

ICT의 가치를 이끄는 사람들!!
ICT의 가치를 이끄는 사람들!!

125회

정보관리기술사 기출풀이 4교시

국가기술자격 기술사 시험문제

정보처리기술사 제 125 회

제 4 교시

분야	정보처리	종목	정보관리기술사	수험 번호	성명
----	------	----	---------	----------	----

※ 다음 문제 중 4 문제를 선택하여 설명하시오. (각 25 점)

- 금융 환경의 변화로 최근 은행권 공동 오픈 플랫폼을 구축하여 핀테크서비스가 활성화 되고 있다. 다음 내용에 대하여 설명하시오.

- 오픈 플랫폼의 개념
- 오픈 플랫폼의 주요기술인 Open API
- 오픈 플랫폼의 제공 기능

- 정보통신분야 정보이론(Information Theory)과 샤논(Shannon)의 정리에 대하여 다음을 설명하시오

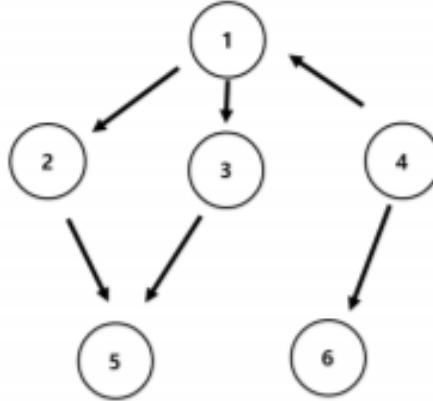
- 정보이론의 개념
- 샤논의 제 1 정리 및 제 2 정리
- 샤논-하틀리(Shannon-Hartley)의 정리

- 한 공장에서 스마트 팩토리(Smart Factory) 적용을 통한 생산시간 단축을 확인하려고 한다. 기존 공정에서의 제품 생산시간이 정규분포를 따르며 평균이 30 분, 표준 편차가 5 분인 것으로 알려져 있다. 스마트 팩토리가 적용된 공정의 제품에 대한 생산시간을 49 개 랜덤 추출하여 관측한 결과 평균이 28.5 분이었다. 스마트 팩토리가 적용된 공정의 생산시간이 표준편차가 5 분인 정규분포를 따른다고 할 때, 스마트 팩토리를 통해 생산 시간이 단축되었는지 가설을 설정하고 검정하시오. (단, 유의수준 5%, $Z_{0.05}=1.645$)

4. 방향성 비순환 그래프(Directed Acyclic Graph)에 대하여 다음 물음에 답하시오.

가. 방향성 비순환 그래프의 개념과 특징

나. 아래 방향성 비순환 그래프에 대하여 위상정렬(Topology Sort)을 실시하고 결과값 제시



5. 절차지향 프로그래밍(Procedure Oriented Programming) 방법과 객체지향 프로그래밍(Object Oriented Programming) 방법을 비교 설명하시오.

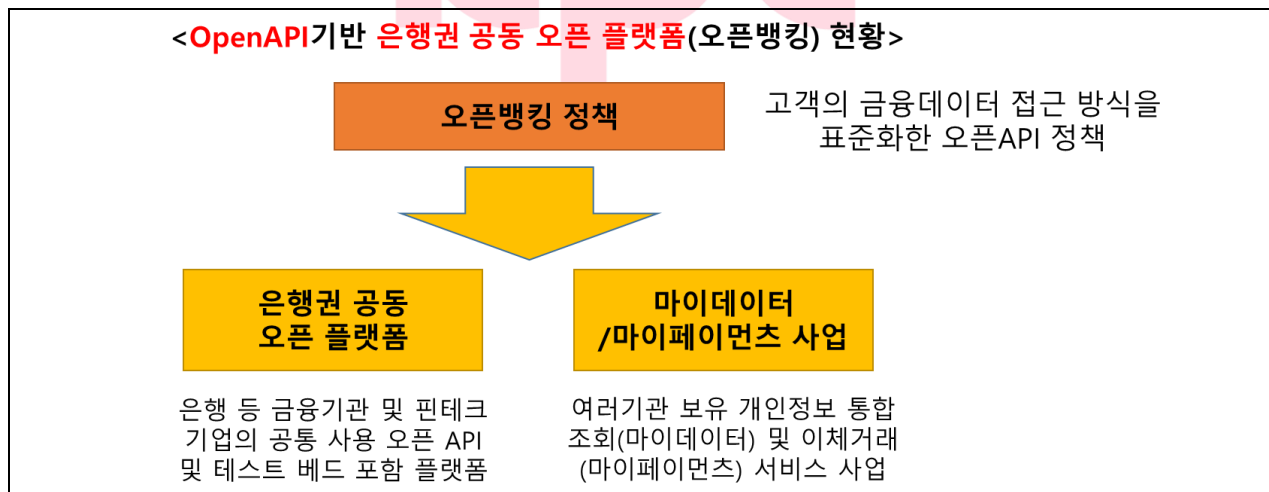
6. 산업제어시스템(Industrial Control System)의 퍼듀(Purdue) 모델에 대하여 다음을 설명하시오

가. 퍼듀 모델의 개념

나. 퍼듀 모델의 계층과 계층별 특징

문 제	4-1 금융 환경의 변화로 최근 은행권 공동 오픈 플랫폼을 구축하여 핀테크서비스가 활성화 되고 있다. 다음 내용에 대하여 설명하시오.		
	가. 오픈 플랫폼의 개념		
	나. 오픈 플랫폼의 주요기술인 Open API		
	다. 오픈 플랫폼의 제공 기능		
출 제 영 역	디지털서비스	난 이 도	★★★★☆
출 제 배 경	2016 년 8 월부터 금융 환경에 공동 오픈플랫폼이 구축되고 데이터 3 법 개정 등 변화에 따라 국내 오픈뱅킹 정책이 인프라 개방 및 법 정비, 새로운 서비스업 신설되는 등 추진이 활발히 되고 있어 이에 대한 이해도 확인		
출 제 빈 도	120 관리 4, 107 관리 3, 108 컴시응 1		
참 고 자 료	<ul style="list-style-type: none"> - 오픈뱅킹센터 홈페이지, 금융결제원 - 핀테크포털 홈페이지, 금융결제원/코스콤/한국예탁결제원 - 오픈 API 기반의 금융생태계 변화와 시사점, 보험연구원 		
Key word	오픈 API, 테스트베드, 오픈뱅킹, 잔액조회 API, 거래내역조회 API, 계좌실명조회 API, 입금이체 API, 출금이체 API, 송금인 정보조회 API, 수취조회 API, 사용자 인증 API, 관리 API, OAuth2.0, RestAPI, JSON		
풀 이	권태용(120 회 정보관리기술사)		

