 테스트 개요

☑ 대표적인 종류

- 동등 분할(Equivalence partitioning)
- 경계값 분석(Boundary Value Analysis)
- 페어와이즈 조합(Pairwise Combination)
- 결정 테이블(Decision Table)
- 상태 전이(State-transition)



2 명세 기반 테스트

동등 분할 (Equivalence partitioning)

✓ 개요

- 프로그램 입력이나 출력 영역을 동등 클래스 (Equivalent class)라 불리는 몇 개의 영역으로 분할
- 각 클래스에서 하나의 값을 선택하여 테스트 케이스로 사용
- 동등 클래스에서 선정한 값으로 프로그램이 올바르게 동작한다면 동등 클래스의 나머지 값들도 올바르게 동작할 것이라는 가정
- (주의점) 동등 클래스들은 서로 공통된 값이 없어야 함



2 명세 기반 테스트

동등 분할 (Equivalence partitioning)

✓ 절차

- 명세에서 입력과 출력을 식별
- 각 입력/출력 영역을 동등 클래스들로 분할
 - ✓ 이때 도메인 지식이나 과거 경험을 이용하여 입출력 영역 분할 가능
- 각 동등 클래스에서 최소한 하나의 대표 값을 선정하여 테스트 케이스에 반영한 다음, 테스트 케이스 테이블을 작성

2 명세 기반 테스트

동등 분할 (Equivalence partitioning) 예시

명세)

- 나이가 10세 이하이면 입장을 할 수 없고,
- 나이가 10~15세이면 보호자가 동반해야 입장이 가능하고,
- 나이가 15세가 넘으면 혼자서도 입장이 가능
- 또한, 80세가 넘는 경우에도 보호자가 동반해야 입장 가능
- 나이가 0세 이하이거나 정수가 아니면 "Invalid input" 메시지 출력
- 나이가 100세를 넘는 경우에는 "Too old" 메시지 출력



2 명세 기반 테스트

동등 분할 (Equivalence partitioning) 예시

	유효 분할	유효하지 않은 분할
입력	$0 < \text{나이} \leq 100$	$\text{나이} \leq 0$ $\text{나이} > 100$ $\text{나이} = \text{문자열}$ $\text{나이} = \text{실수}$
출력	“입장 불가”: $(0 < \text{나이} \leq 10)$ “보호자 동반 입장”: $(10 < \text{나이} \leq 15), (80 < \text{나이} \leq 100)$ “입장 가능”: $(16 < \text{나이} \leq 80)$ “Invalid input”: $(\text{나이} \leq 0), (\text{나이} = \text{non-integer})$ “Too old”: $(\text{나이} > 100)$	메시지 $\neq \{ \text{“입장 불가”, “보호자 동반 입장”, “입장 가능”, “Invalid input”, “Too old”} \}$ \Rightarrow “노인 할인 입장” $(60 < \text{나이} \leq 100)$

2 명세 기반 테스트

동등 분할 (Equivalence partitioning) 예시

학습하기

테스트 케이스	입력(나이)	예상출력	동등 클래스
1	50	"입장 가능"	$0 < \text{나이} \leq 100$
2	-10	"Invalid input"	$\text{나이} \leq 0$
3	105	"Too old"	$100 < \text{나이}$
4	"abc"	"Invalid input"	문자열
5	5	"입장 불가"	$0 < \text{나이} \leq 10$
6	13	"보호자 동반 입장"	$10 < \text{나이} \leq 15$

2 명세 기반 테스트

경계값 분석(Boundary Value Analysis)

✓ 개요

- 소프트웨어 결함은 보통 입력 영역의 경계에서 발생
- 입력 영역 경계 근처에 있는 값들을 이용하여 테스트 케이스를 설계하는 테스트 방법
- 동등 분할과 마찬가지로 입력/출력 영역을 여러 클래스로 분할
- 클래스의 경계와 경계 근처에 있는 값들을 사용하여 테스트 케이스 설계



2 명세 기반 테스트



세종사이버대학교

💬 경계값 분석 (Boundary Value Analysis)

