



# 4강. 소프트웨어 품질

컴퓨터과학과 김희천 교수



# 목차

- ① 소프트웨어 품질 개요와 분류
- ② 소프트웨어 제품의 품질 표준
- ③ 프로세스 품질 표준
- ④ 소프트웨어 품질 보증
- ⑤ 신뢰도





## Chapter. 1

# 소프트웨어 품질 개요와 분류

# 1. 소프트웨어 품질 개요

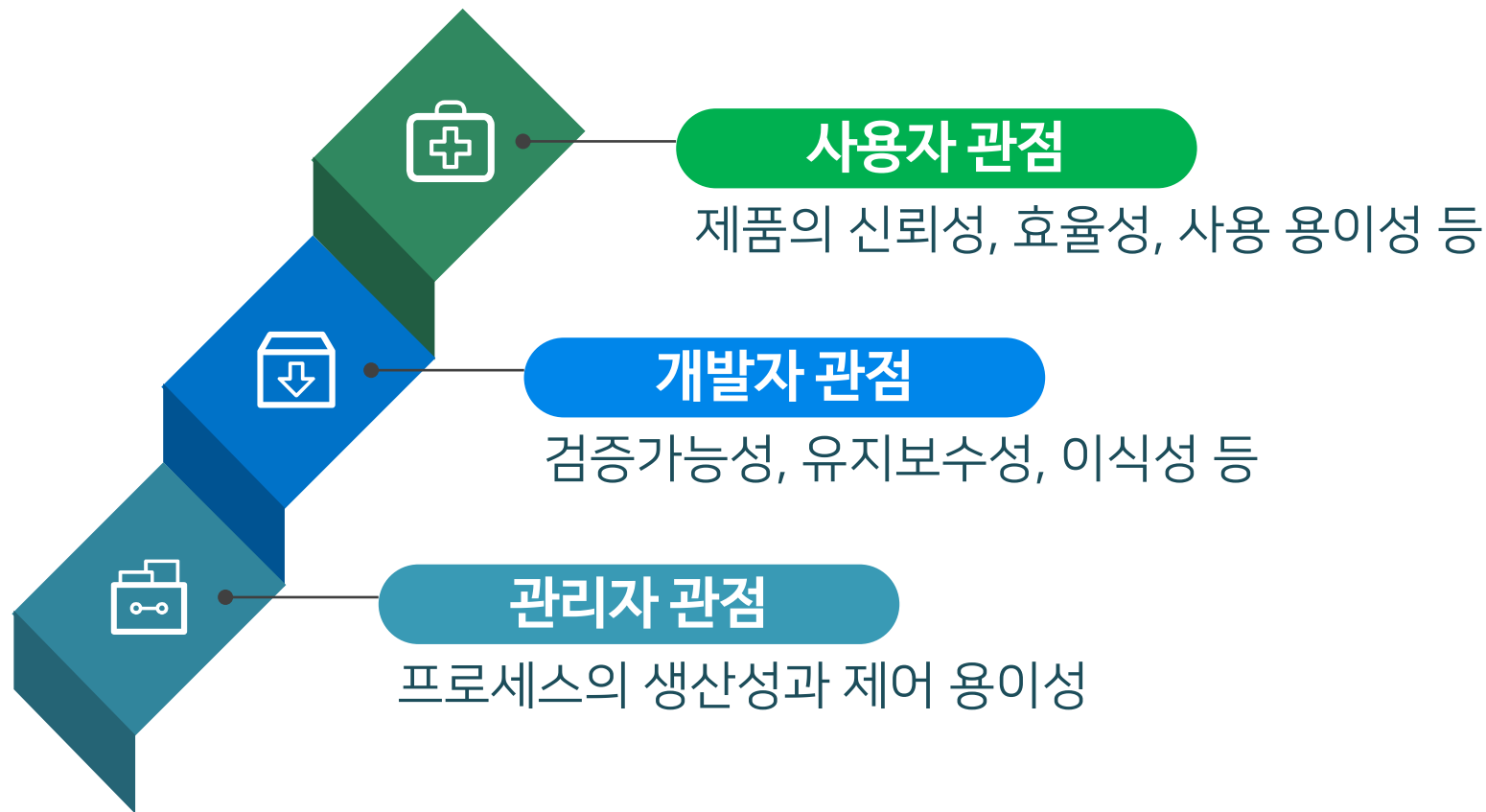
## + 품질의 정의

- × 제품이나 서비스가 가지는 수월성에 관한 종합적 특성
- × 생산자 입장의 품질: 명시된 요구사항을 만족시키는 정도
- × 고객 입장의 품질: 고객의 기대나 사용 목적에 부합하는 정도