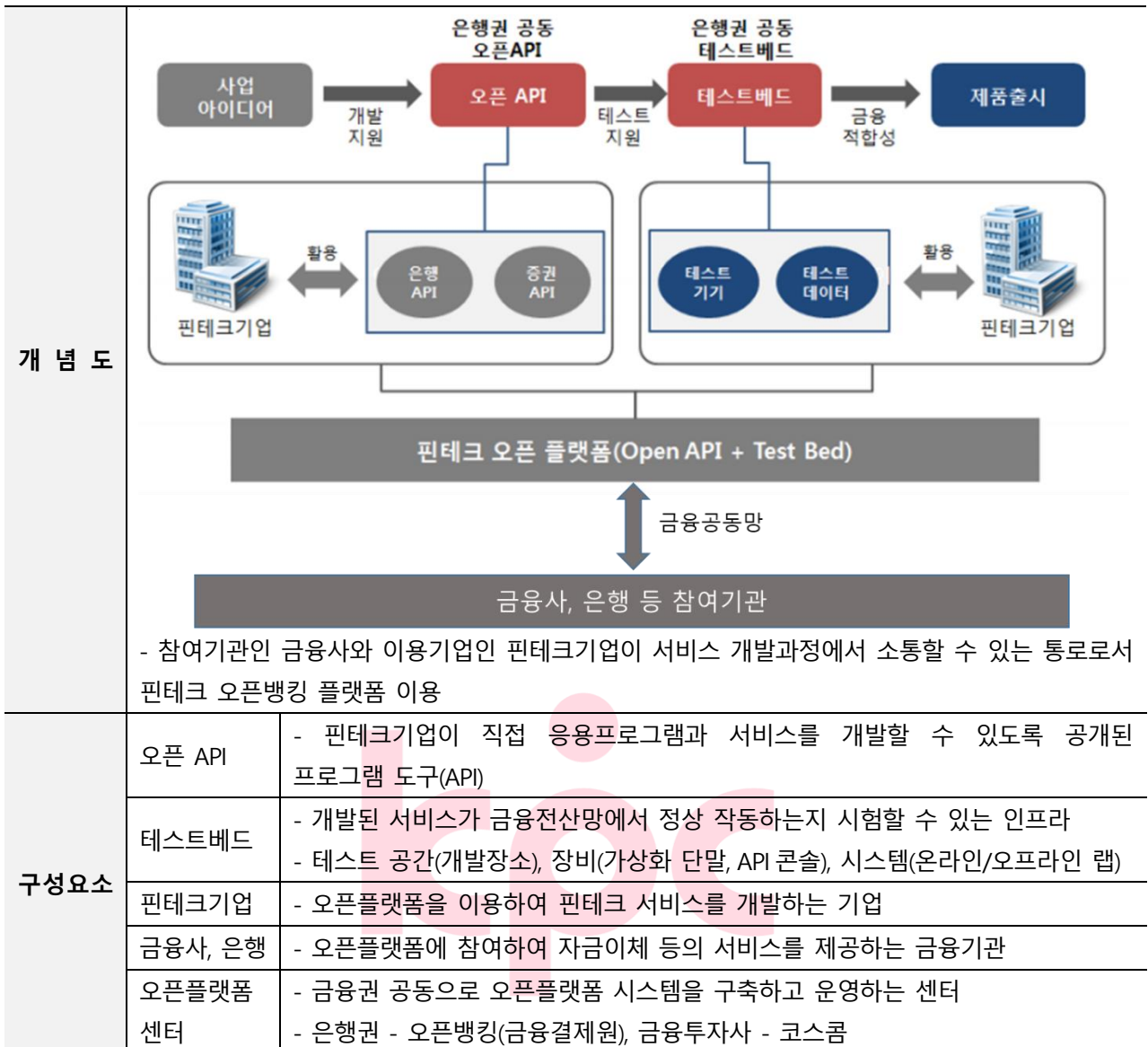
1. 은행권 공동 오픈 플랫폼 등장 및 현황



- 데이터3법 개정, 공인인증서 의무사용 폐지와 더불어 은행권 공동 오픈 플랫폼 등장으로 핀테크서비스 활성화가 핀테크서비스가 다양하게 활성화 되고 있음

2. 핀테크서비스 활성화 위한 오픈 플랫폼의 개념

구 분	설 명
개 념	- 금융회사 내부의 금융 서비스를 표준화한 API 형태로 제공하는 OpenAPI 와 개발된 핀테크 서비스를 금융전산망에서 시험해볼 수 있는 테스트베드로 구성된 금융 및 핀테크 공동 플랫폼



- 오픈 플랫폼의 주요 기술인 OpenAPI는 시장의 요구에 따라 지속적으로 확장되고 있음

3. 오픈 플랫폼의 주요 기술 OpenAPI

가. 오픈 플랫폼의 주요 기술 OpenAPI 종류

구 분	OpenAPI	설 명
서비스 API	잔액조회 API	- 본인계좌에 대한 잔액 및 출금가능금액을 조회
	거래내역조회 API	- 본인계좌에 대한 잔액 및 거래내역 조회
	계좌실명조회 API	- 자금을 수취할 수취인 또는 출금할 사용자 계좌의 정상여부 및 실명 실시간 조회
	입금이체 API	- 지급계좌에서 자금을 인출하여 수취인 계좌로 실시간 입금
	출금이체 API	- 출금에 동의한 사용자 계좌에서 출금
	송금인 정보조회 API	- 출금이체 신청을 한 사용자 계좌의 정상여부를 실시간 조회
	수취조회 API	- 입금이체 요청 전 수취계좌의 입금가능여부 및 수취인 성명 조회
인증/관리	사용자 인증 API	- 사용자의 인증 및 동의를 얻고 계좌를 등록하는 기능 제공

API		- OAuth2.0 토큰 기반 인증
	관리 API	- 이용기관 요구를 반영하는 관리 기능 제공 - 참가은행상태조회 API 제공

나. 오픈 플랫폼의 주요 기술 OpenAPI 예시

구 분	설 명
서비스 API	<p>- 입금이체 API에 대한 연동과정 예시</p>
사용자 인증 API	<p>- OAuth 기반 인증 API에 대한 연동과정 예시</p>
관리 API	

- 참가은행 서비스 상태(거래가능 Y, 장애 D, 개시전 L, 종료처리 E) 조회 API 에 대한 연동과정
- 각각의 OpenAPI 를 조합하여 하나의 기능 제공

4. 오픈 플랫폼의 제공 기능

가. OpenAPI 를 조합한 오픈 플랫폼의 제공 기능

제공 기능	API 조합	설 명
종합 자산 관리	잔액조회 API + 거래내역조회 API	- 계좌의 입출금내역을 조회하여 개인별 맞춤형 상품 추천 - 자산상태 모니터링 제공
간편 송금	출금이체 API + 입금이체 API	- 개인 간에 조의금, 용돈, 점심값 등의 각종 대금을 전달
사용자 인증	입금이체 API + 계좌실명조회 API	- 이용고객 계좌의 실명을 확인 - 계좌에 일정금액을 입금(1 원)하여 고객 진위여부 검증
모임 회비관리	출금이체 API + 입금이체 API + 거래내역조회 API	- 동호회, 각종 모임 및 행사에서 회원들의 회비 납부 지원 - 회비납부내역 조회 관리

나. 기술요소 관련 오픈 플랫폼의 제공 기능

구 분	설 명
Rest API	- HTTP 통신에서 자원에 대한 요청을 Resource 와 Method 로 표현하여 전달하는 방식
OAuth 인증	- 표준 사용자 인증 프로토콜 (2.0)
HTTP 통신	- HTTP 프로토콜을 활용한 웹 서비스 기반의 통신
JSON	- 자바스크립트언어에서 객체(object)를 표현하기 위한 데이터 포맷

- 핀테크기업에서 참신한 아이디어를 기반으로 다양한 제공 기능을 만들도록 오픈 플랫폼의 발전 필요

"끝"

문 제	4-2 정보통신분야 정보이론(Information Theory)과 샤논(Shannon)의 정리에 대하여 다음을 설명하시오.		
	가. 정보이론의 개념		
	나. 샤논의 제 1 정리 및 제 2 정리		
	다. 샤논-하틀리(Shannon-Hartley)의 정리		
출 제 영 역	네트워크	난 이 도	★★★★★
출 제 배 경	샤논이 확립 기여한 정보이론은 현대에 5G 등 통신분야를 포함한 인공지능이론, 패턴인식이론 등 여러 IT 분야에서 적용되어 있어 근간이 되는 샤논의 정리까지 이해도 확인 (정보통신기술사에 관련된 문제로 네트워크 도메인에서 출제)		
출 제 빈 도	미출제		
참 고 자 료	- 정보통신기술용어해설 정보이론, 샤논 정리, - [정보이론] 정보량과 엔트로피의 의미, bskyvision 블로그		
Key word	정보량, 채널용량, 엔트로피, 정보원 부호화 정리, 채널 부호화 정리, 채널 용량 정리, 잡음/오류, 신호 대 잡음비, 수신신호전력, 대역폭, 관련 수식		
풀 이	권태용(120 회 정보관리기술사)		

1. 정보통신분야 정보이론(Information Theory)의 발전과정

1928 년 하틀리 기초작업

1948 년 샤논 『통신의 수학적 이론』 논문을 통해 증명



- 정보이론의 아버지로 불리는 샤논(Claude Shannon)의해 정보이론의 여러 개념과 이론이 제시되었고, 실제적인 디지털회로 설계이론 초석마련

2. 신호에 존재하는 정보량 측정 이론, 정보이론(Information Theory)의 개념

개 념	- 통신시스템에서 전달되는 정보의 정량화 및 그 처리, 전달 용량의 한계 및 기준을 설명하는 이론	
	- 정보량(정보의 측정) 및 채널용량(전송 제약/한계)에 대한 이론	
구 성	- 의의 : 정보가 갖는 양의 척도 엔트로피(불확실성) 도입, 정보의 발생에서 전달까지 구조를 종합, 전송 오류, 잡음에 지배되는 메시지의 전송 확률 연구	
	정보량	- 놀람의 정도, 정보량은 발생 확률에 반비례 - “잘 발생하지 않는 사건은 자주 발생하는 사건보다 정보량이 많다.”