학습하기

✅ 절차

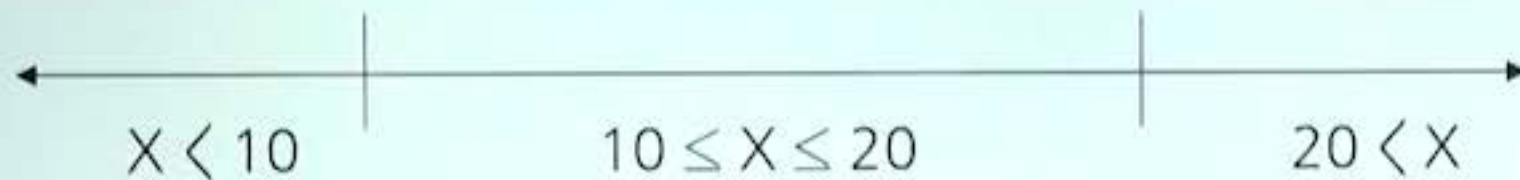
- 명세에서 입력/출력들을 식별
- 각 입력/출력에 대한 동등 분할을 수행
- 각 분할된 클래스의 경계값을 식별
- 2-value BVA나 3-value BVA에 따라 경계값 분석을 수행
- 위 결과로 얻은 각 값에 대해 기대 출력을 명세로 구분하여 테스트 케이스 설계



SEJONG CYBER UNIVERSITY

2 명세 기반 테스트

경계값 분석 (Boundary Value Analysis) 예시

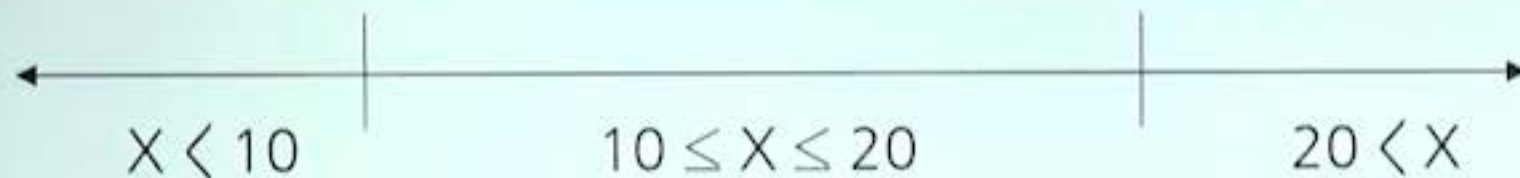


[2 Value BVA]

ID	입력(X)	경계
1	9, 10	9
2	9, 10	10
3	20, 21	20
4	20, 21	21
5	32767, 32768	32767
6	-32768, -32769	-32768

2 명세 기반 테스트

경계값 분석 (Boundary Value Analysis) 예시



[3 Value BVA]

ID	입력(X)	경계
1	8, 9, 10	9
2	9, 10, 11	10
3	19, 20, 21	20
4	20, 21, 22	21
5	32766, 32767, 32768	32767
6	-32767, -32768, -32769	-32768

2 명세 기반 테스트



세종사이버대학교

💬 경계값 분석(Boundary Value Analysis)



학습하기

✅ 경계값 선택 지침

- 입력 혹은 사전조건이 값의 범위로 주어지는 경우
 - 범위의 끝에 속하는 유효한 입력 값
 - 범위의 바로 바깥에 속하는 유효하지 않은 입력 값
- 입력조건이 몇 개의 값으로 주어지는 경우
 - 입력 값의 최소값과 최대값
 - 최소값과 최대값의 바로 아래와 바로 위의 값
- 입력조건이 파일, 리스트, 테이블과 같은 정렬된 집합 형태
 - 첫 번째 항목과 마지막 항목 선택



SEJONG CYBER UNIVERSITY

2 명세 기반 테스트



세종사이버대학교

☞ 페어와이즈 조합

☑ 개요

- 입력들의 모든 가능한 조합들을 테스트하는 대신, 모든 입력 값의 모든 짝(Pair) 조합을 테스트하는 방법
- 모든 입력에 대해 존재할 수 있는 모든 상호 작용을 고려하지 않고, 모든 두 개의 입력 간에 가능한 모든 상호 작용만을 고려
- 모든 입력 값들의 조합과 비슷한 테스트링 효과 가능



