

2. 시험 개요

- 소프트웨어 테스트는 개발 소프트웨어에 대한 **품질** 기준을 만족하기 위해 수행한다.
- 소프트웨어 테스트 프로세스는 현재 구현되어 있는 소프트웨어의 모습과
- 기대치와의 차이점을 판별할 수 있도록 각 테스트 단계의 활동을 정의하는 것이 목적이다.
- 각 테스트 단계는 논리적으로 순차적이지만 동시에 진행될 수 있다.

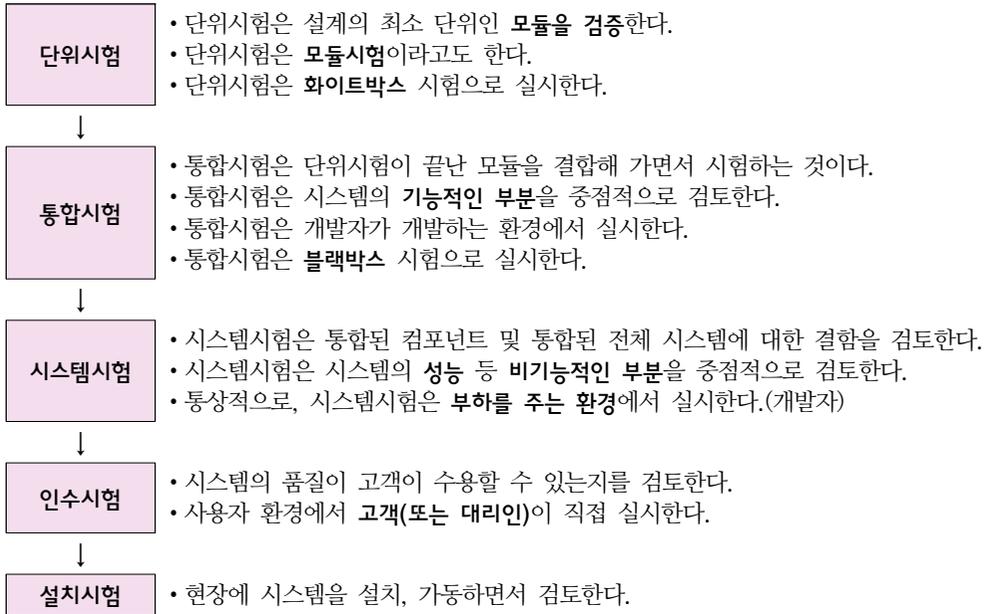
// 소프트웨어 테스트 기본 원칙

1. 완벽한 테스트는 불가능하다. - 예 : 윈도우 운영체제도 오류가 존재함!
2. 테스트는 결함이 존재하는 것을 증명하는 활동이다.(결함이 없다는 것을 증명할 수 없다)
3. 테스트는 가능한 개발 초기에 시작해야 한다.

// 소프트웨어 테스트 프로세스 단계

테스트	세부 내용
테스트 계획	<ul style="list-style-type: none"> • 테스트 목표를 달성하기 위해 필요한 활동 내역을 정의하는 단계이다. • 테스트 범위 정의(단위시험, 통합시험, 시스템시험, 인수시험 계획) • 테스트 기준 검토 • 테스트 목표 수립 • 테스트 기법 할당 • 테스트 환경 구축 • 테스트 용이성 평가 • 테스트 상황, 요구사항, 데이터 식별
테스트 설계	<ul style="list-style-type: none"> • 테스트 케이스 명세, 테스트 케이스 데이터 생성 • 테스트 우선순위 정의 • 선행 테스트 실시 • 테스트 기대 결과 비교
테스트 실행	<ul style="list-style-type: none"> • 테스트 케이스에 의한 테스트 수행 • 테스트 완료 조건의 달성 여부 확인 • 테스트 수행 결과에 따른 최종 보고서 작성
테스트 평가	<ul style="list-style-type: none"> • 테스트 산출물 확인 • 테스트 프로세스 평가

// 시험 과정



〈간단한 성적처리 프로그램에 대한 간단한 명세〉

점수(소공=80, 디비=90)를 입력받아 합계와 평균을 구해 출력하도록 한다.
합계와 평균은 정수형으로 입출력한다.