		$I(x_j) = -\log_a P(x_j)$ <p> $I(x_j)$: 정보량 a : 정보단위 $P(x_j)$: x_j 발생확률 </p>
	채널용량	<ul style="list-style-type: none"> - 신뢰성 있는 통신이 가능한 최대 정보량 - 채널(통신로)이 수용 가능한 데이터 통신 용량(Capacity)
	엔트로피	<ul style="list-style-type: none"> - 정보량의 기댓값(평균), 평균 놀람의 정도(불확실성), 발생 확률에 반비례 - "모든 사건이 비슷한 확률로 발생할 때 엔트로피가 크다." $H(x) = E\{I(x_j)\} = - \sum_{j=1}^n P(x_j) \log_2 P(x_j)$ <p> $H(x)$: 엔트로피 $E\{\}$: 기댓값 $I(x_j)$: 정보량 </p>

- 정보이론에 근간이 되는 정리에는 샤논의 제 1 정리, 제 2 정리, 샤논-하틀리 정리가 존재함

3. 샤논의 제 1 정리 및 제 2 정리

가. 샤논의 제 1 정리

구 분	설 명
개 념	- 무손실 압축의 하한을 제시한 <u>정보원 부호화 정리</u> (Source Coding Theorem)
수 식	$L \geq H$ <p> L : 정보원 평균부호길이 H : 엔트로피 </p> <ul style="list-style-type: none"> - 정보원 평균부호길이 L의 하한은 정보원 엔트로피 H가 됨
의 미	<ul style="list-style-type: none"> - 정보를 디지털화하고 압축하는 과정에서 평균코드길이의 최소화를 지향함(소스코딩의 효율화) - 평균코드길이를 얼마까지 줄일 수 있는지 소스 부호화 문제를 정리함

나. 샤논의 제 2 정리

구 분	설 명
개 념	- 에러/잡음을 최소화 할 수 있는 부호화 과정이 반드시 존재하며 반대로 어떤 부호화 기술을 사용하더라도 에러가 존재한다는 사실을 증명한 <u>채널 부호화 정리</u> (Channel Coding Theorem)
수 식	$R < C$ <p> R : 정보 전송률 C : 채널용량 </p> <ul style="list-style-type: none"> - 대역 제한된 채널로 입력되는 정보 전송률 R 이 채널용량 C 보다 작다면, 에러율을 0 으로 접근시키는 부호화가 반드시 존재함 $R > C$ <p> R : 정보 전송률 C : 채널용량 </p> <ul style="list-style-type: none"> - 반대로 입력되는 정보 전송률 R 이 채널용량 C 를 초과하면 에러율은 어떤 유한값 이하로 떨어질 수 없음
의 미	- ($R < C$) 적절한 부호화 기술만 사용하면 오류 최소화 가능 → 잡음 존재 무시 가능 정도 송수신

- ($R > C$) 어떤 부호화 기술을 사용하더라도 에러/잡음이 반드시 존재함

4. 샤논의 제 3 정리, 샤논-하틀리(Shannon-Hartley) 의 정리

구 분	설 명
개 념	- 정보이론 관점에서 신뢰성 있는(오류 없는) 통신이 가능한 최고 전송률인 채널용량에 대한 정리 (Channel Capacity Theorem)
수 식	$C = W \log_2(1 + S/N)$ <p> C : 채널용량(bps) W : 가용 대역폭(Hz) S : 수신 신호 전력 N : 잡음 전력 S/N : 신호 대 잡음비(dB) </p> <p> - 채널용량(정보 전송률)의 한계를 대역폭, 수신신호, 잡음으로 표현 - 정보 전송률은 대역폭과 수신신호전력에 비례하고, 잡음전력에 반비례 </p>
의 미	- 잡음이 없다면 (또는 신호 전력이 크다면) 임의 대역폭에서도 채널용량을 거의 무한 증가 가능 - 잡음이 있다면 (또는 신호 전력이 작다면) 대역폭을 아무리 증가시켜도 채널용량 증가 불가능 - 채널 용량 C 에 도달하는 이론적인 한계치를 제시 (잡음이 존재하는 곳에서 신뢰 할만한 통신)

- 샤논이 확립한 정보이론은 5G 등 통신분야를 포함한 다양한 분야에 적용되고 있으며, 신호이론, 패턴인식이론, 언어이론, 인공지능이론으로 전개됨

“끝”

문 제	4-3 한 공장에서 스마트 팩토리(Smart Factory) 적용을 통한 생산시간 단축을 확인하려고 한다. 기존 공정에서의 제품 생산시간이 정규분포를 따르며 평균이 30 분, 표준 편차가 5 분인 것으로 알려져 있다. 스마트 팩토리가 적용된 공정의 제품에 대한 생산시간을 49 개 랜덤 추출하여 관측한 결과 평균이 28.5 분이였다. 스마트 팩토리가 적용된 공정의 생산시간이 표준편차가 5 분인 정규분포를 따른다고 할 때, 스마트 팩토리를 통해 생산 시간이 단축되었는지 가설을 설정하고 검정하시오. (단, 유의수준 5%, $Z_{0.05}=1.645$)		
출 제 영 역	알고리즘	난 이 도	★★★★☆
출 제 배 경	데이터 분석의 주요 기법인 통계적 가설 검정에 대한 이해와 실무 적용능력 확인		
출 제 빈 도	102 관리 4, 111 관리 1, 111 관리 3, 117 관리 1, 121 관리 1		
참 고 자 료	<ul style="list-style-type: none"> - 유의성 검정을 위한 기초통계 세미나자료, 부산청 시험분석센터 수입식품분석과 - 통계분석 가설검정 포스트, 김훈위험관리연구소 네이버 블로그 		
Key word	귀무가설, 대립가설, Z-score, Z 검정통계량, 유의수준, p-value, 임계치, 좌측검정, 채택, 기각, Z 검정통계량 수식, -2.1, 중심극한정리		
풀 이	권태용(120 회 정보관리기술사)		

1. 통계적 가설 검정의 개요

가. 통계적 가설 검정의 개념 및 구성요소

개 념	<ul style="list-style-type: none"> - 표본에서 얻은 사실을 근거로 모집단에 대한 가설이 맞는지 참/거짓을 통계적으로 검정하는 분석 방법 - Statistical Hypothesis Test 	
구성요소	귀무가설(H_0)	<ul style="list-style-type: none"> - 영가설(Null), 차이가 없다는 가정하에 시작하여 기각이 목표 - 반박하고 싶은 사실을 가설로 설정
	대립가설(H_1)	<ul style="list-style-type: none"> - 연구/실험가설(Alternative), 차이가 있다는 가정하에 시작하여 채택이 목표 - 주장 하고자 하는(입증하고 싶은) 사실을 가설로 설정 - 귀무가설과 대립가설은 반대이며, 배타적인 관계로 설정되어야 함
	유의수준(α)	- H_0 가 참일 때 H_1 을 채택하는 오류를 범할 최대 허용치
	p-value	- H_0 가 참일 때 표본(관측치)가 발생할 확률
	검정통계량	- 표본 통계량의 모수 근접 수치 (Z-score)
	임계치	- 기각/채택을 구분하는 검정통계량의 값
	기각역	- C_1 , 귀무가설을 기각하는 검정통계량의 범위
	채택역	- C_0 , 귀무가설을 채택하는 검정통계량의 범위

- 통계적 가설 검정 유형으로는 단측검정(좌측검정/우측검정)과 양측검정이 존재
- 해당 스마트 팩토리 사례에서는 생산시간 단축(미만)을 검정하므로 단측검정 중 좌측검정을 사용하여 검정

