

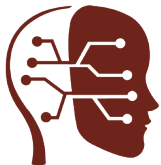
# Software Engineering

## Lecture 05: 계획 (프로젝트 관리와 계획) (Part 2)

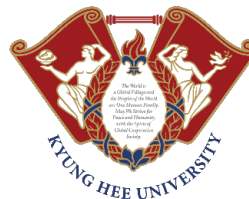
**Professor: Jung Uk Kim**

[ju.kim@khu.ac.kr](mailto:ju.kim@khu.ac.kr)

Computer Science and Engineering, Kyung Hee University



**Visual AI Lab.**



# 일정 계획

## • 일정 계획

- 소프트웨어를 개발하기 위해 어떤 작업이 필요한지 찾은 후, **진행할 순서를 결정**하거나, **주어진 개발 기간에 소작업의 개발 기간** 및 그들 간의 **순서**, **필요한 자원** 등과 같은 일정을 계획하는 것

작업일정 작업 단계	이정표												산출물
	1	2	3	4	5	6	7	8	9	10	11	12	
계획	■												시스템 계획서 프로젝트 정의서
분석		■											요구 분석 명세서
기본 설계			■										기본 설계서
상세 설계				■	■	■	■						상세 설계서
사용자 지침서			■	■	■								사용자 지침서
시험 계획					■	■	■	■					시험 계획서
구현							■	■	■	■			원시 코드
통합 테스트										■			시스템 통합 계획서
시스템 테스트											■		시스템
인수 테스트												■	개발 완료 보고서

