

# 정보처리기사 필기

소프트웨어 개발 4 통합구현

양문자 선생님

출처 : ncs 학습모듈(NCS능력단위 통합구현)  
참조 : 데이터베이스개론/김연희/한빛아카데미

# 소프트웨어 개발

## 차례

### 1 데이터 입출력 구현

- 1) 논리 데이터저장소 확인
- 2) 물리 데이터저장소 설계
- 3) 데이터 조작 프로시저 작성

### 2 통합 구현

- 1) 모듈 구현
- 2) 통합 구현 관리

### 3 제품소프트웨어 패키징

### 4 애플리케이션 테스트 관리

### 5 인터페이스 구현

# 소프트웨어 개발

## 1) 모듈 구현

단위모듈 구현 : 단위 기능을 모듈로 구현

### 1. 단위 기능 명세서 작성 : 설계 과정에서 단위 기능을 명세화한 문서

- **추상화**: 복잡한 시스템을 단순히 구현, 상위 수준에서 제품의 구현을 먼저 생각, 큰 흐름 잡기
- **구조화**: 시스템을 분해하여 단위 기능별로 구분하고 계층적으로 구성
  - 문제를 하위 수준의 독립된 단위로 나누기
  - 구분된 문제의 자세한 구현은 뒤로 미루기
  - 점증적으로 구체화 작업 계속하기
- **캡슐화(정보 은닉)**: 단위 모듈(Unit Module)의 내부 내용은 숨기고 인터페이스를 통해서만 메시지를 전달하도록 하여 다른 모듈의 접근과 변경을 막는다.
- **기능적 독립성**: 모듈은 뚜렷한 목적을 가지면서, 다른 모듈과의 상호 의존도가 낮도록 함.

## 2. 입출력 기능 구현 : 명세서에 정의한 데이터 형식에 따라 알고리즘 및 데이터 구현

- IPC(Inter-Process Communication)

: 모듈간 통신 방식을 구현하기 위해 사용되는 대표적인 인터페이스 집합

- 1) PIPE, Named PIPE(FIFO) : 'PIPE'라는 메모리를 이용하여 모든 프로세스들 사이에서 통신이 가능.
- 3) Message Queue : 메시지가 발생하면 이를 전달. PIPE가 데이터의 흐름이라면 메시지 큐는 메모리 공간.
- 4) Shared Memory(공유 메모리) : 다수의 프로세스가 공유 가능한 메모리를 구성하여 프로세스간 통신 수행
- 5) Socket : 프로세스들 사이의 통신을 가능하게 함.
- 6) Semaphore : 공유 자원에 대한 접근제어를 통해 프로세스간 데이터를 동기화 하고 보호