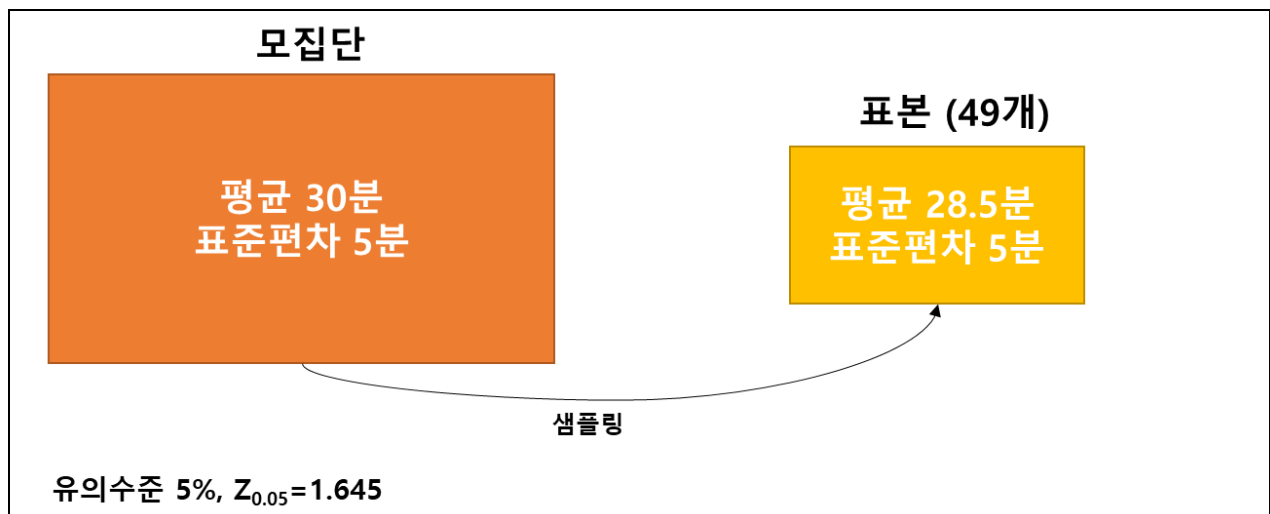
나. 통계적 가설 검정의 절차 및 수식

구 분	설 명
절 차	(1) H_0, H_1 가설 설정
	(2) 통계 분석 방법, 검정통계량 선택
	(3) 유의수준 설정
	(4) 통계분석, 검정통계량 계산 (p-value 계산)
	(5) 검정통계값과 임계치 비교 (p-value 와 유의수준 비교)
	(6) H_0 기각/채택 결정 $\rightarrow H_1$ 기각/채택 결정 (반증/귀류법 사용) H_0 기각 : 검정통계량 \geq 임계치 또는 $p\text{-value} \leq \alpha$ (좌측 검정인 경우 반대)
수 식	<p>Z검정 통계량 수식:</p> $Z_0 = \frac{(\bar{x} - \mu)}{\frac{\sigma}{\sqrt{n}}}$ <p> Z_0: Z 검정 통계량 \bar{x}: 표본 평균 μ: 모 평균 (귀무가설에서 주장하는 평균) σ: 모 표준편차 n: 표본 개수 </p>

- 위 절차 중 검정통계량과 임계치 비교 방법과 Z검정 통계량 수식을 토대로 제시된 사례에 대해 가설 검정

2. 스마트 팩토리 사례에 대한 가설 설정

가. 스마트 팩토리 사례 문제 분석



나. 스마트 팩토리 사례에 대한 가설 설정

가 설	설 명
귀무가설 H_0	<ul style="list-style-type: none"> - 기존 생산 시간보다 스마트 팩토리 적용 후, 생산 시간이 단축되지 않았다. - $\mu \geq \mu_0$ (스마트 팩토리 적용 후 생산 시간 \geq 기존 생산 시간) - (좌측검정) 검정통계량 \geq 임계치 이면 채택
대립가설 H_1	<ul style="list-style-type: none"> - 기존 생산 시간보다 스마트 팩토리 적용 후, 생산 시간이 단축되었다. - $\mu < \mu_0$ (스마트 팩토리 적용 후 생산 시간 $<$ 기존 생산 시간) - (좌측검정) 검정통계량 $<$ 임계치 이면 채택

- 귀무가설 H_0 를 기각 하고 채택가설 H_1 을 채택하여 가설을 검정

3. 스마트 팩토리 사례에 대한 통계적 가설 검정

가. 스마트 팩토리 사례에 대한 통계적 가설 검정 수행

절 차	설 명
통계분석 방법 선택	<ul style="list-style-type: none"> - 검정통계량을 이용하는 Z-검정 방법 선택 - Z-검정 : 모집단의 평균과 표준편차가 얼마라는 것이 알려져 있을 때, 새롭게 조사된 표본의 평균이 모집단 평균과 같은 지 추정하는 검정
유의수준	- 유의수준 5%와 $Z_{0.05}$ 임계치 -1.645 제시 (좌측 검정이므로 음수)
검정통계량 계산	$Z_0 = \frac{(\text{표본평균} - \text{모평균})}{\frac{\text{모 표준편차}}{\sqrt{\text{표본 개수}}}}$
	$Z_0 = \frac{(28.5 - 30)}{\frac{5}{\sqrt{49}}} = \frac{-1.5}{\frac{5}{7}} = -2.1$
검정통계량 임계치 비교	<p>- 표준정규분포 : 모평균이 0 이고 모표준편차가 1 인 정규분포</p> <p>- Z 검정통계량(Z-score)인 -2.1 과 임계치($Z_{0.05}$) -1.645 를 비교</p>

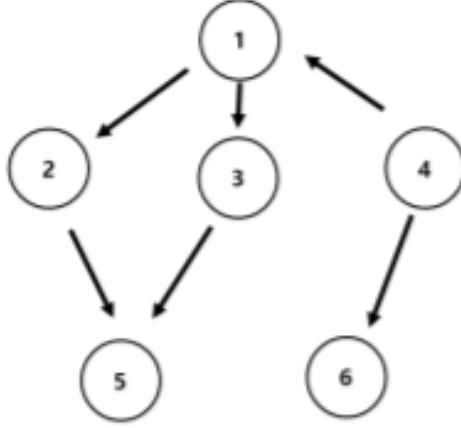
가설 기각/채택	<ul style="list-style-type: none"> - 귀무가설 채택 조건 = 검정통계량 > 임계치(-1.645) - 좌측 검정에서 검정통계량(-2.1) ≤ 임계치(-1.645) 가 성립되므로 - 귀무가설(H0) 기각, 대립가설(H1) 채택
-------------	--

나. 스마트 팩토리 사례에 대한 통계적 가설 검정 결론

가설검정 결론	<ul style="list-style-type: none"> - 스마트 팩토리 적용 후 생산 시간 (표본평균 28.5 분) < 기존 생산 시간 (모평균 30 분) - 위 결론에 대해 통계적 가설 검정 수행 결과, 검정통계량 ≤ 임계치 이므로 49 개 관측치에 대한 통계가 유의함(유의수준에 만족함) - 귀무가설 기각으로 "기존 생산 시간보다 스마트 팩토리 적용 후, 생산 시간이 단축되지 않았다." 가설이 거짓으로 검정 - 대립가설 채택으로 "기존 생산 시간보다 스마트 팩토리 적용 후, 생산 시간이 단축되었다." 가설이 참으로 검정 완료
------------	--

- Z-검정의 경우 중심극한정리에 의해 30 개 이상의 표본 크기에서 정규분포를 사용하며 해당 사례에서도 30 개 미만인 25 개로 설정 시, 검정통계량(-1.5) > 임계치(-1.645)가 되어 귀무가설이 채택되어 통계가 유의하지 않음
- "끝"



문 제	4-4 방향성 비순환 그래프(Directed Acyclic Graph)에 대하여 다음 물음에 답하시오.		
	가. 방향성 비순환 그래프의 개념과 특징 나. 아래 방향성 비순환 그래프에 대하여 위상정렬(Topology Sort)을 실시하고 결과값 제시		
			
출 제 영 역	알고리즘	난 이 도	★★★★★
출 제 배 경	블록체인 3.0의 미래 기술로 부각되는 방향성 비순환 그래프(DAG)의 이해와 위상정렬 계산 방법 숙지 여부를 확인		
출 제 빈 도	미출제		
참 고 자 료	- 『소프트웨어 알고리즘 - 위상 정렬 알고리즘』, 네이버 지식백과 - 방향성 비순환 그래프, 해시넷		
Key word	블록체인, $G=(V,E)$, 방향 있음, 순환 없음, 노드(정점, 꼭지점), 간선(엣지), 진입차수, 큐, 순서(위상) $4 \rightarrow 1 \rightarrow 6 \rightarrow 2 \rightarrow 3 \rightarrow 5$, $4 \rightarrow 1 \rightarrow 6 \rightarrow 3 \rightarrow 2 \rightarrow 5$, $4 \rightarrow 6 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 5$, $4 \rightarrow 6 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 5$, 방향성 비순환 그래프일 때 위상 정렬이 가능함		
풀 이	권태용(120 회 정보관리기술사)		