SEJONG CYBER UNIVERSITY

2 명세 기반 테스트

☞ 페어와이즈 조합

학습하기

☑ 수행 방법

- 입력 값의 모든 짝(pair)들의 조합이 테스트 케이스 내에 적어도 한 번 이상 들어가도록 테스트 케이스를 도출
- 예시

Machine	Web Browser	Connection Type
Machine 1	Edge	LAN
Machine 2	Chrome	Bluetooth

➡ (Machine, Web Browser)
(Machine, Connection Type)
(Web Browser, Connection Type)

2 명세 기반 테스트

☞ 페어와이즈 조합

[모든 짝들의 조합]

Machine	Web Browser	Machine	Connection Type
Machine 1	Edge	Machine 1	LAN
Machine 1	Chrome	Machine 1	Bluetooth
Machine 2	Edge	Machine 2	LAN
Machine 2	Chrome	Machine 2	Bluetooth

Web Brower	Connection Type
Edge	LAN
Edge	Bluetooth
Chrome	LAN
Chrome	Bluetooth

2 명세 기반 테스트

☞ 페어와이즈 조합

학습하기

[테스트 케이스]

	Machine	Web Browser	Connection Type
TC1	Machine 1	Edge	LAN
TC2	Machine 1	Chrome	Bluetooth
TC3	Machine 2	Edge	Bluetooth
TC4	Machine 2	Chrome	LAN

(전체 조합) 테스트 케이스 수 = $2 \times 2 \times 2 = 8$

(페어와이즈) 테스트 케이스 수 = 4

2 명세 기반 테스트

결정 테이블

✓ 개요

- 결정표를 이용하여 테스트 케이스를 설계하는 방법

✓ 절차

- 명세 등을 분석하여 모든 조건을 분석
- 모든 조건의 조합에 대한 행위를 결정
- 앞 두 단계를 통해 결정표를 작성
- 가능하지 못한 조건의 조합은 배제
- 결정표를 축약할 수 있는지 파악
- 결정표의 각 규칙이 최소 1번 테스트 될 수 있도록 TC 생성

2 명세 기반 테스트



세종사이버대학교

💬 결정 테이블 - 예시

(명세)

- H 대학교에서는 성적이 C 이하인 학생들을 대상으로 여러 학습 상담 프로그램을 운영
- 성적이 B 이상인 학생들은 학습 상담 프로그램 이수하지 않아도 됨
- 성적이 C 이하인 학생들은 결석 일수가 3일 이상
 - A 학습 상담 프로그램
- 또는 결석 일수가 2일 이하
 - B 학습 상담 프로그램
- 또한, 1학년은 C학습 프로그램을 동시에 받도록 함



SEJONG CYBER UNIVERSITY

2 명세 기반 테스트



세종사이버대학교

💬 결정 테이블 - 예시



학습하기

(명세)

- H 대학교에서는 성적이 C 이하인 학생들을 대상으로 여러 학습 상담 프로그램을 운영
- 성적이 B 이상인 학생들은 학습 상담 프로그램 이수하지 않아도 됨
- 성적이 C 이하인 학생들은 결석 일수가 3일 이상
 - A 학습 상담 프로그램
- 또는 결석 일수가 2일 이하
 - B 학습 상담 프로그램
- 또한, 1학년은 C학습 프로그램을 동시에 받도록 함



SEJONG CYBER UNIVERSITY

2 명세 기반 테스트

결정 테이블 - 예시

		규칙 (조건들의 모든 가능한 조합)							
		1	2	3	4	5	6	7	8
조건	성적 C 이하	Y	Y	Y	Y	N	N	N	N
	결석 일수 3일 이상	Y	Y	N	N	Y	Y	N	N
	1학년	Y	N	Y	N	Y	N	Y	N
행위	A 상담 프로그램	Y	Y	-	-	-	-	-	-
	B 상담 프로그램	-	-	Y	Y	-	-	-	-
	C 상담 프로그램	Y	-	Y	F	-	-	-	-