## + 소프트웨어 품질

- × 소프트웨어 공학 목표 중 하나
- × 명확히 기술된 요구사항을 만족하고, 묵시적인 좋은 품질 특성을 가지는 것
- × 소프트웨어 제품과 프로세스에 적용되는 많은 구체적 품질 특성이 존재함

## 2. 품질 관점



### 3. 소프트웨어 품질의 분류 (1/2)

#### + 소프트웨어 제품의 품질 특성

- 내부 특성이 외부 특성에 영향을 줌
- 외부 특성과 내부 특성이 분명히 구분되는 것은 아님

#### 외부 특성

- 사용자 관점의 품질 특성
- 실행을 통해, 행위를 측정하여 평가할 수 있음
- 신뢰성이나 사용성

#### 내부 특성

- 개발자 관점의 품질 특성
- 개발자가 외부 특성을 개선하고자 할 때 도움을 줌
- 개발 문서나 코드에 대해 정적으로 측정하여 평가함

### 3. 소프트웨어 품질의 분류 (2/2)

- + 프로세스의 품질이 제품의 품질에 영향을 줄 수 있음
  - 제품과 함께 프로세스의 품질도 중요함

#### 제품 특성

- 제품이 가지는 품질 특성
- 고객 관점에서 제품이란 고객에게 전달되는 것
- 개발자는 요구사항, 설계 문서, 소스 코드, 사용자 매뉴얼 등을 모두 제품으로 생각함

#### 프로세스 특성

- 체계적인 프로세스가 정의되고 개발 과정에 적용되는 것
- 프로세스 품질은 소프트웨어 제품의 품질에 영향을 줌
- 프로세스 품질 관리와 개선을 위한 노력이 필요함



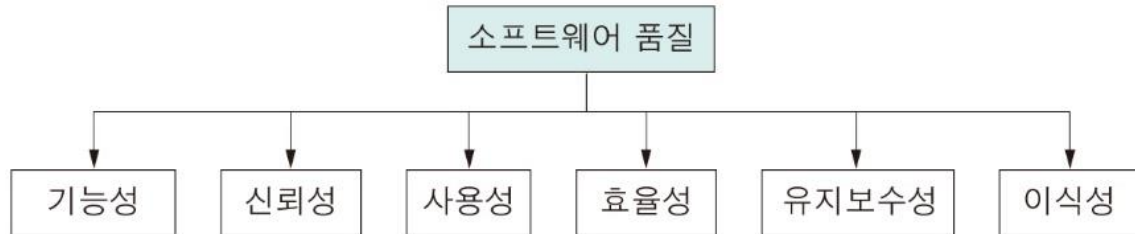
## Chapter. 2

# 소프트웨어 제품의 품질 표준



# 1. 제품 품질의 표준 - ISO/IEC 9126 (1/2)

- + 소프트웨어 제품의 품질 분류와 메트릭을 정의한 표준
- + 사용자 관점에 따라 제품의 품질 특성을 6가지로 분류(9126-1)
- + 6가지 주 품질 특성들은 각각 여러 부특성들로 구성됨



