



# 3장. SW 비용산정



# Contents

1. 기능 점수 개요
2. 기능 점수 산정 절차
3. 정규법과 간이법
4. 기능 점수 활용

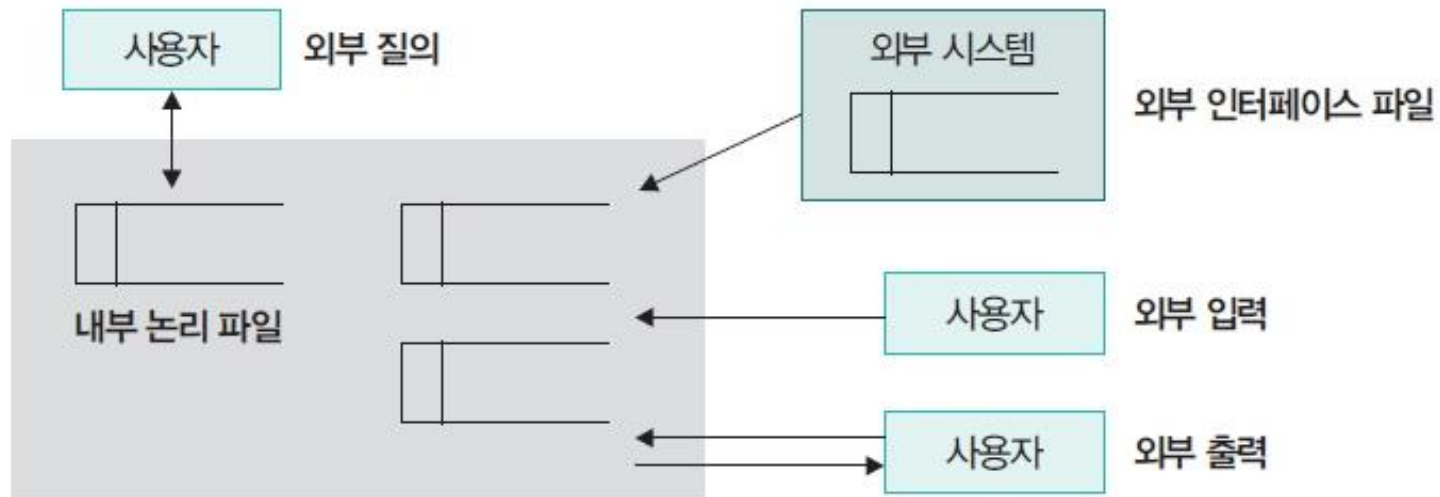
# 1. 기능 점수 개요 (1/2)

- 과거의 소프트웨어 비용 산정
  - LOC(Lines Of Codes)로 소프트웨어의 규모 파악
  - 문제 : 사용하는 프로그래밍 언어마다 기능 구현을 위한 코드 라인 수가 달라짐
- 기능점수(Function Point) 방법
  - SW의 기능과 복잡한 정도를 기준으로 규모 산정
  - IFPUG 모델 : 2003년 국제표준 채택 (ISO/IEC 20926:2003)
  - 현재 모든 소프트웨어 개발 프로젝트의 규모 및 비용 산정의 표준