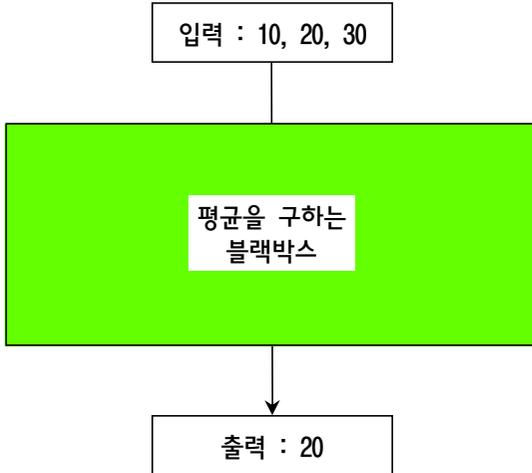
↓ 프로그램 명세에 대한 원시코드
↓

〈성적처리 프로그램〉

```
int 합계(int 소공, int 디비) { return 소공 + 디비; } //단위시험(모듈시험)
int 평균(int 합) { return 합 / 2; } //단위시험(모듈시험)
void main(void)
{
    int 소공 = 80, 디비 = 90; //입력
    int 합 = 합계(소공, 디비);
    printf("%d\n", 합); //출력 : 80 + 90 = 170
    printf("%d\n", 평균(합)); //출력 : 170 / 2 = 85
}
```

다음은 블랙박스 시험 방법을 보여주는 그림이다.

〈블랙박스 시험〉



// 화이트박스 시험과 블랙박스 시험 비교

	화이트박스 시험	블랙박스 시험
방법	<ul style="list-style-type: none"> • 프로그램 내부 로직을 참조하여 시험 • 프로그램 모든 경로를 시험 	<ul style="list-style-type: none"> • 프로그램의 외부 명세를 직접 시험 • 외부 명세는 프로그램 기능에 대한 설명이다.
특징	<ul style="list-style-type: none"> • 구조시험 • 모듈시험 • 로직 위주(logic-driven) 시험 	<ul style="list-style-type: none"> • 기능시험 • 데이터 위주(data-driven) 시험 • 입출력 위주(i/o-driven) 시험
관점	<ul style="list-style-type: none"> • 개발자 관점 	<ul style="list-style-type: none"> • 사용자 관점