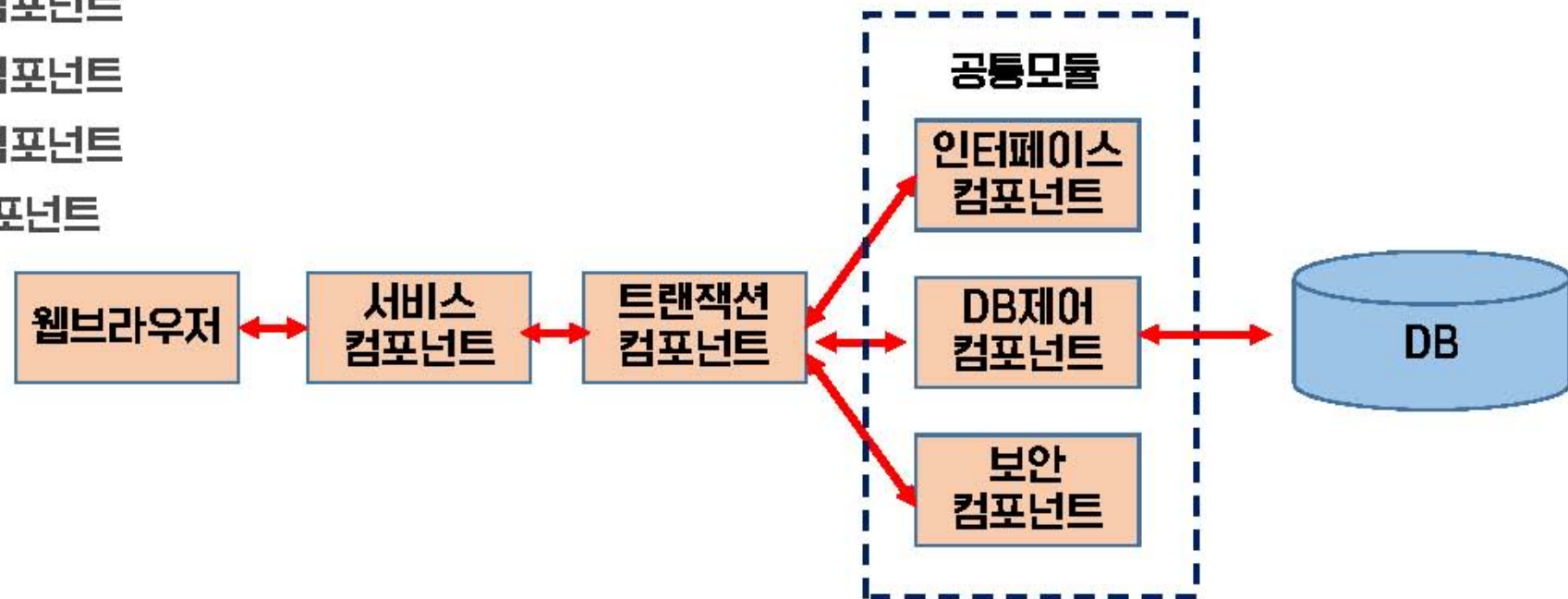
## 3. 알고리즘 구현 단계

- 단위 기능별 요구 사항들을 프로그래밍 언어를 이용하여 모듈로 구현
- 디바이스 드라이버 모듈, 네트워크 모듈, 파일 모듈, 메모리 모듈, 프로세스 모듈
- 단위 기능들이 사용자의 요구와 일치하는지 확인하는 과정 필요

# 소프트웨어 개발

## 단위모듈 종류

- 화면모듈
- 화면에서 입력 받은 데이터 처리를 위한 서비스 컴포넌트
- 비즈니스 트랜잭션 컴포넌트
- 데이터베이스 접근 컴포넌트
- 내외부 인터페이스 컴포넌트
- 데이터 암호/복호화 컴포넌트



## 단위모듈 구현시 고려사항

- 모듈 내 구성요소들 간의 응집도는 높이고 모듈간의 결합도는 낮춤
- 공통모듈 구현 -> 개별 단위모듈 구현(재사용)
- 예외처리 로직을 고려하여 구현



# 소프트웨어 개발

## 단위 모듈 테스트(Unit Test)

- 소프트웨어 개발 후 테스트 가능한 최소단위 기준으로 결함을 찾고, 기능을 검증하는 테스트 활동
- 소스 코드의 특정 모듈이 의도된 대로 정확히 작동하는지 검증하는 절차

## 단위 모듈 테스트의 장점

- **문제점 발견의 용이성**
  - 모듈 각각의 부분이 정확하게 동작하는지 확인하여 문제점을 발견하기 용이
- **구조 변경의 용이성**
  - 기능은 동일하게 유지하면서 프로그램의 구조를 변경하는 작업 가능
- **통합의 용이성**
  - 각 모듈을 검증하고, 합쳐서 다시 검증하는 통합 테스트에서 매우 유용(상향식(Bottom-Up) 방식)



# 소프트웨어 개발

## 테스트 방법 비교

Black Box	White Box
사용자 관점	개발자 관점
프로그램의 외부 사용자 요구사항 명세를 보면서 테스트, 주로 구현된 기능을 테스트 한다. (동치분할 검사, 경계값 분석)	프로그램의 내부 로직(수행 경로 구조, 루프 등)을 보면서 테스트를 수행한다 (기초 경로검사, 제어 구조 검사)
문제의 검증과 프로그램 구조에 관련된 결과에 중점을 두고 시험	각 모듈이 하나의 시스템 단위로서 올바른 기능을 수행하는지 여부 테스트
테스트 쉬움	오류 발견이 쉬움
해당 기능의 작동 여부 테스트	소스코드의 논리적인 경로 테스트