# 일정 계획

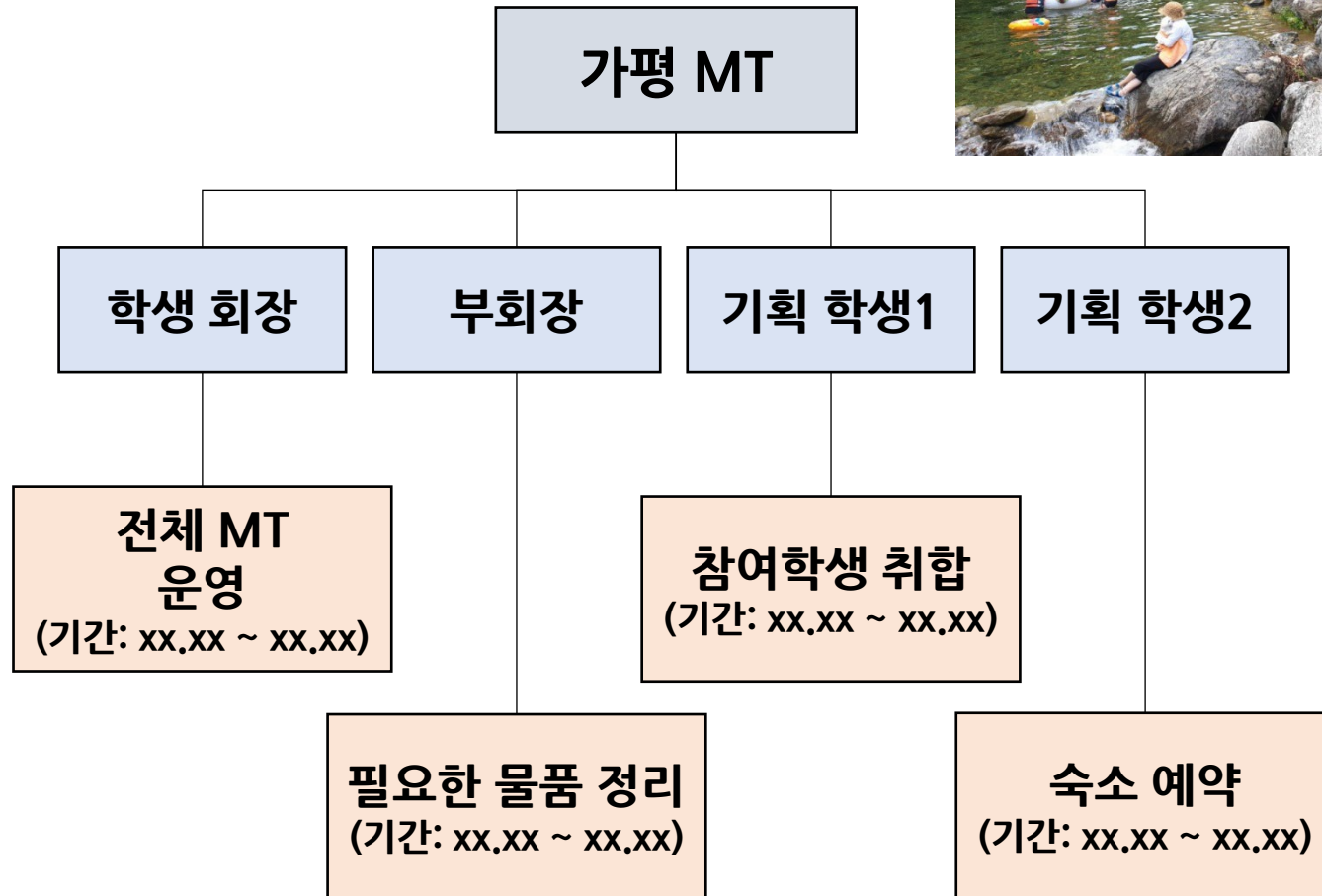
## • 일정 계획

- 작업 순서 결정, 소작업의 개발 기간, 순서, 필요한 자원 등의 일정 계획

총 개발 기간	6개월(0000. 1. 1. ~ 0000. 6. 31.)															
소작업	교과 과정 관리, 수업 관리, 수강 관리, 성적 관리, 사용자 정보 관리															
소작업별 개발 기간	<table><tr><td>소작업</td><td>개발 기간</td></tr><tr><td>사용자 정보 관리</td><td>1개월(1. 1. ~ 1. 31.)</td></tr><tr><td>교과 과정 관리</td><td>1개월(2. 1. ~ 2. 28.)</td></tr><tr><td>수업 관리</td><td>1개월(3. 1. ~ 3. 31.)</td></tr><tr><td>수강 관리</td><td>1개월(4. 1. ~ 4. 30.)</td></tr><tr><td>성적 관리</td><td>1개월(5. 1. ~ 5. 31.)</td></tr><tr><td>테스트</td><td>1개월(6. 1. ~ 6. 30.)</td></tr></table>		소작업	개발 기간	사용자 정보 관리	1개월(1. 1. ~ 1. 31.)	교과 과정 관리	1개월(2. 1. ~ 2. 28.)	수업 관리	1개월(3. 1. ~ 3. 31.)	수강 관리	1개월(4. 1. ~ 4. 30.)	성적 관리	1개월(5. 1. ~ 5. 31.)	테스트	1개월(6. 1. ~ 6. 30.)
소작업	개발 기간															
사용자 정보 관리	1개월(1. 1. ~ 1. 31.)															
교과 과정 관리	1개월(2. 1. ~ 2. 28.)															
수업 관리	1개월(3. 1. ~ 3. 31.)															
수강 관리	1개월(4. 1. ~ 4. 30.)															
성적 관리	1개월(5. 1. ~ 5. 31.)															
테스트	1개월(6. 1. ~ 6. 30.)															
개발 순서	사용자 정보 관리 → 교과 과정 관리 → 수업 관리 → 수강 관리 → 성적 관리															
필요 자원	개발 툴(파워빌더), 개발용 PC, 개발 공간															

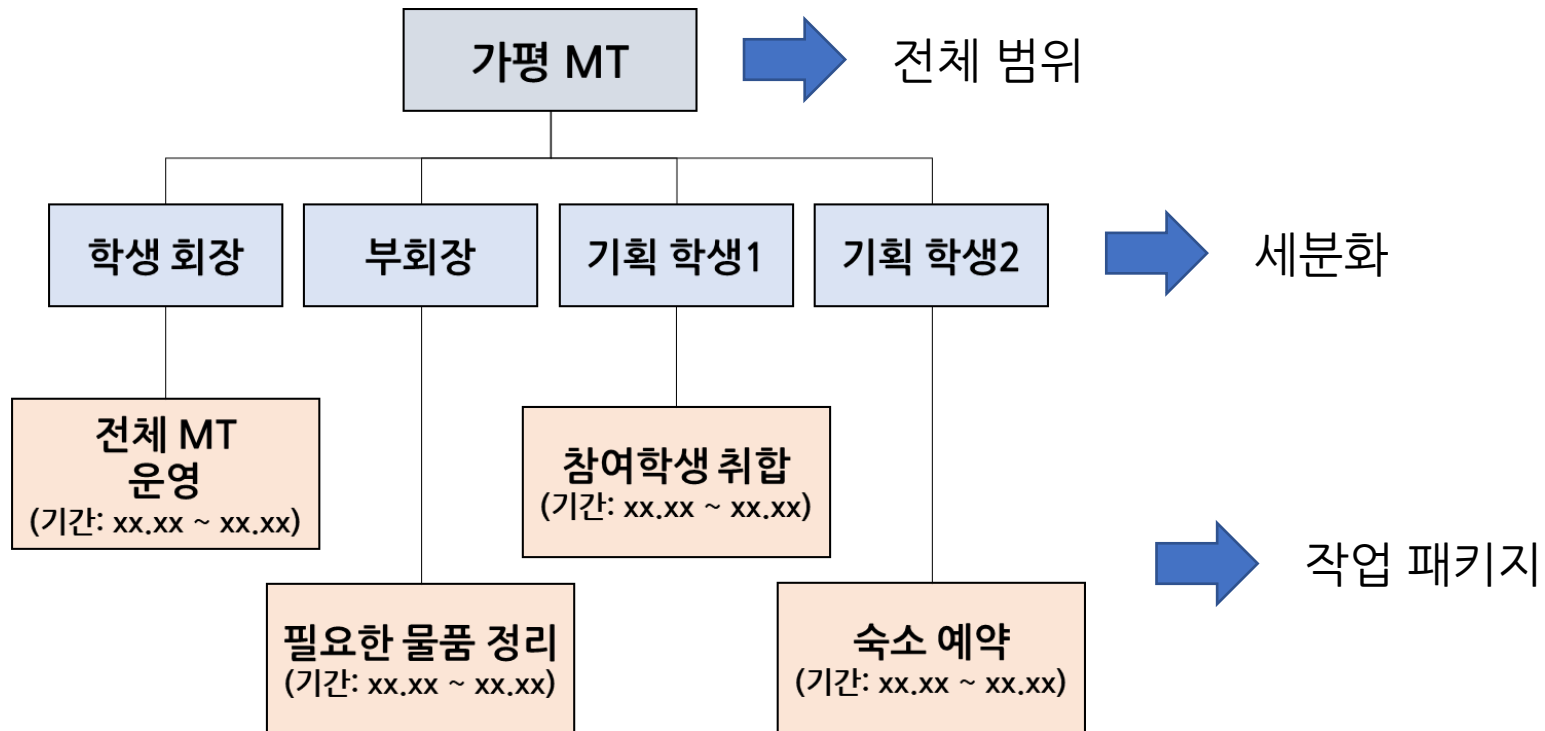
# 일정 계획 - 작업 분할 구조도 (WBS)

- 일정 계획 (실생활 예제)



# 일정 계획 - 작업 분할 구조도 (WBS)

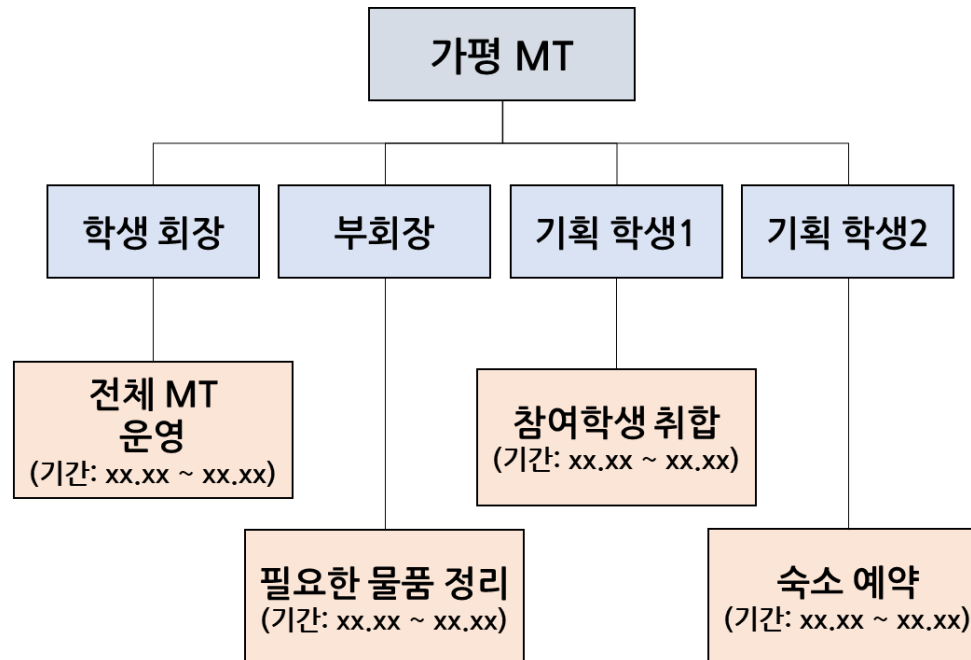
- 작업 분할 구조도 (Work Breakdown Structure, WBS)
  - 프로젝트 목표를 달성하기 위해 **필요한 활동과 업무를 세분화**하는 작업
  - 프로젝트 **구성 요소들을 계층 구조로 분류** (프로젝트의 작업 파악이 가능)
  - 프로젝트의 **전체 범위 정의**
  - 프로젝트 **작업을 세분화** (최하위에 있는 항목: **작업 패키지**)