# 1. 제품 품질의 표준 - ISO/IEC 9126 (2/2)

**기능성**

명시적 또는 묵시적 사용자 요구를 만족시키기 위한 기능의 제공 능력

**신뢰성**

명시된 기간과 조건에서 성능 수준을 유지하는 소프트웨어 능력

**사용성**

시스템을 사용하는데 드는 노력과 사용자의 평가를 나타내는 특성

**효율성**

소프트웨어의 성능 수준과 필요한 자원 사이의 관계를 나타내는 특성

**유지보수성**

소프트웨어 제품의 수정에 드는 노력을 나타내는 특성

**이식성**

어떤 환경에서 다른 환경으로 이식될 수 있는 소프트웨어 제품의 능력

## 2. ISO/IEC 9126에서 부특성의 설명 (1/2)

품질 특성	부특성	설 명
기능성	적합성	소프트웨어가 요구된 작업을 수행하는가?
	정확성	결과가 기대와 같은가?
	상호운용성	시스템이 다른 시스템과 연동되는가?
	보안성	시스템이 불법적 접근을 방지하는가?
	준수성	시스템이 표준을 준수하는가?
신뢰성	성숙성	시간이 지나면서 소프트웨어 결함이 제거되는가?
	결함 내구성	소프트웨어가 오류를 처리할 능력을 가지는가?
	복구성	고장으로 인한 작업과 데이터 손실을 복구할 수 있는가?
사용성	이해성	시스템의 개념과 응용 분야를 쉽게 이해하는가?
	학습성	사용자가 시스템 사용법을 쉽게 배울 수 있는가?
	운영성	사용자가 큰 노력 없이 시스템을 사용할 수 있는가?
	매력성	인터페이스가 보기 좋은가?

## 2. ISO/IEC 9126에서 부특성의 설명 (2/2)

품질 특성	부특성	설 명
효율성	시간 효율성	시스템이 빠르게 응답하는가?
	자원 효율성	시스템이 효율적으로 자원을 사용하는가?
유지보수성	분석성	결함에 관해 빠른 진단이 가능한가?
	변경성	쉽게 소프트웨어를 수정할 수 있는가?
	안정성	변경 후에도 소프트웨어가 계속 기능을 할 수 있는가?
	시험성	쉽게 테스트할 수 있는가?
이식성	적응성	다른 환경으로 쉽게 옮겨질 수 있는가?
	설치성	소프트웨어를 쉽게 설치할 수 있는가?
	공존성	소프트웨어가 이식을 위한 표준을 준수하는가?
	대체성	다른 소프트웨어를 쉽게 대체할 수 있는가?

### 3. 외부 메트릭과 내부 메트릭

#### 외부 메트릭 (9126-2)

완성된 소프트웨어를 실행하여 제품의 품질을 측정하기 위한 것

+ 부특성을 측정하는 메트릭이 정의됨

+ 예 : 신뢰성을 높이려면 검토 과정에서 많은 결함을 발견하여 사용 중에 고장으로 연결되지 않도록 해야 함

- 외부 메트릭으로 평균 고장 간격(MTBF), 내부 메트릭으로 검토 중 발견된 결함의 밀도를 사용

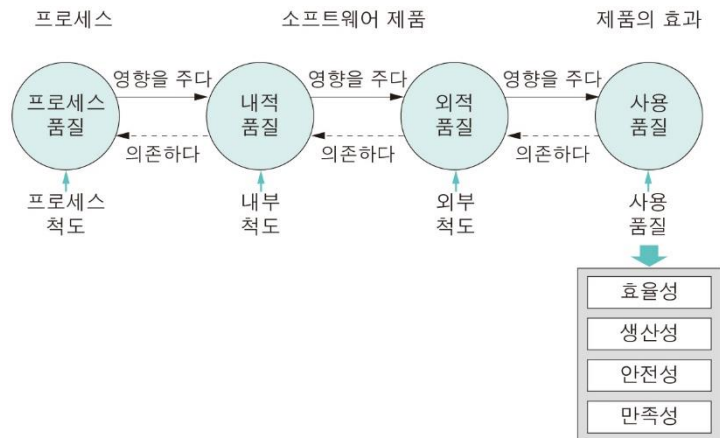
#### 내부 메트릭 (9126-3)

개발 과정 중에 나오는 소프트웨어 산출물의 품질을 측정하기 위한 것

## 4. 사용 품질

### + 사용자가 느끼는 제품의 실제 효과를 의미

- × 실제 사용 환경에서 효율성, 생산성, 안전성, 만족성을 가지고 명시된 목표를 달성하는 소프트웨어 제품의 능력
- × 소프트웨어 자체 특성이 아니라, 사용해 본 결과를 사용자가 측정
- × 사용 품질 관련 4 가지 특성과 메트릭을 정의함(9126-4)