# 1. 기능 점수 개요 (2/2)

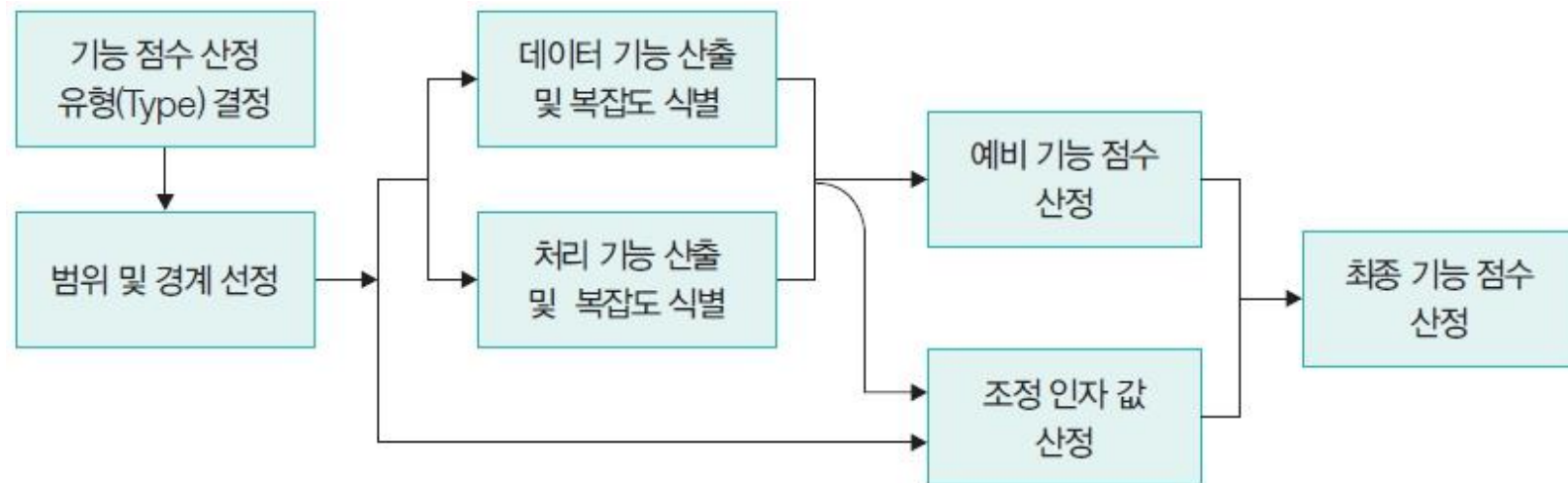
- 소프트웨어 시스템의 구성요소를 2가지로 정의
  - **처리 기능**(Transaction Function) : 외부입력, 외부출력, 외부질의
  - **데이터 기능**(Data Function) : 내부 논리파일, 외부 인터페이스 파일

→ 시스템에서 가지고있을필요없는것  
 ↳ ex) 참여계산시 환율계산



## 2. 기능 점수 산정 절차

- FP 산정절차 : IFPUG CPM(ISO/IEC 20926)