# 일정 계획 - 작업 분할 구조도 (WBS)

## • 작업 분할 구조도 (장점)

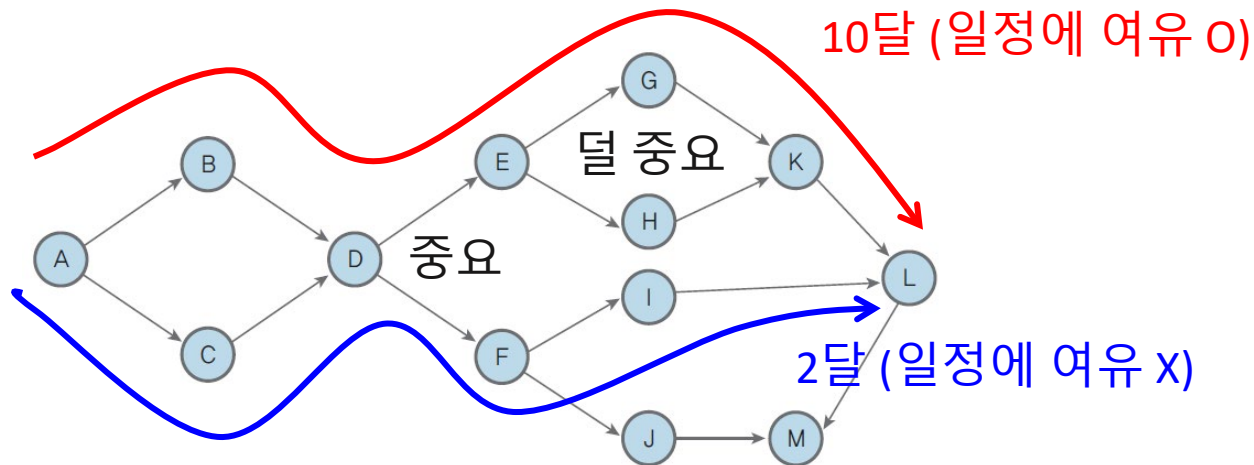
- 사용자와 개발자 간의 **의사소통 도구로 사용함**
- 프로젝트 업무 내역을 가시화할 수 있어 **관리가 용이함**
- 프로젝트 팀원의 **책임과 역할이 분명함**
- 필요 인력, 일정 계획, 개발비 산정 시 **기초로 활용함**
- **성과 측정 및 조정 시 기준선으로 활용**할 수 있음



# 일정 계획 - 네트워크 차트 (PERT/CPM)

## • 네트워크 차트

- WBS의 작업 순서, 소요 기간 등을 **네트워크 형태의 그래프로 표현**
- 이를 통해 **어떤 작업이 중요한지, 또 일정에 여유가 있는 작업은 어떤 것인지** 찾아내 **중점 관리**를 해야 하는 작업을 명확히 하는데 사용



# 일정 계획 - 네트워크 차트 (PERT/CPM)

- 네트워크 차트 (예시) - 학사 관리 어플리케이션
  - 네트워크 차트를 그리기 위한 목록

작업	작업 설명	선행 작업	소요 기간(주)
A	개인 정보 등록/수정/조회/삭제 프로그램 개발	-	2
B	학적 변동 자료 등록/수정/조회/삭제 프로그램 개발	A	6
C	휴 · 복학 및 자퇴 등록/수정/조회/삭제 프로그램 개발	A	4
D	교육 과정 등록/수정/조회/삭제 프로그램 개발	B, C	2
E	유사/동일 과목 등록/수정/조회/삭제 프로그램 개발	D	4
F	개설 강좌 등록/수정/조회/삭제 프로그램 개발	D	3
G	수강 과목 등록/수정/조회/삭제 프로그램 개발	E	2
H	시간표 등록/수정/조회/삭제 프로그램 개발	E	4
I	성적 등록/수정/조회/삭제 프로그램 개발	F	2
J	장학생 등록/수정/조회/삭제 프로그램 개발	F	1
K	등록금 등록/수정/조회/삭제 프로그램 개발	G, H	2
L	졸업 사정 등록/수정/조회/삭제 프로그램 개발	I, K	2
M	사회봉사 실적 등록/수정/조회/삭제 프로그램 개발	J, L	3