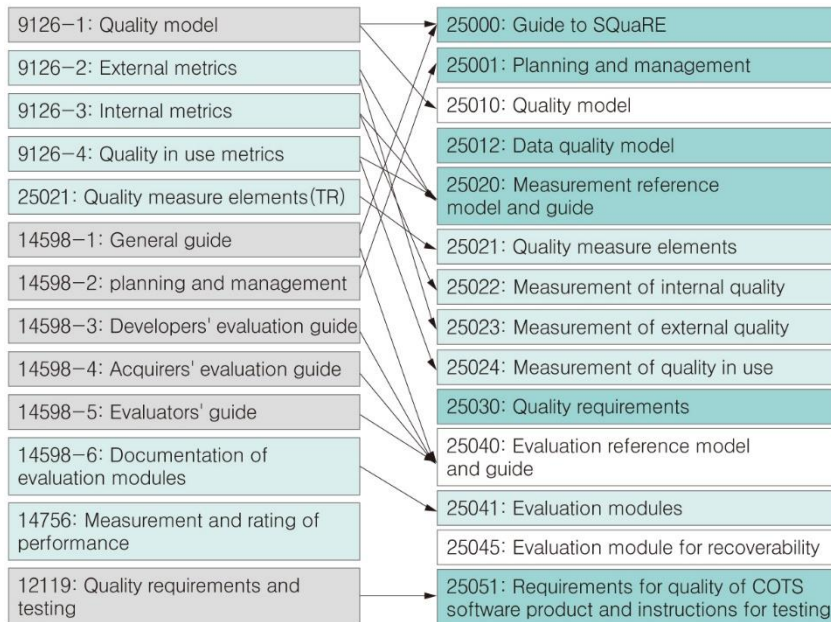


# 5. ISO/IEC 25000

## + SQuaRE (Software product Quality Requirements and Evaluation)

- × 품질 관리(2500n)
- × 품질 모델(2501n)
- × 품질 메트릭(2502n)
- × 품질 요구사항(2503n)
- × 품질 평가(2504n)

### ● ISO/IEC 25000 국제 표준 구성



## 6. 맥콜의 제품 품질 특성 (1/2)

### + 11개의 품질 요인을 제시

- × 제품 운영: 정확성, 신뢰성, 효율성, 무결성, 사용성
- × 제품 개선: 유지보수성, 시험성, 유연성
- × 제품 전이: 이식성, 재사용성, 상호운영성

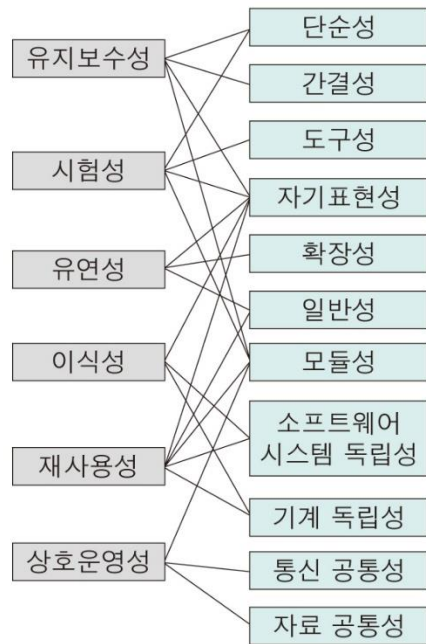
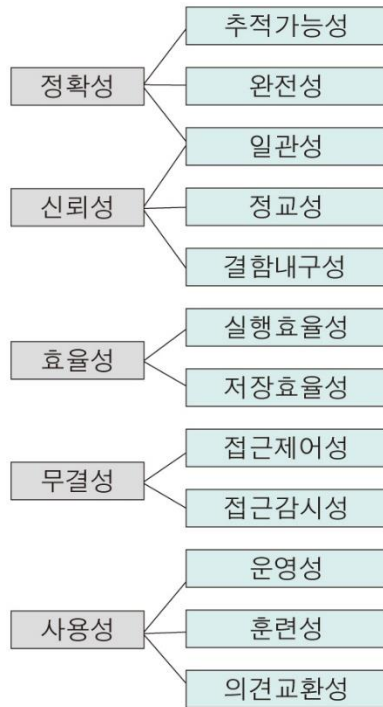
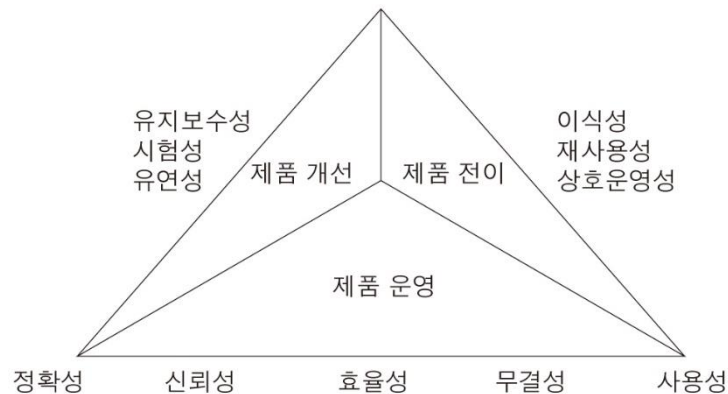
### + 23개의 품질 기준을 제시(개발자 관점의 내부 품질 기준)