- FP산정 유형

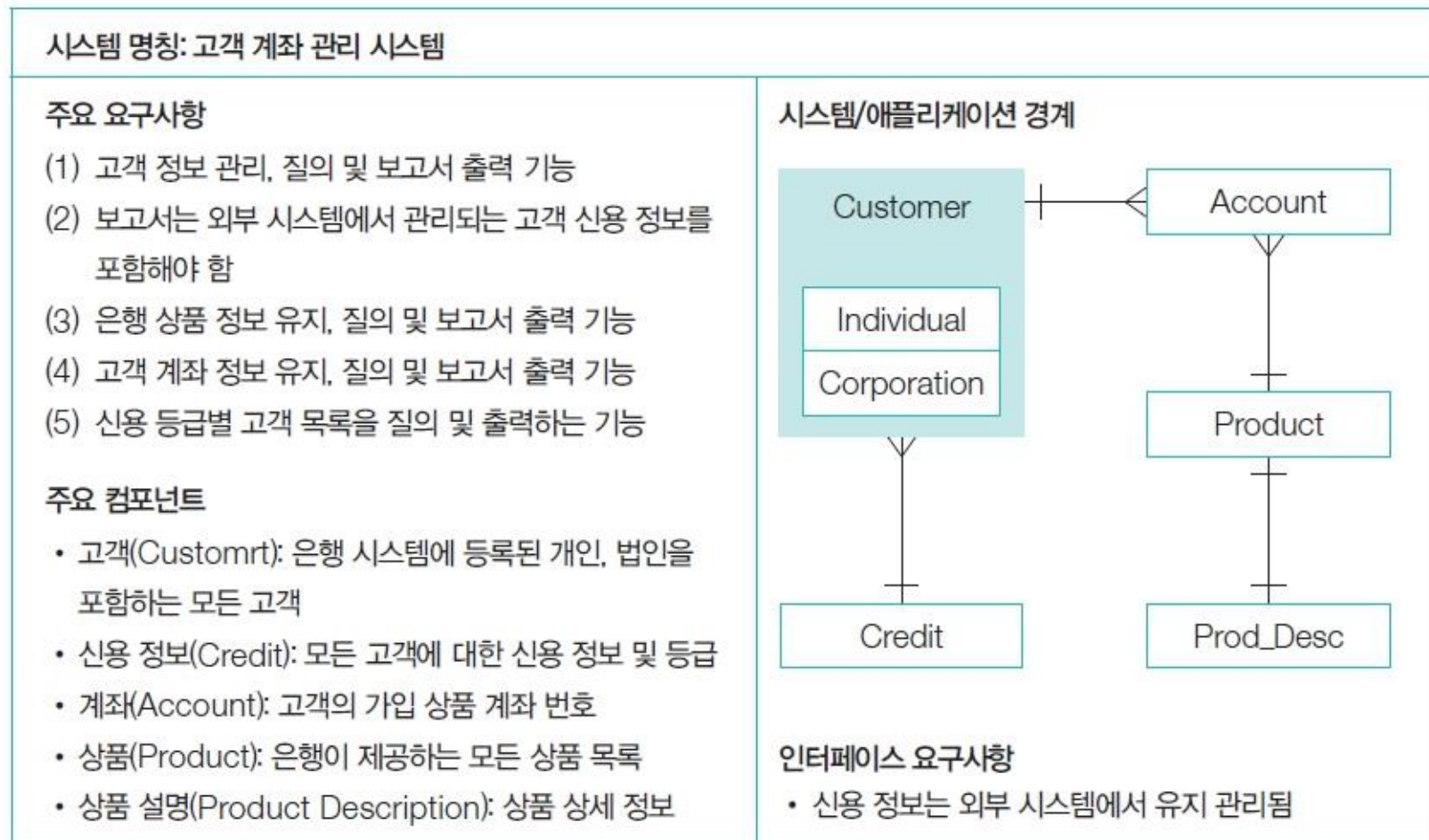
- 개발 프로젝트, 개선 프로젝트, 유지보수 프로젝트

## 2.1 범위 및 경계 선정 (1/2)

- 기능 점수 산출 범위
  - 시스템 전체(System)
  - 응용소프트웨어 전체(Application)
  - 응용소프트웨어 일부(Subset of Application)
- 고려사항 : 소프트웨어 획득 방식
  - 자체개발(In-House Development)
  - 외주개발(Outsourcing Part)
  - 패키지 구매(Package Purchase)
- 대상 시스템 범주 표현
  - Use Case Diagram, ER Diagram 등

## 2.1 범위 및 경계 선정 (2/2)

### 예제



## 2.2 데이터 기능점수 산정 (1/3)

- 데이터 기능의 유형

- 복합.기능 ← • ILF(내부논리파일, Internal Logic File) : 시스템의 내부에서 생성되고 변경 보관되는 파일 또는 레코드 타입
- EIF(외부인터페이스파일, External Interface File) : 애플리케이션 사이에서 공유하는 파일로써 참조만 하는 데이터 파일



## 2.2 데이터 기능점수 산정 (2/3)

- ILF와 EIF를 식별하여 **복잡도 가중치** 결정

*Record Element Type*

ILF와 EIF			
RET 개수	자료요소(DET) 개수		
	1-19	20-50	51+
1	단순	단순	중간
2-5	단순	중간	복잡
6+	중간	복잡	복잡

→

기능타입	복잡도 가중치		
	단순	중간	복잡
ILF	7	10	13
EIF	5	7	10

- DET**(Data Element Type) : ILF와 EIF에 유지되는 필드 중 **식별 가능하고 비반복적인 필드** → *고객정보 중*
- RET**(Record Element Type) : ILF와 EIF안에서 식별가능한 **데이터 요소의 하위그룹** *ex. 개인고객 / 법인고객의 경우 RET=2*  
 ⇒ *공통 / 각자 갖는 필드가 있음*  
 ↓  
*RET로 관리*

## 2.2 데이터 기능점수 산정 (3/3)

### ● 고객계좌 관리시스템의 자료 구성 (예시)

기능 그룹	파일명	파일 타입	필드명	필드 타입	비고
고객 정보	Customer	ILF	C_Name	DET_1	Primary Keys : C_Name, : C_IDN
			C_IDN (SSN / CRN)	DET_2	
			C_Phone	DET_3	
			C_Address	DET_4	
			C_Type_Code	DET_5	
			Credit_Grade (FK)	DET_6	
	Individual	SubFile RET	Job_Name	DET_7	
			Year_Income	DET_8	
			No_Family	DET_9	
	Corporation	SubFile RET	Total_Fund	DET_10	
			Year_Revenue	DET_11	
			Service_Type	DET_12	
			No_Employee	DET_13	
신용 등급	Credit	EIF & RET	Credit_Grade	DET_1	
			Grade_Name	DET_2	
			No_Shift	DET_3	
			Constraints	DET_4	

고객 컴포넌트 (ILF) : DET 13개, RET 2개 ⇒ 단순  
 창2 → 신용등급 컴포넌트 (EIF) : DET 4개, RET 1개 ⇒ 단순