2 명세 기반 테스트



세종사이버대학교

결정 테이블 - 예시



학습하기

		규칙 (축약된 조건)				
		1	2	3	4	5
조건	성적 C 이하	Y	Y	Y	Y	N
	결석 일수 3일 이상	Y	Y	N	N	Y
	1학년	Y	N	Y	N	Y
행위	A 상담 프로그램	Y	Y	-	-	-
	B 상담 프로그램	-	-	Y	Y	-
	C 상담 프로그램	Y	-	Y	F	-



SEJONG CYBER UNIVERSITY

2 명세 기반 테스트

결정 테이블 - 예시

학습하기

		규칙 (조건들의 모든 가능한 조합)							
		1	2	3	4	5	6	7	8
조건	성적 C 이하	Y	Y	Y	Y	N	N	N	N
	결석 일수 3일 이상	Y	Y	N	N	Y	Y	N	N
	1학년	Y	N	Y	N	Y	N	Y	N
행위	A 상담 프로그램	Y	Y	-	-	-	-	-	-
	B 상담 프로그램	-	-	Y	Y	-	-	-	-
	C 상담 프로그램	Y	-	Y	F	-	-	-	-

2 명세 기반 테스트

결정 테이블 - 예시

		규칙 (축약된 조건)				
		1	2	3	4	5
조건	성적 C 이하	Y	Y	Y	Y	N
	결석 일수 3일 이상	Y	Y	N	N	Y
	1학년	Y	N	Y	N	Y
행위	A 상담 프로그램	Y	Y	-	-	-
	B 상담 프로그램	-	-	Y	Y	-
	C 상담 프로그램	Y	-	Y	F	-

2 명세 기반 테스트

결정 테이블 - 예시

학습하기

[테스트 케이스]

	입력			기대 출력
	학점	결석일수	학년	
TC1	D	3	1	A/C 상담
TC2	C+	4	3	A 상담
TC3	C	1	1	B/C 상담
TC4	D+	2	2	B 상담
TC5	B+	3	4	해당 없음

2 명세 기반 테스트

상태 전이

학습하기

✓ 개요

- 시스템을 상태 전이도(State-transition diagram)로 모델링한 후 테스트 케이스를 상태 전이도에서 체계적으로 선정하는 방법

✓ 절차

- 테스트하려고 하는 프로그램의 명세를 상태 전이도를 사용하여 모델링
- 상태 전이도에서 전이 트리를 생성
- 전이 트리에서 각 전이 경로를 테스트하는 TC 생성
- 유효하지 않은 전이를 테스트하기 위한 TC 생성

2 명세 기반 테스트



세종사이버대학교

💬 상태 전이 - 예시



학습하기

(명세)

- 테이프 재생기 기능
 - 재생(Play) / 빠른 전진 이동(Fast forward) ³
 - 빠른 재생(Fast play) ✓
- 재생과 빠른 전진 이동은 각각 재생과 빠른 전진 이동 버튼을 통해 기능이 활성화되며 정지(Stop) 버튼을 사용하여 취소될 수 있음
- 재생 모드에 있을 때, 빠른 재생을 위해 빠른 전진 이동 버튼을 사용
- 빠른 재생 모드에 있을 때는 빠른 전진 이동 버튼을 사용하여 빠른 전진 이동 모드로 전환하거나
- 정지 버튼을 사용하여 재생 모드로 돌아갈 수 있음
- 빠른 전진 이동 모드에 있을 때는 재생 버튼을 사용하여 재생 모드로 바로 갈 수 있음