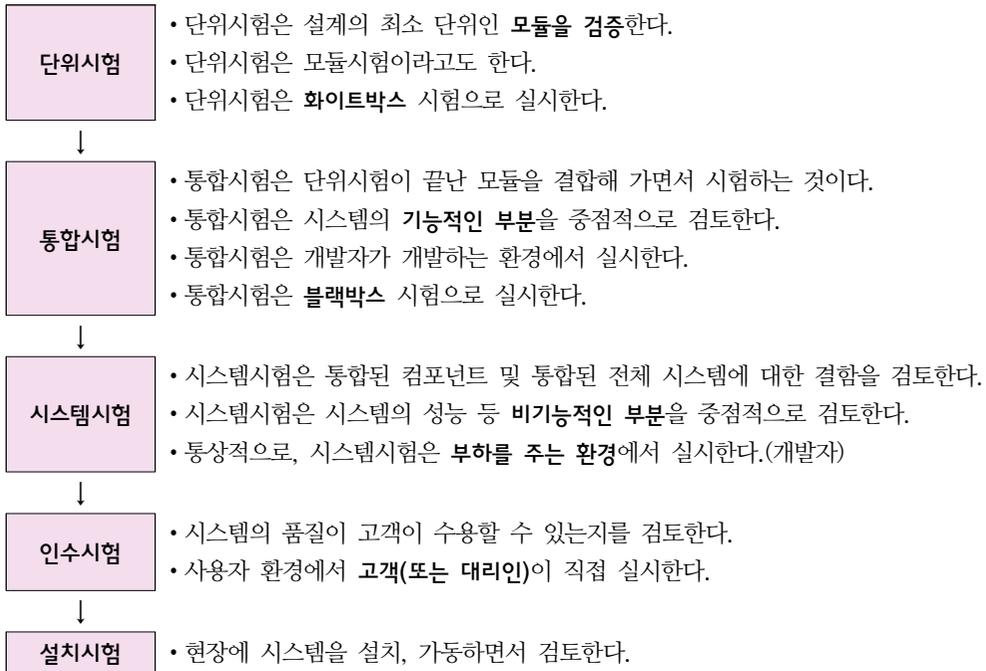
기출문제 분석

1. 소프트웨어 개발 단계에 따른 테스트의 순서는? [2020년 국가 7급]

- ㉠. 통합 테스트
- ㉡. 인수 테스트
- ㉢. 단위 테스트
- ㉣. 시스템 테스트

- ① ㉣ - ㉠ - ㉡ - ㉢
- ② ㉣ - ㉠ - ㉢ - ㉡
- ③ ㉣ - ㉡ - ㉠ - ㉢
- ④ ㉣ - ㉢ - ㉠ - ㉡

☞ 소프트웨어 테스트 순서



2. 화이트박스(white box) 시험에 해당하는 것은? [2008년 국가 7급]

- ① 구조시험(structure testing)
- ② 동등분할(equivalence partitioning)
- ③ 경계값 분석(boundary value analysis)
- ④ 원인 결과 그래프(cause-effect graph)

☞ 소프트웨어 시험

화이트박스 시험	구조시험, 문장시험, 분기시험, 조건시험, 경로시험, 기초경로시험 등
블랙박스 시험	기능시험, 동등분할, 오류예측, 원인-효과 그래프, 경계값분석, 비교시험 등

정답 : ①

3. 화이트박스 테스트만을 모두 고르면? [2022년 국가 7급]

- ㉠. 제어흐름 테스트 ㉡. 경로 테스트
- ㉢. 상태기반 테스트 ㉣. 원인 결과 그래프

- ① ㉠, ㉡ ② ㉠, ㉢ ③ ㉡, ㉣ ④ ㉢, ㉣

☞ 테스트

화이트박스 시험	<ul style="list-style-type: none"> • 경로 시험(path testing) • 기본경로 시험(basic path testing) • 분기 시험(branch testing) • 조건 시험(condition testing) • 제어흐름 시험(control flow testing) • 데이터흐름 시험(data flow testing)
블랙박스 시험	<ul style="list-style-type: none"> • 비교 시험(comparison testing) • 동치 분할 시험(equivalence partitioning testing) • 오류 예측 시험(error guessing testing) • 원인-효과 그래프 시험(cause-effect graph testing) • 경계값 분석 시험(boundary value analysis testing) 등 • 상태 기반 시험(state-based testing)

정답 : ①

4. 블랙박스 테스트 기법에 해당하지 않는 것은? [2015년 국가 7급]

- ① 경계값 분석(boundary value analysis)
- ② 원인-결과 그래프(cause-effect graph)
- ③ 기본경로 테스트(basic path test)
- ④ 동등 분할(equivalence partitioning)

☞ 테스트 기법

화이트박스 시험	<ul style="list-style-type: none"> • 경로 시험(path testing) / 기본경로 시험(basic path testing) • 제어흐름 시험(control flow testing) • 데이터흐름 시험(data flow testing)
블랙박스 시험	<ul style="list-style-type: none"> • 비교 시험(comparison testing) • 동치 분할 시험(equivalence partitioning testing) • 원인-효과 그래프 시험(cause-effect graph testing) • 경계값 분석 시험(boundary value analysis testing) 등

정답 : ③

5. 소프트웨어 시험 기법에서 블랙박스 시험 기법의 특징으로 가장 옳지 않은 것은? [2018년 서울 7급]

- ① 동등 분할(equivalence partitioning)
- ② 조건 검사(condition coverage)
- ③ 경계값 분석(boundary value analysis)
- ④ 원인-결과 그래프(cause-effect graphing)