### + 메트릭을 제시함

- × 품질 기준을 정량적으로 측정하는 방법과 단위



# 6. 맥콜의 제품 품질 특성 (2/2)





## Chapter. 3

# 프로세스 품질 표준

# 1. 프로세스 품질 표준 (1/2)

## 프로세스 품질



- 품질 목표의 달성을 위해서 고품질 소프트웨어의 개발을 유도하는 시스템을 갖추어야 함
- 원칙과 실무 지침을 갖춘 **성숙한 개발 프로세스**가 필요함

### + SPICE - ISO/IEC 15504

- Software Process Improvement and Capability dEtermination
- 개발 조직의 능력을 평가하고 개발 공정을 개선하여 품질과 생산성을 높이기 위한 **프로세스 평가 프레임워크**

# 1. 프로세스 품질 표준 (2/2)

## + ISO/IEC 12207

- × 소프트웨어의 구입, 공급, 개발, 운영, 유지보수를 위한 소프트웨어 **생명주기 프로세스의 공통 프레임워크**를 제공

## + CMMI

- × 조직의 프로세스 **역량 성숙도를 평가**하고 개선하기 위한 모델

## + ISO 9001

- × 조직에 **품질 경영체제**를 도입하고 프로세스 품질 인증 획득을 위한 모델

## 2. ISO 9000 시리즈

- + 품질 경영을 위한 기본 요소를 규정하고 실천을 위한 활동지침을 제시
- + ISO 9000
  - × 품질 관리 시스템의 기본과 관련 용어들을 설명
- + ISO 9001
  - × 품질 관리 시스템의 요건을 설명
  - × ISO 9001 인증은 정형화된 업무 프로세스가 적용되어 품질 문화가 정착되었다는 의미
- + 특징
  - × 프로세스 평가 보다 품질 관리 자체에 중점을 둠
  - × 평가를 등급화하지 않고 ISO 9001의 인증 여부만을 결정

### 3. CMMI

- + 조직의 프로세스 성숙도를 평가하는 모델

- × 개발을 위한 CMMI, 발주를 위한 CMMI, 서비스를 위한 CMMI가 존재

- + 다양한 기업에 프로세스의 구축, 평가, 개선을 위한 프레임워크를 제공

- + ISO 9001과 ISO 15504(SPICE)의 관심사를 포함한 통합 모델

- + CMMI 모델의 구조

- × 단계적 모델(성숙도 수준 평가): 조직 전체의 프로세스 성숙도를 1~5 수준으로 평가

- × 연속적 모델(역량 수준 평가): 20여 개의 실무 영역에 대해 각각 0~3 사이의 등급을 부여

## 4. CMM 모델의 구성 요소

- + '개발을 위한 CMMI' 는 20개의 실무 영역(PA)으로 구성됨
  - × 평가 영역을 여러 실무 영역으로 구분함
  - × 개별 실무 영역에서 수준(1~5)별로 분리된 지침 그룹(PG)이 존재함
- + 실무 영역은 4개 범주, 12개 역량 영역(CA)로 분류됨
  - × 범주로는 이행(Doing), 관리(Managing), 지원(Enabling), 개선(Improving) 이 있음

표 4-2 | 프로세스 품질 보증(PQA) 영역에 존재하는 수준별 실무 지침

실무 영역	수준	지침 번호	지침
프로세스 품질 보증 (PQA)	1	1.1	프로세스와 작업 산출물 문제를 파악하고 해결한다.
	2	2.1	과거 품질 데이터를 기반으로 품질보증 방법과 계획을 개발하고 업데이트하고 준수한다.
		2.2	작업 노력을 통해서, 수행된 프로세스와 작업 산출물을 프로세스 기록 대비 객관적으로 평가한다.
		2.3	품질과 부적합 문제를 알리고 해결방법을 보장한다.
		2.4	품질보증 활동의 결과를 기록하고 활용한다.
	3	3.1	품질보증 활동 중에 개선점에 대한 기회를 파악하고 기록한다.