4. 방향성 비순환 그래프(Directed Acyclic Graph)의 의의

블록체인 1.0 비트코인	블록체인 2.0 Smart Contract	블록체인 3.0 방향성 비순환 그래프
- 통화로서 시작 단계	- 금융, 경제 분야 블록체인 - 이더리움	- 사회전반에 블록체인 영향 - 탈중앙화, 보안, 공개 장부 - IOTA(탱글), 로커스체인 (AWTC), 하이퍼캐시, ADK 등

- 방향성 비순환 그래프(DAG)는 기존 블록체인에서 제기된 문제점을 보완하기 위하여 사용되고 있으며, 미래의 블록체인 기술로 합의 과정에서 적용되고 있음

5. 방향성 비순환 그래프(Directed Acyclic Graph)의 개념과 특징

가. 방향성 비순환 그래프의 개념

개념	<ul style="list-style-type: none"> - 방향성이 존재하며 순환이 없는 그래프 - 정점들과 간선들로 구성되며 각 간선은 하나의 정점에서 다른 정점으로 방향을 가지고 있으며, 어떠한 정점 v에서 시작하여 끝내 다시 v로 돌아가지 못하는 비순환 그래프 - 위상정렬이 있는 유향 그래프
개념도	

나. 방향성 비순환 그래프의 특징

구분	특징	설명
일반적 특징	그래프	<ul style="list-style-type: none"> - $G = (V, E)$ - 꼭지점 집합 V와 간선 집합 E로 이루어져 있음 - 연결되어 있는 객체간의 관계 표현 가능
	방향성 존재	- E(간선)이 방향을 가지고 있음
	비순환	<ul style="list-style-type: none"> - Cycle(사이클)이 존재하지 않음 - 출발 노드에서 시작해서 지나간 노드로 다시 돌아올 수 없음
블록체인 특징	빠른 속도	<ul style="list-style-type: none"> - 블록 생성 없이 트랜잭션이 다른 트랜잭션을 검증하여 매우 빠른 속도 수행 - 검증 과정이 병렬구조라 사용자가 많을수록 더 빨라짐
	수수료 없음	<ul style="list-style-type: none"> - 작업증명을 사용하지 않아 마이너가 존재하지 않음 - 블록 개념 미존재하며, 채굴자가 없어 수수료가 발생하지 않음
	높은 확장성	- 블록 대비 트랜잭션 수가 넘치는 병목현상이 발생하지 않고 오히려 검증 신뢰도가 좋고 확장성이 좋음
	확정합의 문제	- 파셜 오더링 방식의 원장으로 전체 원장 상태가 존재하지 않아 확정합의에 적합하지 않은 구조, 완전한 탈중앙화가 아님

- 방향을 가지고 사이클이 없는 방향성 비순환 그래프(DAG)에 대해서만 위상정렬이 가능함

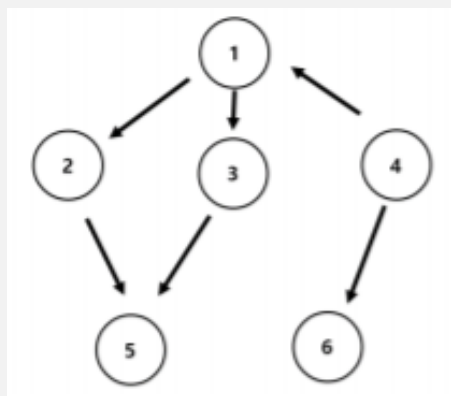
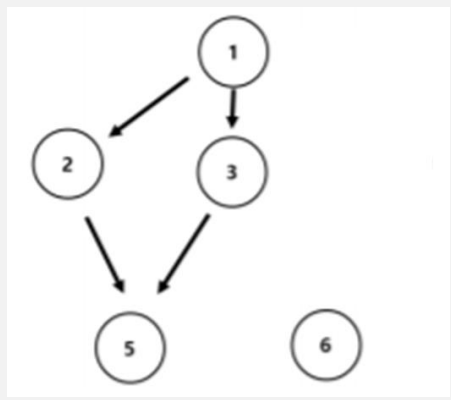
6. 위상정렬(Topology Sort) 실시 과정

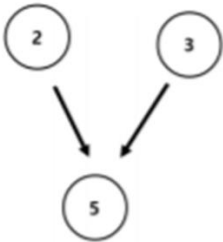

가. 위상정렬 개념 및 일반적인 실시 과정

구분	설명
개념	<ul style="list-style-type: none"> - 순서가 정해져 있는 작업을 차례대로 수행해야 할 때 그 순서(위상)를 결정해주기 위해 사용하는 정렬 알고리즘 - 방향 그래프의 꼭지점들(Vertex)을 변의 방향을 거스르지 않도록 나열하는 알고리즘

의사코드	$G=(V,E)$ // G 는 그래프, V 는 꼭지점(vertex) 정점 노드, E 는 엣지 간선 모든 V 의 진입차수를 계산 // 진입 차수는 노드로 들어오는 다른 노드 갯수 if (V 가 하나도 없다면) Then 위상 정렬 종료 // 완료 else if (진입 차수가 0인 정점이 하나도 없다면) Then 위상 정렬 불가능 // 사이클 존재하기 때문 (방향성 비순환 그래프가 아니기 때문) else { V (진입 차수 == 0) 출력 // 큐 스택 활용하는 방법 V (진입 차수 == 0)와 V 와 연결된 E 삭제 각 정점의 차수 진입차수 수정 }
실시과정	- 큐(Queue)를 사용한 위상정렬 실시 과정 (1) 각 노드들의 진입 차수 계산 (2) 진입 차수가 0인 노드를 큐에 삽입 (3) 큐에서 노드를 꺼내 연결된 간선 제거 (4) 제거로 인해 진입 차수가 0이 된 노드를 큐에 삽입 (5) 3~4번 반복하며, 큐가 비었으면 종료

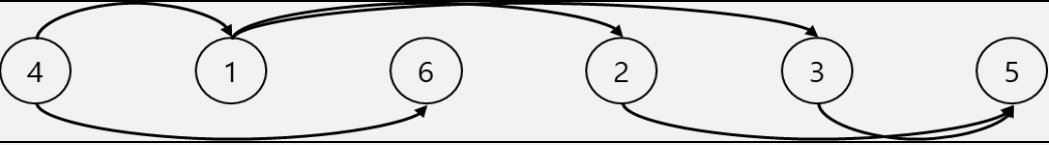
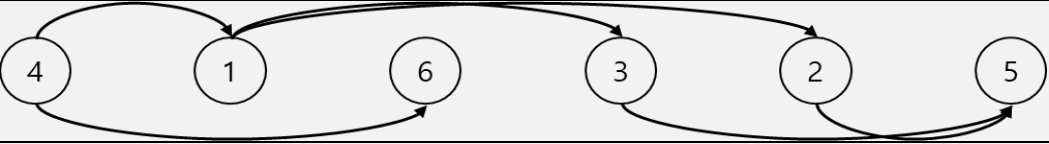
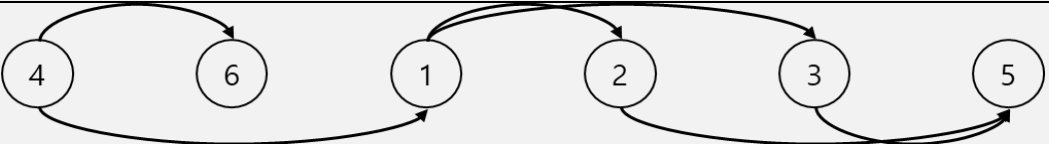
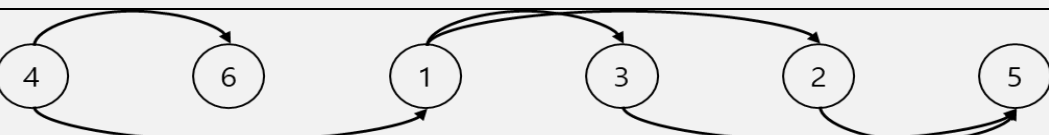
나. 주어진 문제의 위상정렬 실시 과정