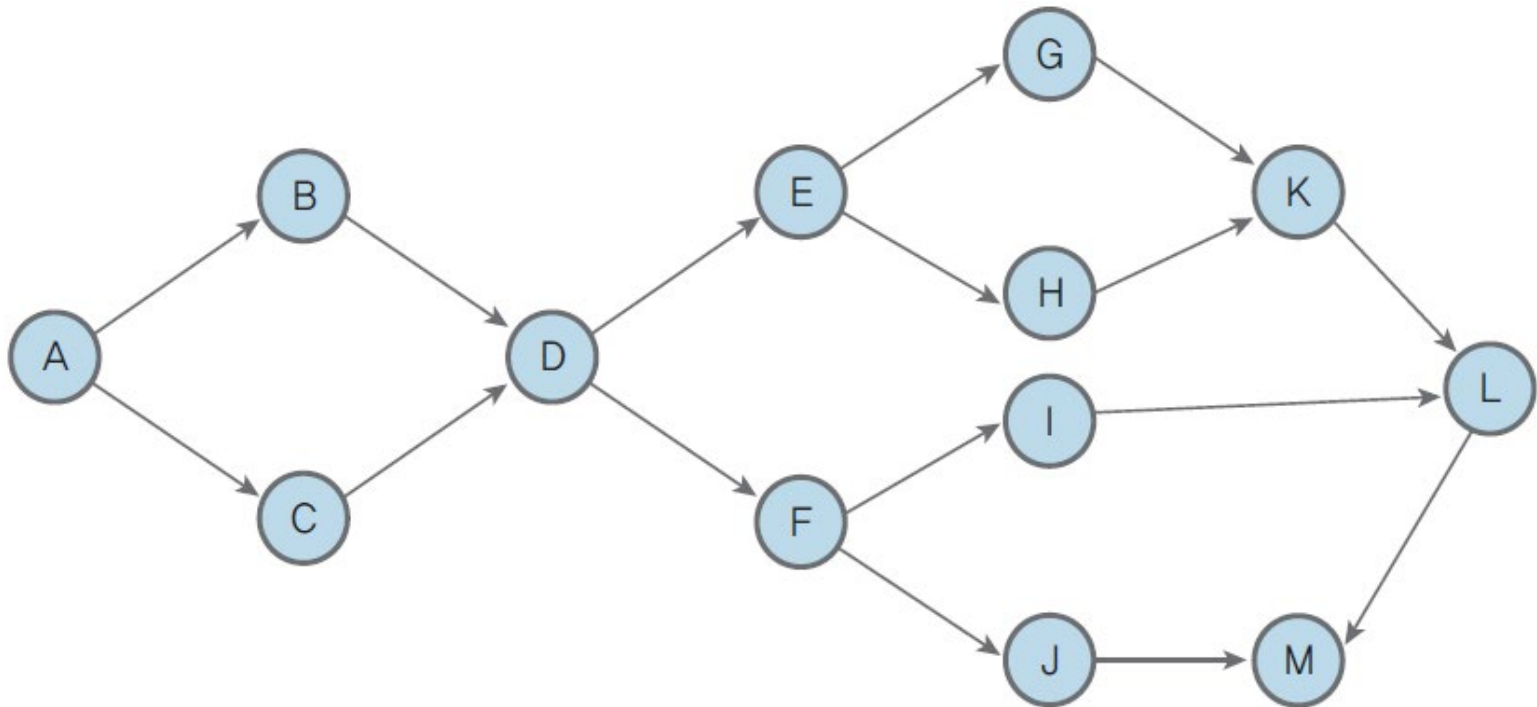


# 일정 계획 - 네트워크 차트 (PERT/CPM)

- 네트워크 차트 (예시) - 학사 관리 어플리케이션

- **(1) CPM (Critical Path Model)을 그린다**

- 노드(Node)는 작업, 간선은 노드 간의 선후 의존 관계 (e.g., D는 B,C가 끝나야 시작)