## 2.3 처리 기능점수 산정 (1/2)

### ● 처리 기능 유형

- 외부입력(EI, External Input) : ILF에 자료를 생성, 변경, 삭제하는 것
- 외부출력(EO, External Output) : 시스템이 밖으로 방출하는 자료와 제어를 의미 (보고서나 메시지 등)
- 외부질의(EQ, External Query) : 단순한 사용자 질의

참고파일개수

EO와 EQ				EI			
FTR 개수	자료요소 개수			FTR 개수	자료요소 개수		
	1-5	6-19	20+		1-4	5-15	16+
0-1	단순	단순	중간	0-1	단순	단순	중간
2-3	단순	중간	복잡	2	단순	중간	복잡
4+	중간	복잡	복잡	3+	중간	복잡	복잡



기능 타입	복잡도 가중치		
	단순	중간	복잡
EI	3	4	6
EO	4	5	7
EQ	3	4	6

## 2.3 처리 기능점수 산정 (2/2)

### ● 고객계좌 관리시스템의 자료 구성 (예시)

유형	ILF/EIF 명칭	처리 태스크	데이터 항목 수	참조 파일 <u>FTR</u>
입력	Customer	Create_Customer	13 DETs of Customer	Customer Credit
			3 DETs of Credit	
		Update_Customer	13 DETs of Customer	Customer Credit
			3 DETs of Credit	
		Delete_Customer	2 DETs of Customer	Customer Account
			1 DETs of Account	
출력	Customer	Customer_Report	13 DETs of Customer	Customer Credit
			3 DETs of Credit	
			6 DETs of Derived	
질의	Customer	Customer_Inquiry	15 DETs of Customer	Customer

EI : 생성(FTR 2개, DET 16개) ⇒ 복잡  
 변경(FTR 2개, DET 16개) ⇒ 복잡  
 삭제(FTR 2개, DET 3개) ⇒ 단순  
 EO, EQ에 대해서도 각각 산정

## 2.4 예비 기능 점수 산정

- 모든 데이터 기능과 처리 기능에 대한 복잡도가 산정되면, 예비기능점수 산정
- 예)
  - 데이터 기능
    - Customer, Account, Product 컴포넌트 ILF 단순 → 가중치 7
    - Credit 컴포넌트 EIF 단순 → 가중치 5
  - 처리 기능
    - EI 38점, EO 23점, EQ 12점
  - 예비기능점수 = 데이터기능점수 + 처리기능점수  
$$= 26 + 38 + 23 + 12 = 99$$

## 2.5 조정 인자 값 산출 (1/2)

- 조정 인자 값 산출 과정 : 운영적·환경적 특성을 지원하는 분석
- 14개의 조정 인자

데이터 통신	분산데이터처리
성능	운영환경
트랜잭션 비율	온라인 데이터 입력
사용자 효율성	온라인 갱신
처리 복잡도	재사용성
설치용이성	운영용이성
다중설치성	변경용이성

## 2.5 조정 인자 값 산출 (2/2)

- 영향도(Degree of Influence) 결정 : 각 인자에 대해 0(영향없음) ~ 5(많음) 까지 할당
- 14개의 시스템 특성에 대하여 영향도가 결정되면 각 영향도의 합(TDI, Total Degree of Influence)을 구한 후에 **조정 인자 값 VAF 산정**

개발에서 사용  
개선 유지보수  
다른사 사용! (값들이 다르짐)

$$VAF = (TDI \times 0.01) + 0.65$$

- 단순한 배치 처리 : 15보다 작은 TDI 값이 산출됨  $\Rightarrow$  배치 처리 응용 소프트웨어의 VAF 값 0.8 수준

## 2.6 최종 기능 점수 산출

- 앞 단계에서 결정된 VAF 값을 예비 기능 점수에 곱해주는 것
  - 고객 계좌 관리 시스템의 경우 예비 기능 점수 값 : 99점
  - 고객 계좌 관리 시스템의 VAF 값 : 0.94
- 최종 기능 점수 :  $99 \times 0.94 = 93.1 \text{ FP}$  // 정수법