#	그래프	설 명														
시작		<table><tr><th>노드</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th></tr><tr><th>진입차수</th><td>1</td><td>1</td><td>1</td><td>0</td><td>2</td><td>1</td></tr></table> <ul style="list-style-type: none">- $G = (V, E)$ / V 노드가 6 개이며, E 간선이 6 개인 그래프- 주어진 그래프는 방향성 비순환 그래프로 위상정렬 적용이 가능함- 진입 차수를 계산하면 진입차수가 0 인 노드는 4로 나타남- 정렬 값 : $4 \rightarrow$- 노드 4와 연결된 간선을 제거함	노드	1	2	3	4	5	6	진입차수	1	1	1	0	2	1
	노드	1	2	3	4	5	6									
진입차수	1	1	1	0	2	1										
1		<table><tr><th>노드</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th></tr><tr><th>진입차수</th><td>0</td><td>1</td><td>1</td><td>-</td><td>2</td><td>0</td></tr></table> <ul style="list-style-type: none">- 진입차수를 다시 계산하면 진입차수가 0 이 된 노드는 노드 1과 노드 6으로 나타남- 노드 1과 노드 6은 순서와 상관없이 모두 정렬 가능- 정렬 값 : $4 \rightarrow 1 \rightarrow 6$ $4 \rightarrow 6 \rightarrow 1$- 노드 1과 노드 6이 연결된 간선을 제거함	노드	1	2	3	4	5	6	진입차수	0	1	1	-	2	0
노드	1	2	3	4	5	6										
진입차수	0	1	1	-	2	0										

2		<table><tr><th>노드</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th></tr><tr><th>진입차수</th><td>-</td><td>0</td><td>0</td><td>-</td><td>2</td><td>-</td></tr></table> <ul style="list-style-type: none">- 진입차수를 다시 계산하면 진입차수가 0 이 된 노드는 노드 2와 노드 3으로 나타남- 노드 2와 노드 3의 순서와 상관없이 모두 정렬 가능- 정렬 값 : 4 → 1 → 6 → 2 → 3 / 4 → 1 → 6 → 3 → 2 4 → 6 → 1 → 2 → 3 / 4 → 6 → 1 → 3 → 2- 노드 2와 노드 3이 연결된 간선을 제거함	노드	1	2	3	4	5	6	진입차수	-	0	0	-	2	-
	노드	1	2	3	4	5	6									
진입차수	-	0	0	-	2	-										
완료		<table><tr><th>노드</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th></tr><tr><th>진입차수</th><td>-</td><td>-</td><td>-</td><td>-</td><td>0</td><td>-</td></tr></table> <ul style="list-style-type: none">- 진입차수를 다시 계산하면 진입차수가 0 이 된 노드는 노드 5로 나타남- 정렬 값 : 4 → 1 → 6 → 2 → 3 → 5 4 → 1 → 6 → 3 → 2 → 5 4 → 6 → 1 → 2 → 3 → 5 4 → 6 → 1 → 3 → 2 → 5- 노드 5를 제거하면 노드가 남지 않아 정렬 종료	노드	1	2	3	4	5	6	진입차수	-	-	-	-	0	-
노드	1	2	3	4	5	6										
진입차수	-	-	-	-	0	-										

- 위상정렬은 여러 개의 답이 존재 가능한 특징을 가지고 있음

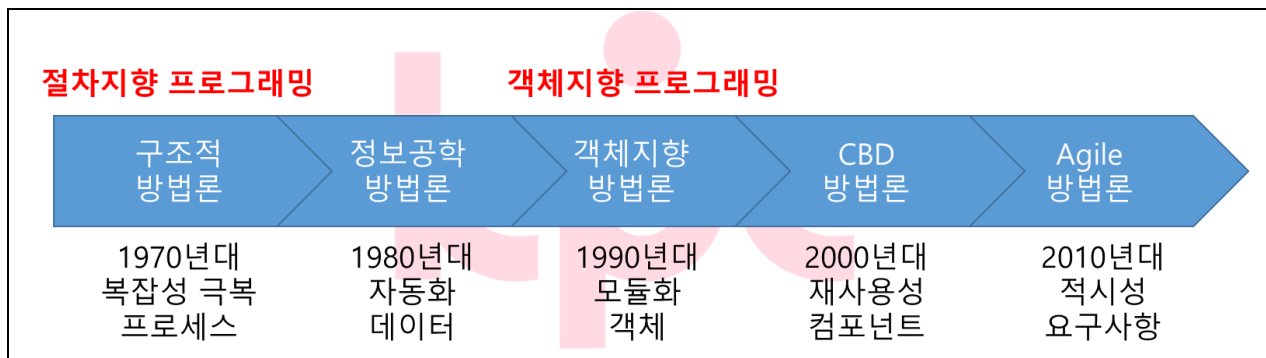
7. 위상정렬(Topology Sort) 결과 값

결과	설 명
결과값 1	$4 \rightarrow 1 \rightarrow 6 \rightarrow 2 \rightarrow 3 \rightarrow 5$
	
결과값 2	$4 \rightarrow 1 \rightarrow 6 \rightarrow 3 \rightarrow 2 \rightarrow 5$
	
결과값 3	$4 \rightarrow 6 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 5$
	
결과값 4	$4 \rightarrow 6 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 5$
	

- 주어진 방향성 비순환 그래프(DAG)에 대한 위상정렬 실시 결과, $4 \rightarrow 1/6 \rightarrow 2/3 \rightarrow 5$ 의 순서로 총 4 개의 결과값이 도출됨 "끝"

문 제	4-5 절차지향 프로그래밍(Procedure Oriented Programming) 방법과 객체지향 프로그래밍(Object Oriented Programming) 방법을 비교 설명하시오.		
출 제 영 역	소프트웨어 공학	난 이 도	★★★★☆
출 제 배 경	소프트웨어 개발 방법론, 프로그래밍 관련 고전 문제로 이해 확인		
출 제 빈 도	유사문제 다수 출제		
참 고 자 료	- 절차지향과 객체지향에 관한 고찰, gbsb tistory 블로그 - 절차지향과 객체지향 언어의 차이, 장단점, 코드 비교, 용도 총정리, 큐 네이버블로그 - 새로 쓴 소프트웨어 공학 책		
Key word	절차, 프로시저, 프로세스, 함수, 순차적, 객체, 클래스, 추상화, 상속, 가상클래스, 다형성, 캡슐화, 정보은닉, 접근지정자, 오버로딩, 오버라이딩, SOLID 5 원칙		
풀 이	권태용(120 회 정보관리기술사)		

1. 절차지향 프로그래밍과 객체지향 프로그래밍 개요



- 1970 년대 이후 절차/기능 중심의 절차지향 프로그래밍과 1990 년대 이후 객체 중심 객체지향 프로그래밍 비교

2. 절차지향 프로그래밍과 객체지향 프로그래밍 개념 비교

구 분	절차지향 프로그래밍	객체지향 프로그래밍
개 념	- 절차와 프로세스를 중심으로 순차적 처리 기반의 프로그래밍 기법	- 객체(Object)를 중심으로 추상화 기반 객체 내 데이터로 처리 프로그래밍 기법
개 념 도	<p>절차적 방법</p> <p>프로그램 = 자료 구조 + 함수</p>	<p>객체 지향 방법</p> <p>객체 = 자료 구조 + 함수 프로그램 = 객체 + 객체</p>

3. 절차지향 프로그래밍과 객체지향 프로그래밍 특징 비교

구 분	절차지향 프로그래밍	객체지향 프로그래밍
접 근 방 식	하향식(Top Down)	상향식(Bottom Up)
모 둘 화	함수(Function) 단위	객체(Object) 단위
캡 술 화	데이터 프로시저 분리	데이터 프로시저 캡슐화
접 근 제 어	불가능	접근지정자(Public, Protected, Private) 가능
추 상 화	불가능	클래스 가능
다 형 성	불가능	오버로딩, 오버라이딩 가능
상 속	불가능	가능
가상클래스	미존재	상속 아래 존재
보 안 성	낮음	높음
데이터공유	모든 함수가 공유 가능	객체 간 멤버함수로만 공유
friend 함수	미존재	존재(C++)
사 용 언 어	C, Visual Basic, Fortran, Cobol, Pascal	Java, C++, C#, Object C, Python, Smalltalk
설 계 방 향	프로세스 중심 위주	데이터 중심 (데이터+연산)
설계난이도	쉬움(단순한 문법, 절차)	어려움(SOLID 원칙 기반 설계)
확 장 성	확장 어려움	확장성 용이
재 사 용 성	낮음	높음
유 지 보 수	어려움	쉬움(쉬운 디버깅)
중 복 코드	작성 가능성 높음	낮음
용 도	자원 한정, 바닥 설계가 필요한 프로젝트	큰 규모 협업이 많고 생산성 중시 프로젝트