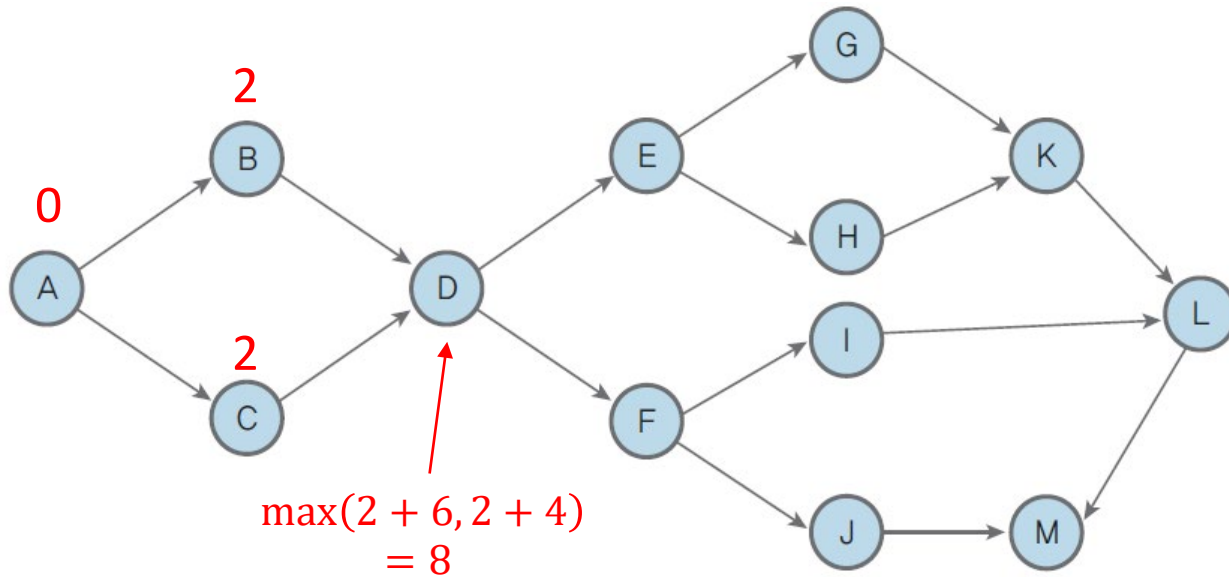


# 일정 계획 - 네트워크 차트 (PERT/CPM)

- 네트워크 차트 (예시) - 학사 관리 어플리케이션

- (2) ES (Earliest Start Time)을 구한다

- ES: 가능한 빨리 시작할 수 있는 시간으로, 선행 작업이 완료되었을 때 해당작업을 시작할 수 있는 가장 빠른 시점



작업	소요 기간(주)
A	2
B	6
C	4
D	2
E	4
F	3
G	2
H	4
I	2
J	1
K	2
L	2
M	3

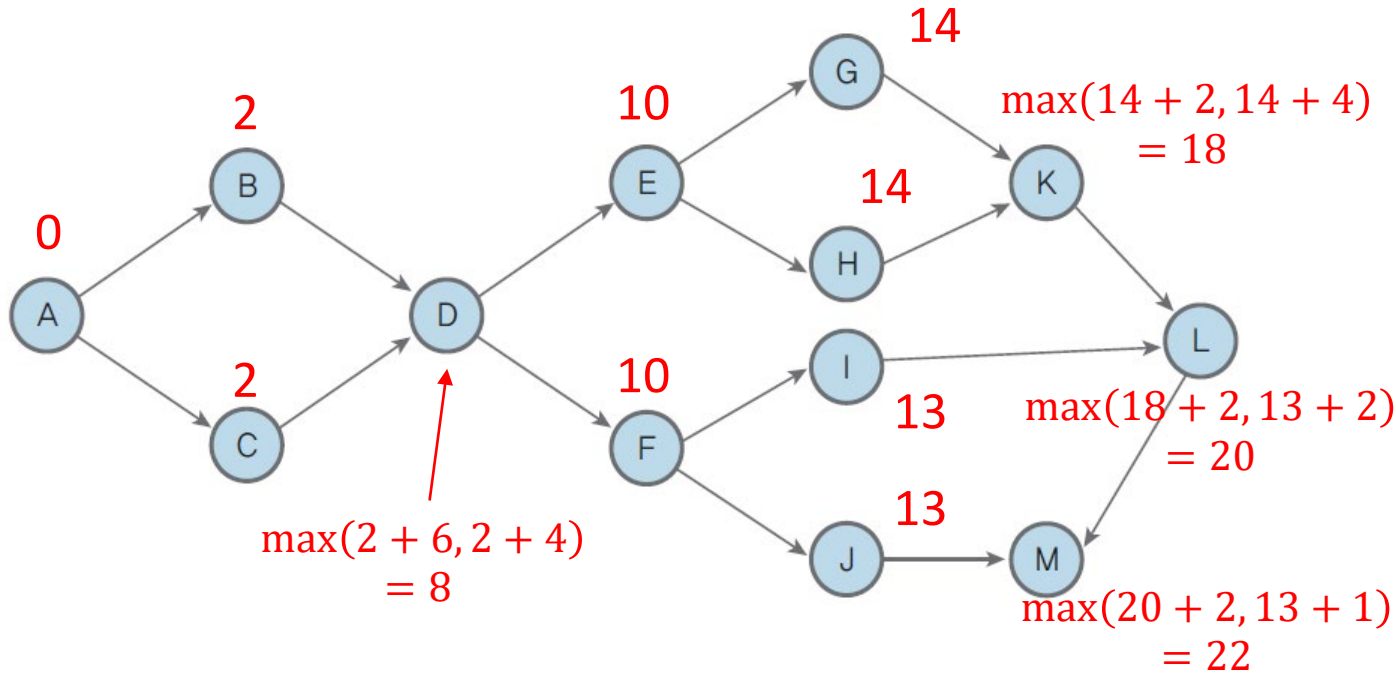
작업	A	B	C	D	E	F	G	H	I	J	K	L	M
작업 시작 시간	0	2	2	8	10	10	14	14	13	13	18	20	22
작업 시간	2	6	4	2	4	3	2	4	2	1	2	2	3

# 일정 계획 - 네트워크 차트 (PERT/CPM)

- 네트워크 차트 (예시) - 학사 관리 어플리케이션

- (2) ES (Earliest Start Time)을 구한다

- ES: 가능한 빨리 시작할 수 있는 시간으로, 선행 작업이 완료되었을 때 해당작업을 시작할 수 있는 가장 빠른 시점



작업	소요 기간(주)
A	2
B	6
C	4
D	2
E	4
F	3
G	2
H	4
I	2
J	1
K	2
L	2
M	3

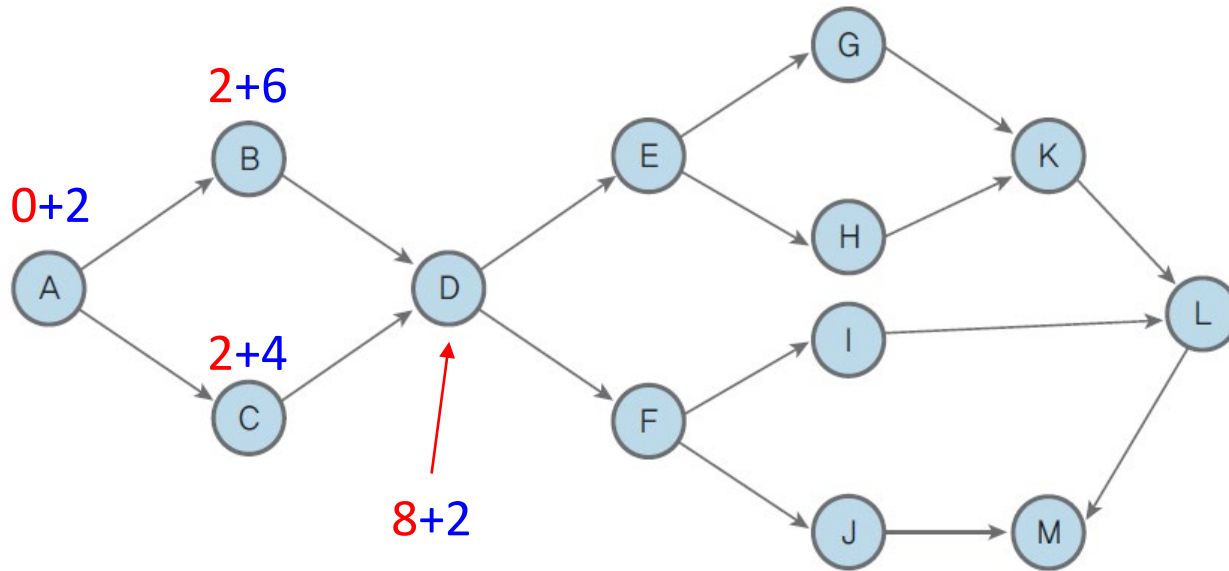
작업	A	B	C	D	E	F	G	H	I	J	K	L	M
작업 시작 시간	0	2	2	8	10	10	14	14	13	13	18	20	22
작업 시간	2	6	4	2	4	3	2	4	2	1	2	2	3

# 일정 계획 - 네트워크 차트 (PERT/CPM)

## • 네트워크 차트 (예시) - 학사 관리 어플리케이션

### • (2) EF (Earliest Finish Time)을 구한다

- EF: 가장 빠른 시작 시간(ES)으로 시작했을 때의 가장 빠른 완료 시간
- 가능한 빨리 끝낼 수 있는 시간으로 'ES+작업 소요 시간'



작업	소요 기간(주)
A	2
B	6
C	4
D	2
E	4
F	3
G	2
H	4
I	2
J	1
K	2
L	2
M	3

작업	A	B	C	D	E	F	G	H	I	J	K	L	M
작업 시작 시간	0	2	2	8	10	10	14	14	13	13	18	20	22
작업	A	B	C	D	E	F	G	H	I	J	K	L	M
EF	2	8	6	10	14	13	16	18	15	14	20	22	25