## 문제풀이

- White Box Testing의 설명으로 옳지 않은 것은?
- ① Base Path Testing, Boundary Value Analysis가 대표적인 기법이다.
  - ② Source Code의 모든 문장을 한번 이상 수행함으로써 진행된다.
  - ③ 모듈 안의 작동을 직접 관찰할 수 있다.
  - ④ 산출물의 각 기능별로 적절한 프로그램의 제어구조에 따라 선택, 반복 등의 부분들을 수행함으로써 논리적 경로를 점검한다.

(2020년 1, 2회 정보처리기사 필기 기출문제 소프트웨어 설계)

## 단위 모듈 테스트 환경

: 스텝(Stub)과 드라이버(Driver) 활용 테스트

- 사용자용 화면이 없는 경우 스텝 혹은 드라이버를 활용한 단위모듈 테스트
- 서비스 컴포넌트, 비즈니스 컴포넌트 등 하위 모듈 개발이 안된 경우에도 스텝과 드라이버를 활용한 단위 테스트 가능
- 테스트 드라이버 (Test Driver): 테스트 대상이 되는 모듈의 상위 가상 모듈
- 테스트 스텝 (Test Stub): 테스트 대상이 되는 모듈의 하위 가상 모듈

## 21 통합 구현 관리

### 통합 개발 환경(Integrated Development Environment, IDE) 도구

- 코딩, 디버그, 컴파일, 배포 등 프로그램 개발에 관련된 모든 작업을 하나의 프로그램 안에서 처리하는 환경을 제공하는 소프트웨어이다. 최근의 통합 개발 환경(IDE)은 그래픽 사용자 인터페이스(GUI) 응용 프로그램 개발용 고속 개발 도구가 많다.(위키백과)
- 테스트할 때 오류가 발생한 부분을 시각화 하므로 수정 용이
- 외부의 다양한 서비스와 연동, 개발에 편의를 제공

# 소프트웨어 개발

## IDE 도구의 종류

	개발사	플랫폼	운영체제	지원언어
<b>이클립스(Eclipse)</b>	IBM, 이클립스 재단	크로스 플랫폼(자바)	윈도, 리눅스, 솔라리스, AIX, 맥OS X	자바, C, C++, PHP, JSP
Dev-C++	Bloodshed Software	운영체제	플랫폼	C, C++
비주얼 스튜디오	마이크로소프트	Win32, win64	윈도 시리즈	비주얼 베이직, 비주얼 베이직 닷넷, 비주얼 C++등
엑스 코드(Xcode)	Apple	Mac, iPhone	MacOS, iOS	C, C++, C#, Java, AppleScript

(위키백과)

# 소프트웨어 개발

## 협업 도구

- 개발자들이 서로 다른 작업 환경에서 원활히 업무를 수행할 수 있도록 도와주는 도구.
- 기본적으로는 소프트웨어형(SaaS) 클라우드 서비스를 기반으로 팀 커뮤니케이션, 할 일 및 일정 관리, 파일 공유, 노트 작성, 프로젝트 관리 등 지식노동자들의 원활한 업무를 도와주는 서비스를 의미한다. (나무위키)

# 소프트웨어 개발

## 협업 도구의 기능

1. **전사관리** (전자결재, 조직도 등)
2. **프로젝트 관리**
  - 업무의 시각화 (캘린더, 타임라인, 간트차트, 대시보드)
  - 업무 관리 기능
  - 리마인더 기능 (예정된 일정, 업무 등)
3. **자체 드라이브 공간, 에디터 지원**
4. **문서 공유 지원**
  - 구글드라이브, 드롭박스 등 외부 서비스와 연동
5. **커뮤니케이션**
6. **타 협업툴간 연동 지원**
  - 구글 캘린더, 구글 스프레드시트



# 소프트웨어 개발

## 협상관리 도구

: 시스템 형상 요소(개발 소스, 산출물 문서 등)의 기능적 특성이나 물리적 특성을 문서화하고 그들 특성의 변경을 관리하며, 변경의 과정이나 실현 상황을 기록·보고하여 지정된 요건이 충족되었다는 사실을 검증하는 것이나 또는 그 과정을 위한 기능을 제공하는 도구

- **협상**

: 구조도, 프로그램 코드, 코드를 담은 디스크, 설계 사양서, 지침서 등 소프트웨어 개발에 필요한 모든 것.