4. 절차지향 프로그래밍과 객체지향 프로그래밍 예시 비교

구 분	절차지향 프로그래밍	객체지향 프로그래밍
예 시	<p>절차지향 분석</p> <pre> graph TD A[고객은 돈을 준비하여 돈을 자판기에 넣는다.] --> B[자판기에서 돈을 검사하여 준비된 제품과 잔돈을 꺼내준다.] B --> C[고객은 잔돈과 제품을 받아간다.] </pre>	<p>객체지향 분석</p> <pre> graph TD subgraph Customer [고객] C1[돈을 넣는다.] C2[돈을 받는다.] C3[제품을 받는다.] end subgraph VendingMachine [자판기] V1[돈을 받는다.] V2[돈을 검사한다.] V3[잔돈을 준다.] V4[제품을 준다.] end C1 --> V1 V2 --> C2 V3 --> C3 V4 --> C3 </pre>

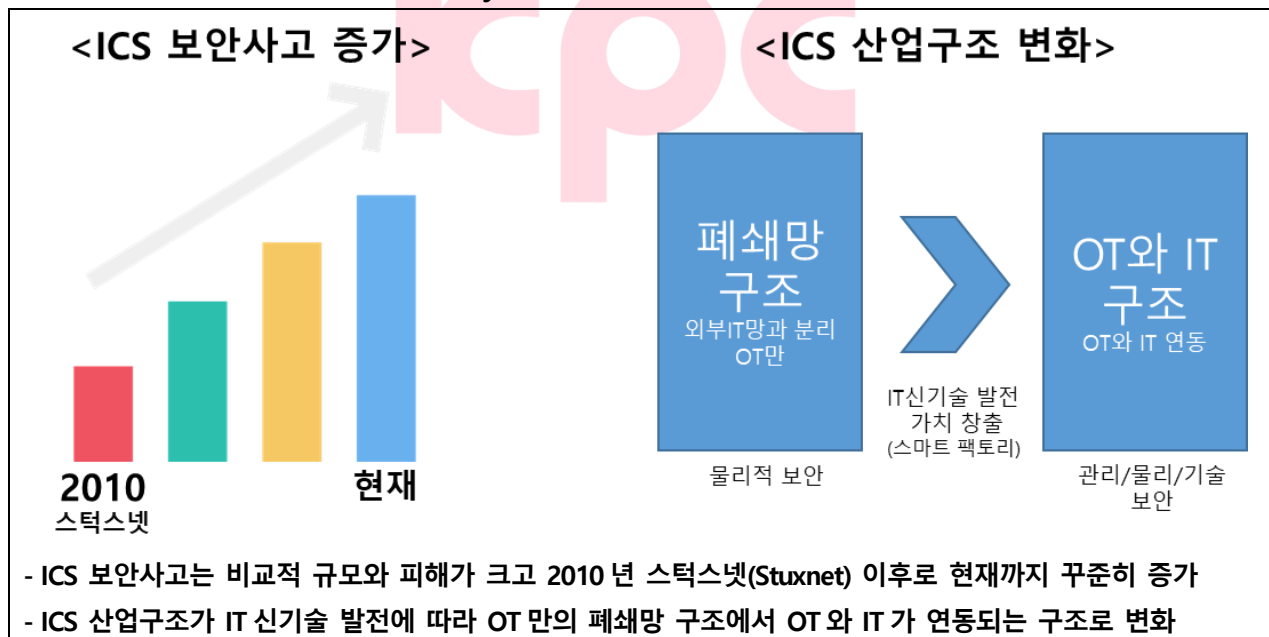
알 고 리 즈	<pre> graph TD Start([Start]) --> Decision1{고객이 돈을 넣었는가?} Decision1 -- No --> Decision1 Decision1 -- Yes --> Process1[돈_수불_및_계산()] Process1 --> Decision2{받은돈 >= 가격?} Decision2 -- No --> Print1[/print "가격 부족" 받은돈 반환/] Decision2 -- Yes --> Process2[제품_및_잔돈제공()] Process2 --> Process3[고객의돈 -= 가격; 자판기돈 += 가격; 고객제품++; 자판기제품--;] Process3 --> End([End]) </pre>	<pre> graph TD Start([Start]) --> Decision1{고객이 돈을 넣었는가?} Decision1 -- No --> Decision1 Decision1 -- Yes --> Call1[고객: 돈을_넣는다()] Call1 --> Call2[자판기: 돈을_받는다()] Call2 --> Call3[자판기: 돈을_검사한다() { 받은돈 >= 가격? }] Call3 -- No --> Print1[/print "가격 부족" 받은돈 반환/] Call3 -- Yes --> Call4[자판기: 잔돈을_준다()] Call4 --> Call5[자판기: 제품을_준다()] Call5 --> End([End]) </pre>
코 드 예 시	<pre> typedef struct Tetris_block { ... }; int score; Tetris_block generate_block(void); void move_block(Tetris_block); ... int main() { Tetris_block current_block; score = 0; current_block = generate_block(); move_block(current_block); } </pre>	<pre> class Game_engine { ... }; class Block_generator { ... }; class UI { ... }; int main() { Game_engine engine; Block_generator generator; UI ui; engine.init_newgame(); return 0; } </pre>

- 규모가 작은 개인 프로젝트에서 절차지향 프로그래밍을 사용하며 규모가 큰 프로젝트에서 객체지향 프로그래밍을 사용하여 적절한 방법 활용

“끝”

문 제	4-6 산업제어시스템(Industrial Control System)의 퍼듀(Purdue) 모델에 대하여 다음을 설명하시오. 가. 퍼듀 모델의 개념 나. 퍼듀 모델의 계층과 계층별 특징		
출 제 영 역	보안	난 이 도	★★★★★
출 제 배 경	산업제어시스템 업계에서 신기술 발전으로 가치창출을 위해 IT와 OT의 접점이 늘어 이에 대한 보안 참조 모델로 퍼듀 모델을 사용하고 있어 관련 이해도 확인		
출 제 빈 도	미출제		
참 고 자 료	<ul style="list-style-type: none"> - ICS/OT의 이해, 쿤텍 네이버블로그 - OT 보안을 위한 넓고 얇은 지식, 아이티데일리 기고문 - 발전제어시스템 보안모니터링 기술 동향, ITFIND - ICS & OT 산업 보안 이해하기, SecuLayer 		
Key word	IT 영역, OT 영역, IDMZ, 3 영역 6 레벨, 엔터프라이즈 보안, 인더스트리얼, PLC, HMI, VFD, PID, 액추에이터, 센서, 로봇, DCS, SCADA, 스텍스넷, ISA-99, PERA,		
풀 이	권태용(120 회 정보관리기술사)		