01

## 연속적 모델(역량 수준 평가)

- ✦ 개별 실무 영역 별로 능력을 평가하여 0~3 사이의 점수를 매김
- 실행되지않음(Incomplete), 초기(Initial), 관리됨(Managed), 정의됨(Defined)

## 단계적 모델(성숙도 수준 평가)

- ✦ 조직의 전체 프로세스 성숙도를 하나의 등급으로 평가(1~5)
  - 수준 1: 초기 상태(Initial)
  - 수준 2: 관리됨(Managed)  
기본적 프로젝트 관리 프로세스가 구축되어 관리됨
  - 수준 3: 정의됨(Defined)  
조직 특성에 맞는 표준 프로세스가 존재
  - 수준 4: 양적으로 관리됨(Quantitatively Managed)  
프로젝트 활동이 정량적으로 관리되고 통제됨
  - 수준 5: 최적화됨(Optimizing)  
지속적인 개선 활동이 정착화됨





## Chapter. 4

# 소프트웨어 품질 보증

# 1. 소프트웨어 품질 보증(SQA)

## + 소프트웨어 품질 요구가 만족됨을 보증하는 품질 관리 활동

- × 개발자나 품질 보증 팀이 수행하는 체계적 품질 관리 활동으로 개발 과정 전체에 적용함
- × 궁극적 목적은 사용자가 제품을 사용할 때, 기대했던 기능과 성능을 보일 것임을 고객에게 보증하는 것

## + 품질 보증 활동의 예

- × 개발 팀이 준수해야 하는 제품과 프로세스의 표준을 정하고 시스템에서 요구되는 품질 속성을 정의
- × 사용하는 절차, 도구 및 기술들이 적정 수준 이상임을 보증
- × 표준에 기초하여 필요한 활동을 취함(QC: 품질 제어)

## 2. 품질 보증 계획

### + SQA 작업 수행을 위한 로드맵으로 품질 보증 프로세스와 방법을 정하는 일

- ×적정 시점에 산출물들이 만들어지는지 확인
- ×품질을 만족하는지 검토
- ×문서로써 확인

### + SQA 계획의 구성

- ×품질 보증 조직의 구성
- ×제품과 품질 프로세스 표준
- ×검토와 감사 방법
- ×테스트 계획과 절차
- ×형상 관리 방법, 위험 관리 방법 등

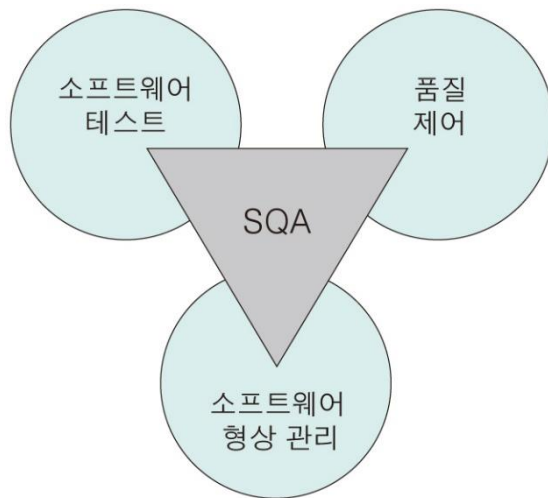
### 3. 품질 제어(QC)

+ 개발 프로세스, 코드 및 관련 문서가 품질 보증 절차에 따라 수행되고 표준을 따르며 품질 목표를 만족하도록 필요한 활동을 취하는 것

- × 프로세스에 따라 활동을 수행하는지 검토
- × 요구 명세서, 설계 문서, 코드 등을 검토
- × 테스트 계획이나 형상 관리 절차에서 나오는 문서들을 검토

+ 용어의 혼용

- × 품질 제어는 결함을 발견하고 수정하는 목적을 가지며, 품질 보증은 사전 예방 활동의 의미가 있음
- × 제품을 대상으로 하는 것을 품질 제어, 프로세스를 대상으로 하는 것을 품질 보증으로 보기도 함



## 4. 확인과 검증(V&V)

- ✦ 요구사항 명세서의 검토, 설계 문서와 코드의 인스펙션, 프로그램의 테스트를 모두 포함하는 개념(정적 검토와 동적 테스트를 포함)