- **협상관리 프로세스**

식별 : 소프트웨어가 변경될 만한 부분을 미리 정의한다.

버전관리 : 변경전과 변경 후를 정확히 기록한다.

변경관리 : 변경으로 인한 성능 및 품질을 평가한다.

형상감사 : 제대로 변경됐는지 확인한다.

보고 : 변경 사항을 문서화한다.

([www.egovframe.go.kr](http://www.egovframe.go.kr) > wiki > doku)

# 소프트웨어 개발

## 형상관리도구의 주요기능

- 체크인 : 개발자가 수정한 소스를 저장소로 업로드
- 체크아웃 : 저장소로부터 최신 버전을 개발자 PC로 다운로드
- 커밋 : 개발자가 소스를 업로드 후 최종적인 사항을 형상 관리 서버에 반영시키는 기능

## 형상관리도구의 종류

- Client/Server 타입 : Subversion(SVN), CVS, Perforce, ClearCase, TFS
- 분산저장소 타입 : Git, Mercurial, Bitkeeper, SVK, Darcs
- Folder 공유 타입 : RCS, SCCS

# 소프트웨어 개발

## 형상 관리 도구 종류

### 1) CVS (Concurrent Version System)

- 90년에 출시된 무료 서버-클라이언트 형상관리 시스템.
- 파일 전체를 저장하는 것이 아니라 변경사항만을 저장, 속도 느림.

### 2) SVN (Subversion)

- 형상관리/소스관리 툴.
- 커밋 실패 시 롤백 기능을 지원.

### 3) Git

- 분산형 방식
- 저장소의 완전한 복사본을 로컬에 저장. 처리속도가 빠르지만 대용량 코드 관리에 부적절.

### 4) Perforce(P4D)

- 빠른 속도, 빠른 Merge가 가능하며 큰 리소스 관리에 좋다.
- 유료, 파일명이 바뀌면 히스토리 추적이 곤란함

## 문제풀이

- 형상 관리 도구의 주요 기능으로 거리가 먼 것은?

- ① 정규화(Normalization)
- ② 체크인(Check-in)
- ③ 체크아웃(Check-Out)
- ④ 커밋(Commit)

(2020년 3회 정보처리기사 필기 기출문제 소프트웨어 개발)

## 문제풀이

- 소프트웨어 형상관리의 대상으로 거리가 먼 것은?
  - ① 소스 레벨과 수행 형태인 컴퓨터 프로그램
  - ② 숙련자와 사용자를 목표로 한 컴퓨터 프로그램을 서술하는 문서
  - ③ 프로그램 내에 포함된 자료구조
  - ④ 시스템 개발 비용

(2015년 정보처리기사 필기 기출문제 소프트웨어 공학)

## 문제풀이

- 제품 소프트웨어의 형상 관리 역할로 틀린 것은?
  - ① 형상관리를 통해 이전 리비전이나 버전에 대한 정보에 접근 가능하여 배포본 관리에 유용
  - ② 불필요한 사용자의 소스 수정 제한
  - ③ 프로젝트 개발비용을 효율적으로 관리
  - ④ 동일한 프로젝트에 대해 여러 개발자 동시개발 가능

(2020년 3회 정보처리기사 필기 기출문제 소프트웨어 개발)

## 문제풀이

- 소프트웨어 형상 관리의 의미로 적절한 것은?
  - ① 비용에 관한 사항을 효율적으로 관리하는 것
  - ② 개발 과정의 변경 사항을 관리하는 것
  - ③ 테스트 과정에서 소프트웨어를 통합하는 것
  - ④ 개발 인력을 관리하는 것

(2020년 1, 2회 정보처리기사 필기 기출문제 소프트웨어 개발)