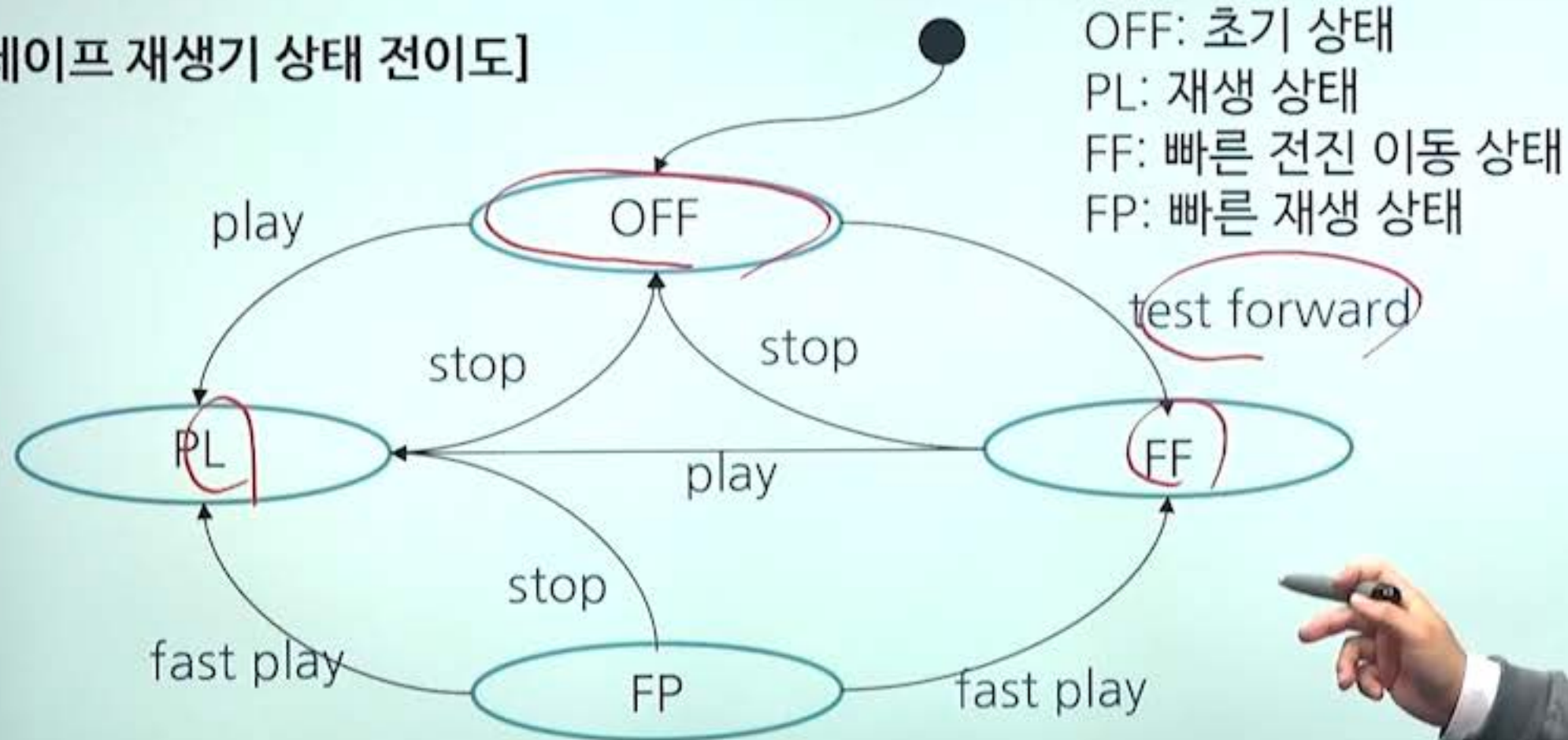
SEJONG CYBER UNIVERSITY

2 명세 기반 테스트

상태 전이 - 예시

[테이프 재생기 상태 전이도]