ES

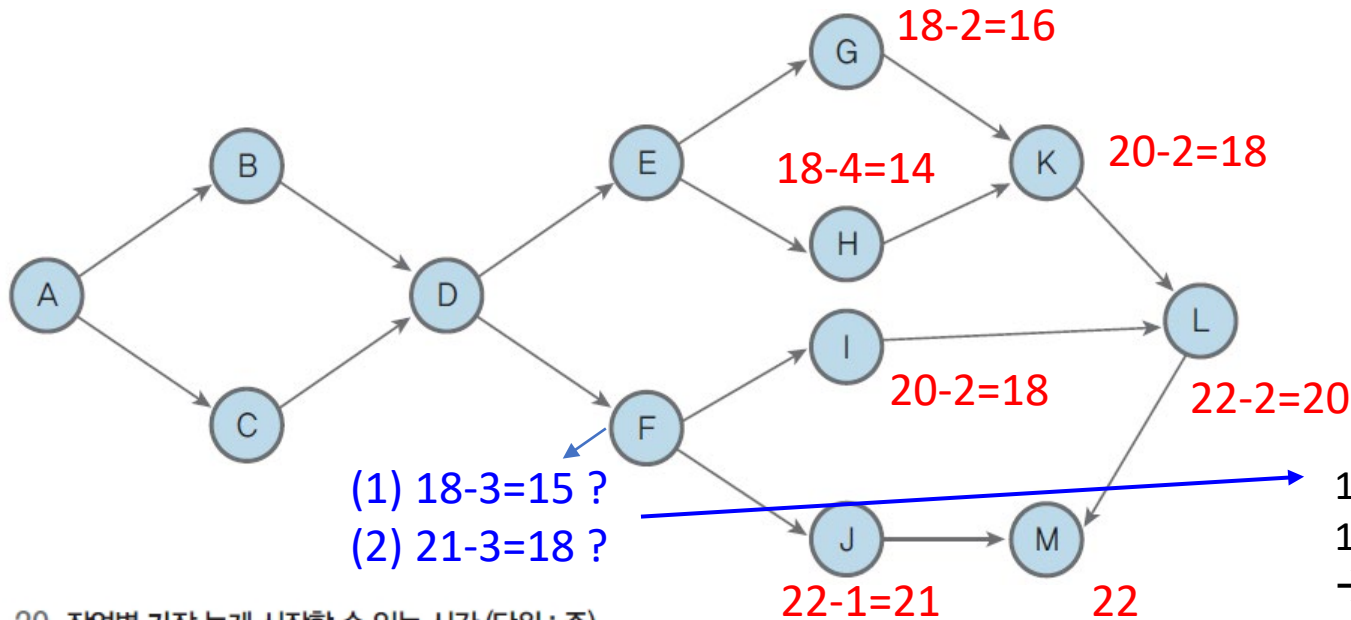
EF

# 일정 계획 - 네트워크 차트 (PERT/CPM)

## • 네트워크 차트 (예시) - 학사 관리 어플리케이션

### • (3) LS (Latest Start Time)을 구한다

- LS: 늦어도 시작해야 하는 시간
- 이 시간에 시작하지 않으면(늦으면) 총 일정이 지연됨(뒤에서 앞 방향으로 계산)



18에 맞추면 I의 시작시간  
18에 영향을 미침  
→ 15에 맞춰야 함  
 $\min(18-3, 21-3)$   
 $= 15$

표 3-20 작업별 가장 늦게 시작할 수 있는 시간(단위: 주)

작업	A	B	C	D	E	F	G	H	I	J	K	L	M
작업 시작 시간	0	2	4	8	10	15	16	14	18	21	18	20	22
작업 시간	2	6	4	2	4	3	2	4	2	1	2	2	3

ES

# 일정 계획 - 네트워크 차트 (PERT/CPM)

- 네트워크 차트 (예시) - 학사 관리 어플리케이션

- (3) LS (Latest Start Time)을 구한다

- LS: 늦어도 시작해야 하는 시간
    - 이 시간에 시작하지 않으면(늦으면) 총 일정이 지연됨(뒤에서 앞 방향으로 계산)

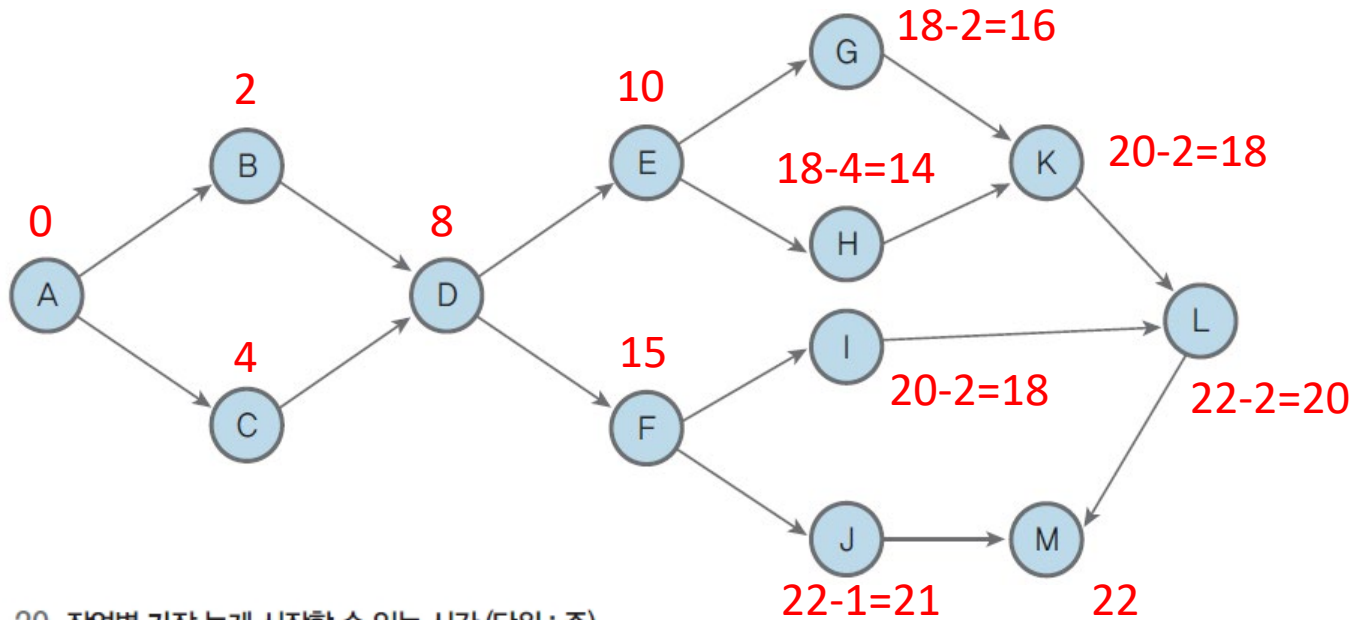


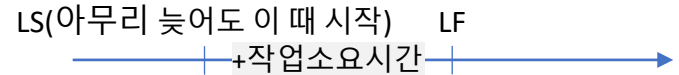
표 3-20 작업별 가장 늦게 시작할 수 있는 시간 (단위: 주)