☞ 소프트웨어 시험 기법

화이트박스 시험	<ul style="list-style-type: none"> • 경로 시험(path testing) / 기본경로 시험(basic path testing) • 분기 시험(branch testing) • 조건 시험(condition testing) • 제어흐름 시험(control flow testing) • 데이터 흐름 시험(data flow testing)
블랙박스 시험	<ul style="list-style-type: none"> • 비교 시험(comparison testing) • 동치 분할 시험(equivalence partitioning testing) • 원인-효과 그래프 시험(cause-effect graph testing) • 경계값 분석 시험(boundary value analysis testing) • 오류 예측 시험(error guessing testing)

정답 : ②

6. 소프트웨어의 화이트박스 테스트에 대한 설명으로 옳지 않은 것은? [2022년 지방 9급]

- ① 클래스 박스(glass-box) 테스트라고 부른다.
- ② 소프트웨어의 내부 경로에 대한 지식을 보지 않고 테스트 대상의 기능이나 성능을 테스트하는 기술이다.
- ③ 문장 커버리지, 분기 커버리지, 조건 커버리지 등의 검증 기준이 있다.
- ④ 모듈의 논리적인 구조를 체계적으로 점검하기 때문에 구조적 테스트라고도 한다.

☞ 화이트박스 테스트

-
- 소프트웨어의 내부 경로에 대한 지식을 보지 않고 테스트 대상의 기능이나 성능을 테스트하는 기술이다.(x)
 - 화이트박스 테스트는 모듈의 내부 경로를 보고 테스트 한다.
 - 기능이나 성능을 테스트는 블랙박스 테스트 이다.
 - 화이트박스시험은 원시코드를 눈으로 직접 보면서 시험한다. - 클래스 박스(glass-box)
-

정답 : ②

7. 소프트웨어 테스트에 대한 설명으로 옳지 않은 것은? [2016년 국가 9급]

- ① 단위(unit) 테스트는 개별적인 모듈에 대한 테스트이며 테스트 드라이버(driver)와 테스트 스텝(stub)을 사용할 수 있다.
- ② 통합(integration) 테스트는 모듈을 통합하는 방식에 따라 빅뱅(big-bang) 기법, 하향식(top-down) 기법, 상향식(bottom-up) 기법을 사용한다.
- ③ 시스템(system) 테스트는 모듈들이 통합된 후 넓이우선방식 또는 깊이우선방식을 사용하여 테스트한다.
- ④ 인수(acceptance) 테스트는 인수하기 전에 사용자의 요구사항이 만족되었는지 테스트한다.

☞ 소프트웨어 테스트

-
- 시스템(system) 테스트는 모듈들이 통합된 후 넓이우선방식 또는 깊이우선방식을 사용하여 테스트한다.(x) → 넓이우선방식 또는 깊이우선방식은 통합시험에서 실시한다.

// 시스템 테스트

- 시스템 테스트는 시스템의 성능 등 비기능적인 부분을 중점적으로 검토한다.
 - 통상적으로, 시스템 테스트는 부하를 주는 환경에서 실시한다.
-

정답 : ③

8. 소프트웨어 테스트 원칙과 가장 거리가 먼 것은? [2023년 군무 7급]

- ① 테스트는 결함이 있다는 것을 증명하는 것이다.
- ② 테스트는 독립적인 조직에 의해 수행되어야 한다.
- ③ 테스트는 가능한 개발 초기에 시작해야 한다.
- ④ 테스트는 정황에 독립적으로 진행되어야 한다.

☞ 소프트웨어 테스트 7가지 기본 원칙

1. 테스트는 결함이 존재하는 것을 증명하는 활동이다(테스트는 결함이 없다는 것을 증명할 수 없다)
소프트웨어 테스트 완료 및 발견된 문제를 모두 해결하여도 결함이 없다는 것을 증명할 수 있는 것은 아니다. 또한 문제가 발견되지 않았다고 해서 결함이 없다는 것이 증명되지는 않는다. 테스트는 프로그램의 결함이 없음을 보장하는 활동이 아니라, 결함이 존재하는 것을 밝히기 위한 활동이다.