### 확인(Verification)

- 확인은 소프트웨어가 명세서와 일치하는지 검사하는 것

### 검증(Validation)

- 검증은 소프트웨어가 고객의 기대를 충족하는지 검사하는 것

## 5. 검토

### + 정적 테스트라고 함

- × 프로그램을 실행하지 않고 검토하는 회의
- × 정적 분석 도구를 사용하여 분석할 수 있음
- × 적은 노력으로 빠르게 결함을 발견할 수 있다고 알려짐

### + 목적

- × 소프트웨어 요소가 요구 명세서와 일치하는지 확인
- × 계획, 표준 및 지침에 맞게 개발되었는지 확인
- × 소프트웨어 요소의 변경이 적절히 구현되었는지 확인

## 6. 검토 방법

### 공식 기술 검토 (FTR)

- 제품의 완전성, 정확성, 일관성, 기술적 타당성, 효율성 및 표준과 지침의 준수 여부를 조사하는 공식적 회의

### 인스펙션 (Inspection)

- 공식 기술 검토 전이나 테스트를 수행하기 전에 설계 문서나 코드를 **작성자**가 아닌 **동료나 전문가 팀**이 **검토**하는 것
- 회의 전에 필요한 자료를 참가자에게 나누어 사전 검토 작업을 하고 회의 참석
- 코드 인스펙션의 경우 범하기 쉬운 흔한 오류의 발견에 초점을 둠

### 코드 워크스루 (Code Walkthrough)

- 알고리즘이나 코드상의 경로를 따라가면서 결함을 찾을 목적의 비공식적 검토 방법
- 작성자 본인이 다른 구성원들과 질의 응답함



## Chapter. 5

# 신뢰도



# 1. 신뢰도

- + 의도된 기능을 고장 없이 실행할 수 있는 프로그램 능력
  - × 가장 중요한 소프트웨어 품질 특성의 하나
  - × 서비스를 요청할 때, 시스템이 고장 없이 정확하게 작동할 확률
- + 신뢰도는 사용 환경과 고장의 결과에 따라 다르게 인식될 수 있음
  - × 내재된 결함(fault)이 있더라도 고장(failure)으로 연결되지 않을 수 있음
  - × 사용 환경에 따라 고장 빈도가 다를 수 있음
  - × 고장의 결과가 심각하지 않다면 신뢰성이 있다고 할 수 있음

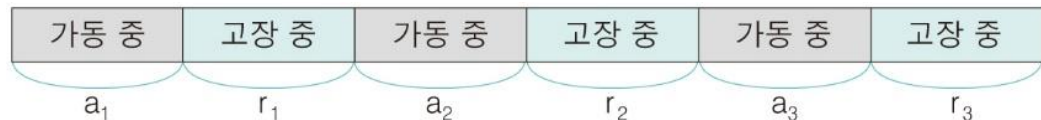
## 2. 신뢰도 메트릭 (1/2)

### + MTTF(Mean Time To Failures)

- × 가동되어 고장이 발생할 때까지의 평균 시간(평균 수명)
- × 고장의 복구를 고려하지 않음
- ×  $MTTF = (a_1 + a_2 + a_3) / 3$

### + MTBF(Mean Time Between Failures)

- × 고장이 수리되어 가동된 후 다시 고장이 일어날 때 사이의 평균 간격(MTTF)
- × 고장의 복구를 고려함
- × 또는 MTTF에 평균 복구 시간(MTTR)을 더한 것으로 정의함(아래)
- ×  $MTBF = ((a_1 + a_2 + a_3) + (r_1 + r_2 + r_3)) / 3$



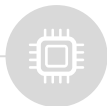
## 2. 신뢰도 메트릭 (2/2)

### + AVAIL(가용성)

- × 시스템이 가동될 확률, 전체 시간에서 가용한 시간의 비율
- × **논스톱 시스템의 신뢰도 측정에 사용**
- × 가용성을 높이려면, 고장 발생 후 빠른 복구가 필요함
- ×  $AVAIL = MTBF / (MTBF + MTTR) = (a_1 + a_2 + a_3) / (a_1 + a_2 + a_3 + r_1 + r_2 + r_3)$

### + ROCOF(Rate of OCcurrences Of Failures)

- × 고장의 발생 비율
- × 규칙적이고 빈번한 서비스 요청이 들어오는 시스템의 신뢰도 측정
- × 예: 예약 시스템의 ROCOF가 0.002라면 1000회의 요청에서 2회의 오류가 발생



다음강의

## 5강. 소프트웨어 테스트