학습하기

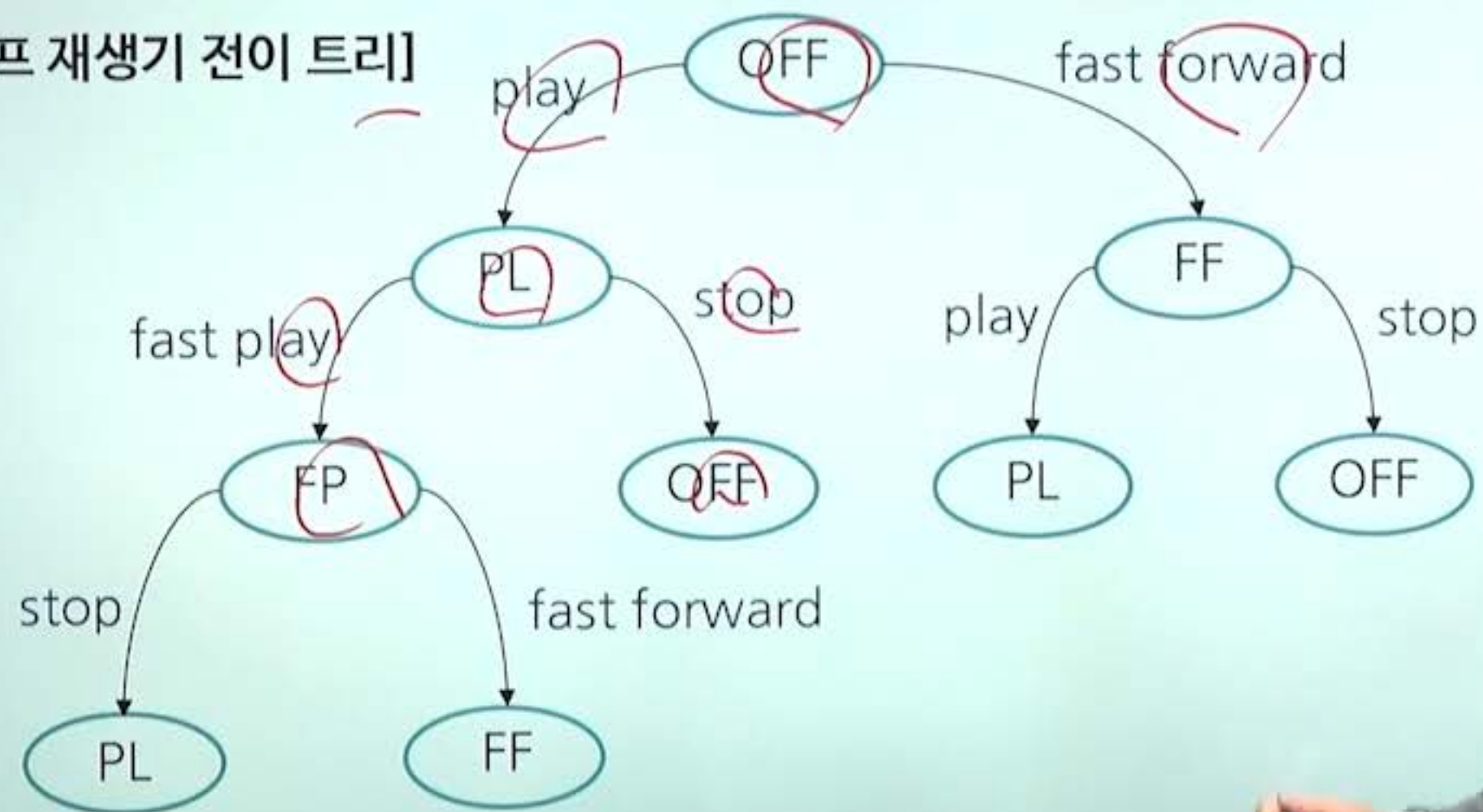


OFF: 초기 상태
PL: 재생 상태
FF: 빠른 전진 이동 상태
FP: 빠른 재생 상태

2 명세 기반 테스트

상태 전이 - 예시

[테이프 재생기 전이 트리]



2 명세 기반 테스트

상태 전이 - 예시

학습하기

[테이프 재생기 테스트 케이스 집합]

	입력		예상 출력	
	시작 상태	이벤트	행위	목적 상태
TC1	OFF	play	-	PL
TC2	PL	fast play	-	FP
TC3	FP	stop	-	PL
TC4	FP	fast forward	-	FF
TC5	PL	stop	-	OFF
TC6	OFF	fast forward	-	FF
TC7	FF	play	-	PL
TC8	FF	stop	-	OFF

2 명세 기반 테스트



세종사이버대학교

☞ 상태 전이 - 예시

[유효하지 않은 전이들을 테스트하는 TC 집합]



학습하기

	입력		예상 출력	
	시작 상태	이벤트	행위	목적 상태
TC1	OFF	play	예외 발생	OFF
TC2	OFF	fast play	예외 발생	OFF
TC3	PL	stop	예외 발생	PL
TC4	PL	fast forward	예외 발생	PL
TC5	FP	play	예외 발생	FP
TC6	FP	fast play	예외 발생	FP
TC7	FF	fast forward	예외 발생	FF
TC8	FF	fast play	예외 발생	FF



SEJONG CYBER UNIVERSITY