작업	A	B	C	D	E	F	G	H	I	J	K	L	M
작업 시작 시간	0	2	4	8	10	15	16	14	18	21	18	20	22
작업 시간	2	6	4	2	4	3	2	4	2	1	2	2	3

# 일정 계획 - 네트워크 차트 (PERT/CPM)

- 네트워크 차트 (예시) - 학사 관리 어플리케이션

- (4) LF (Latest Finish Time)을 구한다



- LF: 작업을 가장 늦게 끝낼 수 있는 시간 (LS + 작업소요시간)

작업	A	B	C	D	E	F	G	H	I	J	K	L	M	
작업 시작 시간	0	2	4	8	10	15	16	14	18	21	18	20	22	LS
작업 시간	2	6	4	2	4	3	2	4	2	1	2	2	3	
작업 완료 시간	2	8	8	10	14	18	18	18	20	22	20	22	25	LF

- (5) ST (Slack Time)을 구한다

- ST: 여유 있는 시간
    - 가장 늦게 시작하는 시간 (LS) - 가장 빨리 시작하는 시간 (ES)

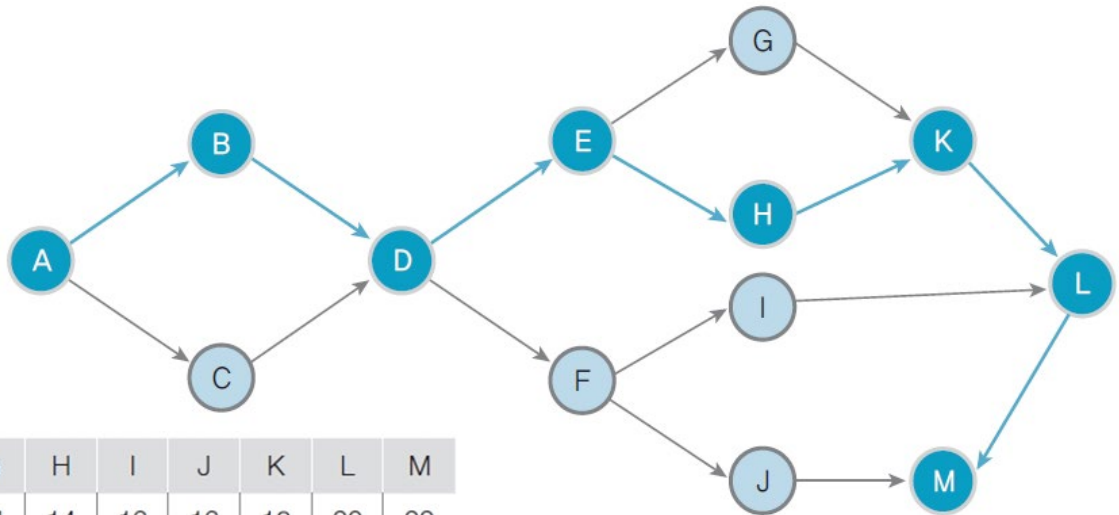
작업	A	B	C	D	E	F	G	H	I	J	K	L	M	
작업별 빠른 시작 시간	0	2	2	8	10	10	14	14	13	13	18	20	22	LS
작업별 늦은 시작 시간	0	2	4	8	10	15	16	14	18	21	18	20	22	ES
여유 시간	0	0	2	0	0	5	2	0	5	8	0	0	0	

# 일정 계획 - 네트워크 차트 (PERT/CPM)

- 네트워크 차트 (예시) - 학사 관리 어플리케이션

- (6) 임계 경로(Critical Path)를 구한다**

- 임계 경로: 그래프에 여유시간이 없는 경로
- 모든 일정 계획은 임계 경로에 의해 좌우 됨
- 임계 경로를 벗어난 작업은 시간 한도 내에서는 여유가 있음  
(e.g., 작업 I가 2주 지연되더라도 5주의 여유시간이 있어 프로젝트 최종 완료일에 영향 X)



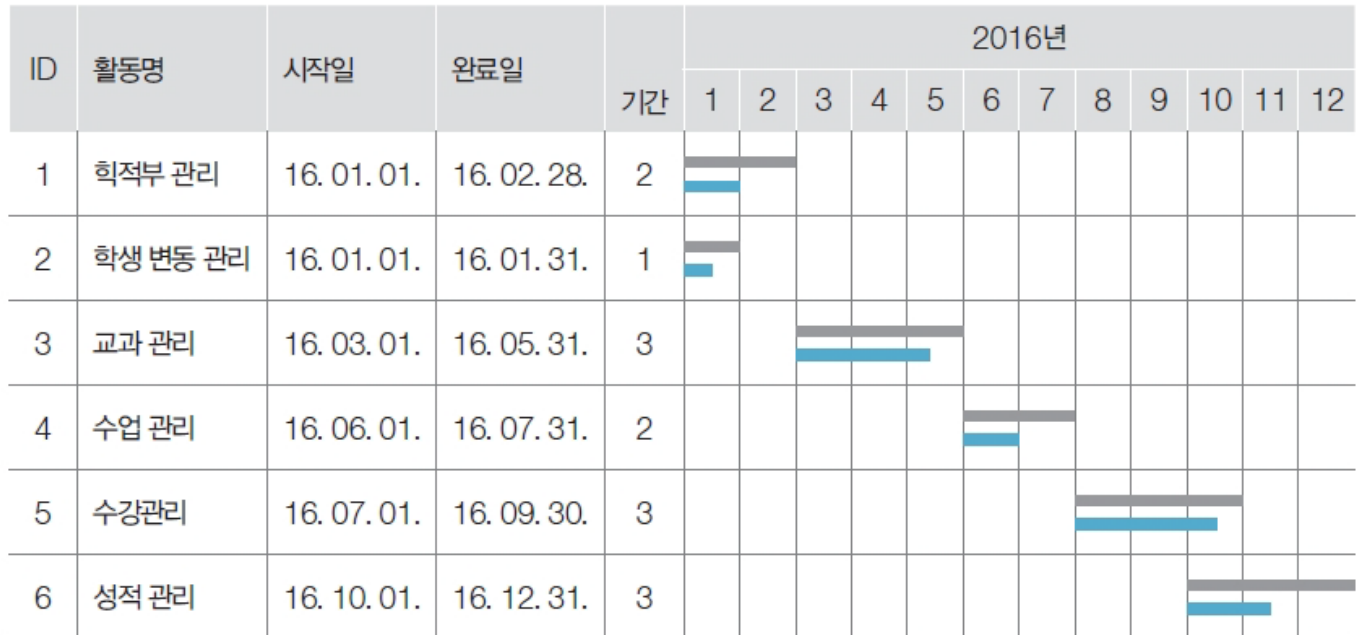
작업	A	B	C	D	E	F	G	H	I	J	K	L	M
작업별 빠른 시작 시간	0	2	2	8	10	10	14	14	13	13	18	20	22
작업별 늦은 시작 시간	0	2	4	8	10	15	16	14	18	21	18	20	22
여유 시간	0	0	2	0	0	5	2	0	5	8	0	0	0