2. 완벽한 테스트는 불가능하다(모든 테스트는 불가능하다)

단순한 소프트웨어가 아닌 이상 내부 조건, 입력값 등에 대한 모든 조합을 확인할 수 없다. 따라서 테스트 대상의 위험분석 후에 가장 중요한 부분부터 집중적으로 테스트한다.

3. 테스트는 가능한 개발 초기 단계에서부터 시작한다

요구사항 분석 단계부터 테스트를 진행하면 문서의 결함을 확인할 수 있다. 시작부터 테스트를 실시하면 전체 테스트 기간을 단축할 수 있다.

4. 결함 집중(결함 편재)

소프트웨어 결함은 소수의 특정 모듈에 집중되는 경향이 있다.

예1 : 자체적으로 복잡한 구조의 모듈

예2 : 소프트웨어의 다른 부분과 복잡하게 상호작용하는 모듈

예3 : 개발 난이도가 높거나 최신 기술이 적용된 모듈

예4 : 경험이 적은 개발팀에서 개발한 모듈 등

5. 살충제 패러독스

동일한 테스트 케이스를 반복적으로 수행하면 더 이상 새로운 결함을 찾아낼 수 없다.

이를 극복하려면 새로운 기법, 다른 테스트 케이스를 정기적으로 사용해야 한다.

벌레에게 같은 살충제를 계속해서 사용하면, 어느샌가 살충제에 대한 내성이 생긴다.

6. 테스트는 정황(context, 상황)에 따라 이루어져야 한다

소프트웨어 종류에 따라 해당 소프트웨어에 맞는 테스트 방식이 적용되어야 한다.

개발 프로젝트인지? 운영중인 시스템의 유지보수인지? 등에 따라 적절한 테스트 방식을 적용한다.

7. 오류 부재의 궤변(결함 제로의 함정)

거의 모든 결함을 확인하고 제거하였다고 해도 사용자 요구 또는 비즈니스 목적을 충족시키지 못하면 품질이 높다고 할 수 없다.

결함이 없다는 것이 곧 고품질의 소프트웨어라는 것은 아니다.

9. 개발과정에서 실시되는 시험의 순서를 바르게 나열한 것은? [2010년 국가 7급]

- ㄱ. 모듈 내부적인 오류를 발견하기 위한 시험
- ㄴ. 사용자의 요구사항을 만족하는지 판단하는 시험
- ㄷ. 모듈간의 인터페이스 오류를 발견하기 위한 시험

- ① ㄱ→ㄴ→ㄷ ② ㄴ→ㄱ→ㄷ
- ③ ㄱ→ㄷ→ㄴ ④ ㄴ→ㄷ→ㄱ

☞ 시험 순서 : 단위시험 → 통합시험 → 인수시험

- 단위시험 : ㄱ. 모듈 내부적인 오류를 발견하기 위한 시험
- 통합시험 : ㄷ. 모듈간의 인터페이스 오류를 발견하기 위한 시험
- 인수시험 : ㄴ. 사용자의 요구사항을 만족하는지 판단하는 시험

정답 : ③

10. 테스트의 목표를 수립하고, 테스트의 범위를 선정하는 활동으로 가장 옳은 것은? [2018년 서울 7급]

- ① 테스트 계획 ② 테스트 설계
- ③ 테스트 실행 ④ 테스트 평가

☞ 소프트웨어 테스트 단계

테스트	세부 내용
테스트 계획	<ul style="list-style-type: none"> • 테스트 목표를 달성하기 위해 필요한 활동 내역을 정의하는 단계이다. • 테스트 범위 정의(단위시험, 통합시험, 시스템시험, 인수시험 계획) • 테스트 기준 검토 / 테스트 목표 수립 / 테스트 기법 할당 / 테스트 환경 구축 • 테스트 용이성 평가 / 테스트 상황, 요구사항, 데이터 식별
테스트 설계	<ul style="list-style-type: none"> • 테스트 케이스 명세, 테스트 케이스 데이터 생성 • 테스트 우선순위 정의 / 선행 테스트 실시 / 테스트 기대 결과 비교
테스트 실행	<ul style="list-style-type: none"> • 테스트 케이스에 의한 테스트 수행 / 테스트 완료 조건의 달성 여부 확인 • 테스트 수행 결과에 따른 최종 보고서 작성
테스트 평가	<ul style="list-style-type: none"> • 테스트 산출물 확인 / 테스트 프로세스 평가