1. 산업제어시스템(Industrial Control System) 보안 모델의 필요



- 산업제어시스템(ICS)의 외부 IT 망과 OT 망을 연동하는 구조에 적합한 보안 모델로 퍼듀 모델 사용

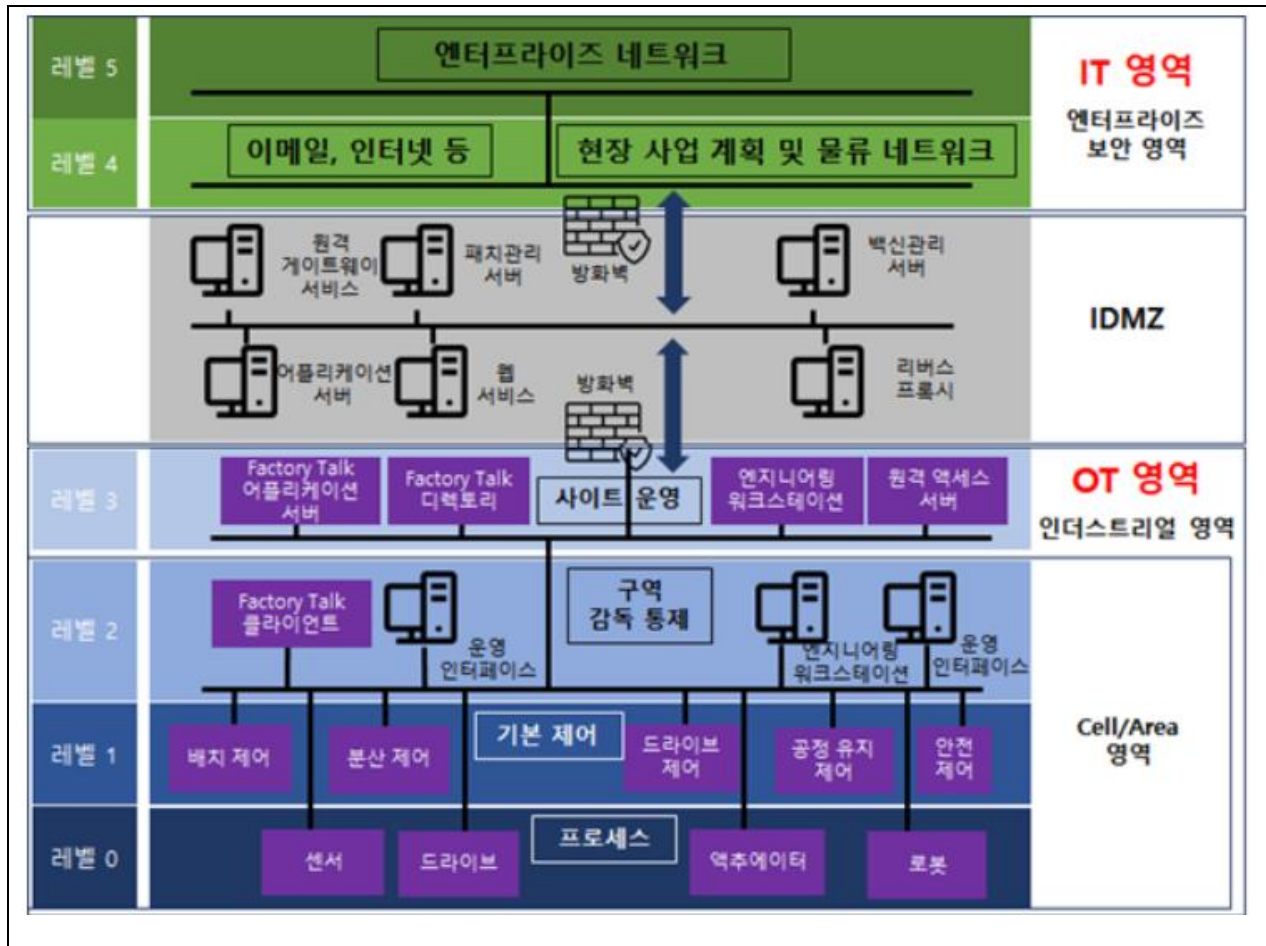
2. 산업제어시스템(ICS) 보안 참조 모델, 퍼듀 모델의 개념 및 계층

가. 퍼듀 모델의 개념

구 분	설 명
개 념	- 산업제어시스템(ICS) 시스템들을 견고하게 연결하기 위한 보안 관점의 네트워크 구조화 모델

주 체	- 2004 년 ISA-99 에서 제정한 PERA(Purdue Enterprise Reference Architecture)로부터 도입
구 성	- 크게 IT 영역과 OT 영역으로 나누고 IDMZ 영역을 가운데 위치해 3 가지 영역으로 구성 - 작게 세분화하여 0~5 의 6 가지 레벨로 구성
의 의	- ICS 네트워크 구조를 시각화하기 유용 - 학계 및 ICS 보안 업계에서 보안 참조 모델로 사용 - 산업제어시스템의 역할/기능에 따른 운영기술/보안고려사항 설정 가능

나. 퍼듀 모델의 계층



- 레벨 4, 5 의 엔터프라이즈 영역은 IT, 레벨 0~3 영역은 OT 로 구분되며 IT와 OT 사이에 레벨 3.5 영역 IDMZ 존재
- 퍼듀 모델 계층은 레벨과 영역을 기준으로 나눌 수 있음

3. 퍼듀(Purdue) 모델의 계층별 특징

가. 퍼듀 모델의 영역 계층별 특징

영역	특징
IT 영역	<ul style="list-style-type: none"> - 엔터프라이즈 보안 영역 - ERP 및 SAP 와 같은 비즈니스 시스템이 일반적으로 사용되는 ICS 영역 - 일정관리, 공급망관리 수행 - Level 4~5 포함
IDMZ 영역	<ul style="list-style-type: none"> - 인더스트리얼 비무장 영역 - IT 영역과 OT 영역 사이에서 직접 통신 차단

	<ul style="list-style-type: none"> - 기존 IT DMZ 존과 흡사하며 방화벽 위치 - Level 3.5
OT 영역	<ul style="list-style-type: none"> - 인더스트리얼 영역 - 실제 프로세스가 가동되는 중요한 영역으로 SCADA, DCS 시스템도 위치 - Level 0~3 포함

나. 퍼듀 모델의 레벨 계층별 특징

영역	레벨	특징
IT 영역 (엔터프라이즈 보안 영역)	Level 5	<ul style="list-style-type: none"> - 엔터프라이즈 네트워크 - ICS 네트워크 연결 기반 비즈니스 결정 필요 데이터 제공 영역 - 개별 공장의 서브시스템 데이터를 받아와 누적된 데이터를 기반으로 전반적인 공장 설비의 생산 상태, 재고, 수요를 리포트
	Level 4	<ul style="list-style-type: none"> - 사이트 비즈니스 및 물류 - 모든 생산 프로세스들을 지원하는 IT 시스템 위치 - 데이터베이스 서버, 어플리케이션 서버(웹, 보고서, MES), 파일 서버, 이메일 클라이언트, 감독자 데스크톱 - 가동 시간, 설비에서 생산된 개체 수와 같은 생산 정보 통계를 리포팅 - 주문이나 비즈니스 데이터를 가져와서 ICS/OT 시스템에 분배
IDMZ (인더스트리얼 비무장 영역)	Level 3.5	<ul style="list-style-type: none"> - Industrial DMZ(비무장) 영역, 분리/검사 계층 - IT와 OT 영역 사이에서 직접적 통신을 막고 브로커 서비스를 통해 통신 중계 - NIST 사이버 보안 프레임워크 및 NERC CIP와 같은 보안 표준 노력 결과 - 하위 계층 시스템 공격 직접 노출 방지 (방화벽) - IDMZ 일부 시스템 손상 시, 서비스 종료로 공정 가용성 보장 - 프록시 서버, 데이터베이스 복제 서버(패치관리, 백신관리 등), 마이크로소프트 도메인 컨트롤러 위치
OT 영역 (인더스트리얼 영역)	Level 3	<ul style="list-style-type: none"> - 사이트 운영 - 시설 전체 제어 기능, 모니터링 담당 - 시설을 동작시키는 모든 시스템에 대한 개요를 제공하는 HMI 및 운영장비가 모여 있는 중앙 집중적 제어실 역할 - 데이터베이스, 어플리케이션 서버(웹, 보고), 파일 서버, 마이크로소프트 도메인 컨트롤러, HMI 서버 엔지니어링 워크스테이션 위치
	Level 2	<ul style="list-style-type: none"> - 영역 감독 통제 - 레벨 3 보다 구체적인, 시스템의 특정부분을 HMI 시스템으로 모니터링하고 관리 - HMI, 라인 컨트롤 PLC 등 감독 통제 시스템, 엔지니어링 워크스테이션 위치 - Cell/Area 영역에 포함
	Level 1	<ul style="list-style-type: none"> - 기본 통제 영역 - 모든 제어 장비가 모여 있음(배치, 분산, 드라이브, 공정유지, 안전 제어) - PLC, VFD, PID 컨트롤러 위치 - Cell/Area 영역에 포함
	Level 0	<ul style="list-style-type: none"> - 프로세스 - 실제 프로세스 수행 장비 집합, 무중단 원칙

		<ul style="list-style-type: none"> - 센서, 액추에이터, 펌프, 밸브, 로봇 위치 - Cell/Area 영역에 포함
--	--	--

- 레벨 3 이하 OT 영역에서 사용되는 프로토콜은 Profibus, DeviceNet, ModBus, CAN, CIP 등이 존재하며 IDMZ 영역에서 이 프로토콜을 차단하여 OT 영역을 벗어나지 못하게 통제함

"끝"