# 일정 계획 - 간트 차트

## • 간트 차트

- 프로젝트 일정 관리를 위한 바(bar) 형태의 도구
- WBS를 통해 프로젝트의 주요 활동을 파악한 후, 각 활동의 일정을 시작하는 지점과 끝나는 시점을 연결한 막대 모양으로 표시
- 전체 일정을 한눈에 볼 수 있음

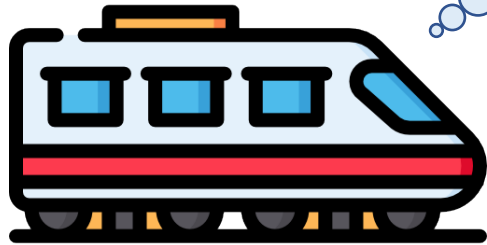


■ 계획된 작업량

■ 현재까지의 작업량

# 위험 분석

- 위험 분석 (예시)



부산!

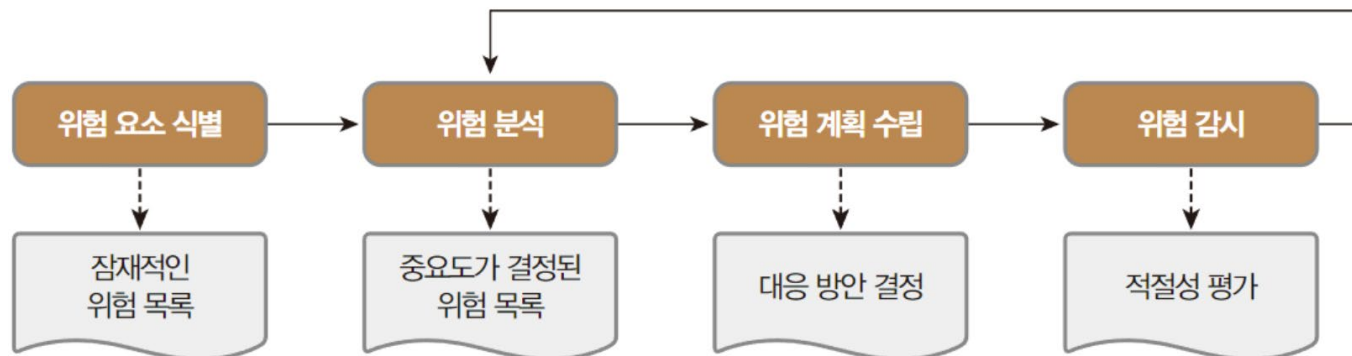


- 부산 여행에서 발생할 수 있는 위험 요소
  - 목표 변경 (부산 → 여수)
  - 기차의 사고 / 회사 파업
  - 소매치기로 인한 여행 경비 0원
  - 친구들과 다툼으로 인한 여행 인원 부족

# 위험 분석

- 위험 분석

- 소프트웨어 개발에 방해가 되는 요소를 미리 예상하고, 이에 관해 적절한 대책을 수립하는 활동
- 위험 요소: 사용자 요구사항 (가장 큰 위험 요소), 인력 부족, 예산 부족 등
- 위험 분석 단계
  - (1) 위험 요소를 미리 파악하고(**위험 요소 식별**),
  - (2) 위험 요소의 발생 확률과 영향도를 평가한 뒤(**위험 분석**),
  - (3) 분석한 결과에 따라 위험 우선순위를 정해 그에 맞게 대책을 세워야 함(**위험 계획 수립**)



# 위험 분석 - 위험 요소 식별

- 위험 요소 식별

- 프로젝트 수행에 영향을 주는 위험 요소 파악
- 브레인스토밍 또는 유사 프로젝트 때의 위험 요소 참조

위험 요소	위험 내용
개발자의 이직	프로젝트 수행 중 개발자들이 이직한다.
요구사항 변경	요구사항 확정 이후에도 변경 요구가 계속된다.
발주사의 재정적 어려움	프로젝트 수행 중 고객사에 경제적인 어려움이 발생한다.
예상을 빚나간 투입 인력	처음에 예측한 인력보다 더 많은 인력이 필요하다.
개발 기간 부족	처음에 예측한 개발 기간을 초과한다.
개발비 초과	처음에 예측한 개발비로 개발을 완료할 수 없다.

# 위험 분석 - 위험 분석, 위험 계획 수립

- 위험 분석
  - 위험 요소 식별 후, **위험 요소가 발생할 가능성과 영향력을 판단**
  - 과거 프로젝트에서 위험을 분석한 **경험이 많은 개발자에 의존**
  - **위험 발생 가능성, 위험 발생 확률** 등 분석
- 위험 계획 수립
  - 위험 분석을 통해 이를 처리하는 위험 대응 방안 수립
  - 우선순위에 따라 진행
- (추가) 위험 감시
  - 식별된 위험은 계속 감시
  - 유사한 프로젝트를 진행할 때 참고할 수 있도록 데이터베이스 내에 기록

---

*Questions?*