

## 소웨어 개념

### ❖ 1.1 소프트웨어

- 프로그램 + 프로그램의 개발, 운용, 보수에 필요한 정보 일체

### ❖ 개념적, 무형적

### ❖ 추가적 소프트웨어의 특징

- 극히 적은 비용으로 복제 가능
- 소프트웨어의 비마모성(오래쓴다고 닳지않음)
- 노동집약성
- 고치기 힘들
- 품질 높이기 쉽지않음
- 잘 훈련받지않으면 제작 어려움
- 코드가 누적되면 오류율 높아짐

### ❖ 소프트웨어의 종류

- 주문형 소프트웨어
  - 특정 고객의 수요를 만족시키기 위해 개발된 소프트웨어
  - 장점: 특정 고객 수요 만족 가능
  - 단점: 사용자가 한정되고 비용이 많이 듦
- 패키지형 소프트웨어
  - 공개된 시장에서 판매되는 소프트웨어, COTS라고도 불림
  - 장점: 저렴하고 신뢰도 높음(다수의 베타테스터)
  - 단점: 특정기관의 요구에 최적화되지 않음

- 임베디드 소프트웨어
  - 장점: 개발 비용이 비교적 저렴
  - 단점: 하드웨어 교체하지 않은 이상 소프트웨어 교체 어려움

## 소프트웨어 공학의 정의

### ❖ 소프트웨어 공학

- 고객의 문제를 해결해주기 위해 대규모의 품질좋은 소프트웨어 시스템을 정해진 시간과 비용으로 개발하거나 발전시키는 체계적인 프로세스
- 소프트웨어 개발에 사용되는 방법이 일회성이 아닌 반복 사용 가능

### ❖ 소프트웨어 공학의 제약사항

- 비용, 시간, 기타 제약
  - 한정된 자원(시간, h/w, 인력, 돈)
  - 얻는 이득이 비용을 초과해야 함
  - 더 빠르고 싸게 개발하도록 다른 기업과 경쟁
  - 비용과 시간의 부정확한 예측으로 대부분의 프로젝트가 실패

### ❖ 소프트웨어 공학의 목표

- 복잡도 낮춤
- 비용 최소화
- 개발기간 단축
- 대규모 프로젝트 관리
- 고품질 소프트웨어
- 효율성 증가

### ❖ 소프트웨어 공학의 주제

➤ 단계적 프로세스

- 소프트웨어에 대한 비전과 개념을 파악하고 만족하는 소프트웨어로 구현될 때까지 정해진 순서 반복(요구사항 명세, 설계, 구현, 테스트 등 단계적 절차)

➤ 품질 보증

- 소프트웨어의 품질 보증(Software Quality Assurance)은 개발작업의 적절한 수행 여부를 확인하는 작업

➤ 프로젝트 관리

- 개발과 품질보증 작업을 관리하고 감독하는 일(예측, 프로젝트 계획, 일정, 리스트 관리, 행정 등)