정답 : ①

11. 다음 중 화이트박스 검사 기법에 대한 설명으로 가장 거리가 먼 것은? [2023년 군무 7급]

- ① 소스코드의 모든 문장을 한 번 이상 수행함으로써 진행된다.
- ② 모듈 안의 작동을 직접 관찰할 수 있다.
- ③ 프로그램 내부에 사용되는 변수나 서브루틴 등의 오류를 찾는다.
- ④ 입력 값에 대한 예상 출력 값을 정해놓고 그대로 결과가 나오는지 확인함으로써 오류를 찾는다.

☞ 화이트박스 검사

-
- 입력 값에 대한 예상 출력 값을 정해놓고 그대로 결과가 나오는지 **확인**함으로써 오류를 찾는다.(x) → 블랙박스 검사

// 화이트박스 시험, 블랙박스 시험 비교

	화이트박스 시험	블랙박스 시험
방법	• 프로그램 내부 로직을 참조하여 시험 • 프로그램 모든 경로를 시험	• 프로그램의 외부 명세를 직접 시험 • 외부 명세는 프로그램 기능에 대한 설명이다.
특징	• 구조시험 • 모듈시험 • 로직 위주(logic-driven) 시험	• 기능시험 • 데이터 위주(data-driven) 시험 • 입출력 위주(i/o-driven) 시험
관점	• 개발자 관점	• 사용자 관점

정답 : ④

12. 모듈 안의 작동을 자세히 관찰할 수 있으며, 프로그램 원시코드의 논리적인 구조를 검사하도록 테스트 케이스를 설계하는 프로그램 테스트 방법은? [2010년 국가 7급]

- ① 화이트박스 테스트
- ② 블랙박스 테스트
- ③ 알파 테스트
- ④ 베타 테스트

☞ 화이트박스 테스트

-
- 원시코드의 논리적인 구조를 검사하는 것은 화이트박스 테스트이다.

정답 : ①

13. 다음 중 소프트웨어 기능 뿐만 아니라 비기능적인 속성도 만족되는지를 검사하는 시험으로 가장 적합한 것은? [2023년 군무 7급]

- ① 단위 테스트
- ② 통합 테스트
- ③ 시스템 테스트
- ④ 인수 테스트

☞ 소프트웨어 시험

시스템시험	<ul style="list-style-type: none">• 시스템시험은 통합된 컴포넌트 및 통합된 전체 시스템에 대한 결함을 검토한다.• 시스템시험은 시스템의 성능 등 비기능적인 부분을 중점적으로 검토한다.• 통상적으로, 시스템시험은 부하를 주는 환경에서 실시한다.(개발자)
-------	--

정답 : ③

14. 소프트웨어 테스트에 대한 설명으로 옳지 않은 것은? [2016년 계리직]

- ① 베타(beta) 테스트는 고객 사이트에서 사용자에게 의해서 수행된다.
- ② 회귀(regression) 테스트는 한 모듈의 수정이 다른 부분에 미치는 영향을 검사한다.
- ③ 화이트박스(white box) 테스트는 모듈의 내부 구현보다는 입력과 출력에 의해 기능을 검사한다.
- ④ 스트레스(stress) 테스트는 비정상적으로 과도한 분량 또는 빈도로 자원을 요청할 때의 영향을 검사한다.

☞ 소프트웨어 테스트

<ul style="list-style-type: none">• 화이트박스 테스트는 모듈의 내부 구현보다는 입력과 출력에 의해 기능을 검사한다.(x)→ 화이트박스 테스트는 모듈의 내부 구현에 대해 테스트 한다.→ 기능을 검사하는 블랙박스 테스트 이다.
--

정답 : ③