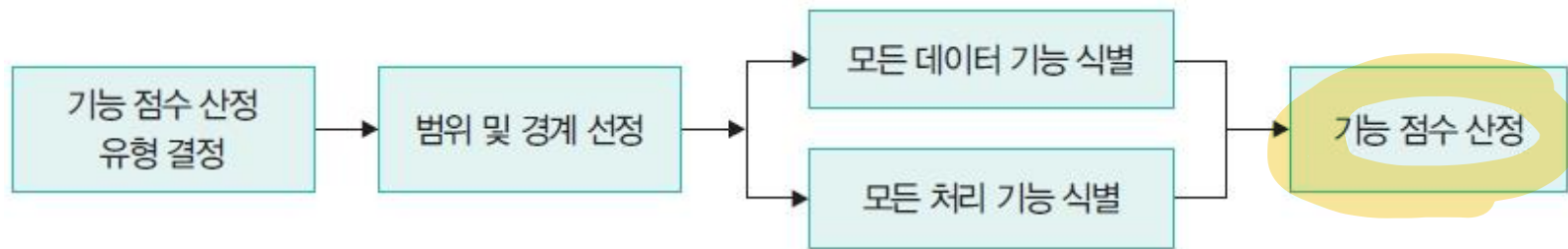
## 3. 간이법 (1/6)

- 기능의 **복잡도 판단**이 어려운 경우, **평균 복잡도를** **사용해서 계산**

구분	정규법	간이법
선정 목적	상세하고 정확한 기능 점수 측정	개략적인 예측용 기능 점수 산정
측정 항목	데이터 기능과 처리 기능	데이터 기능과 처리 기능
복잡도 적용	기능별, 단위 파일별 복잡도	평균 복잡도
적용 시기	개발 수명주기 전체에서 가능하나 상세 설계 이후 정확성이 높음	사업 준비 단계, 제안서 작성 단계 등에서 활용 가능함
장점	측정 정확도가 높으며, 다양하게 활용 가능함	신속한 측정이 가능함
단점	사업 초기 단계에서 적용하기 어려움	제한적 데이터 사용으로 정확성이 부족함

# 3. 간이법 (2/6)

## ● 간이법에 의한 기능 점수 산정 절차



기능타입	복잡도 가중치		
	단순	중간	복잡
ILF	7	10	13
EIF	5	7	10
EI	3	4	6
EO	4	5	7
EQ	3	4	6

간이법  
  
 조정인자 X

기능타입	평균가중치
ILF	7.3
EIF	5.4
EI	4.0
EO	5.1
EQ	3.8

## 3. 간이법 (3/6)

- 데이터 기능과 처리 기능의 카운트(예시)

단위 업무 식별자	데이터 기능		처리 기능				
	ILF	EIF	EI			EO	EQ
			입력	수정	삭제		
P1	1	0	1	1	1	0	1
P2	2	0	2	1	1	1	2
P3	0	1	1	1	0	1	2
P4	1	0	2	1	1	0	2
합계	5	1	6	4	3	2	7

$$FP = (5 * \underline{7.3}) + (1 * \underline{5.4}) + (13 * \underline{4.0}) + (2 * \underline{5.1}) + (7 * \underline{3.8})$$

평균기능치 곱

# 3. 간이법 - 산정예제 (4/6)

☆ 사적적인 조회(EQ)가 아닌 EO ⇒ "부서단위로" 지정하므로

1. 담당자는 고객주문을 입력,수정,삭제 한다(고객DB) → IF
2. 인사담당자는 사원목록을 부서단위로 조회한다(사원DB, 부서DB)
3. 인사담당자는 사원복록을 단순 출력한다(사원DB) EQ
4. 인사담당자는 일정 금액 이상의 급여 수령자 사원목록을  
검색한다(사원DB, 급여DB)
5. 원화에 대한 미국 달러(USD) 가치를 찾기 위해 A은행 외환DB에서  
환율을 검색한다(외환DB) EIF
6. 인사담당자는 5년 경력 이상이고, 해당 직무 수행경험이 있는 사원목록을  
추출한다(사원DB)
7. 인사담당자는 신입/경력사원 입사 시 사원 파일을 갱신한다(사원DB)
8. 인사담당자는 외국 사원 입사 시 사원의 급여를 결정하기 위해 H연합회  
통화정보를 참조한다(사원DB, 급여DB, 통화DB)
9. 인사담당자는 사원현황을 엑셀 파일을 업로드 시킨다(사원DB)

만들어야 된다면 IF  
값대신면 EIF

# 3. 간이법 - 산정예제 (5/6)

## 1. 데이터기능산정

데이터베이스	기능유형	평균복잡도
고객 DB	ILF	7.4
사원 DB	ILF	7.4
부서 DB	ILF	7.4
급여 DB	ILF	7.4
외환 DB	EIF	5.5
통화 DB	EIF	5.5

- 데이터기능점수 =  $4 \times 7.4 + 2 \times 5.5 = 40.6$

# 3. 간이법 - 산정예제 (6/6)

## 2. 처리 기능산정

트랜잭션	유형	복잡도
담당자는 고객주문을 입력,수정,삭제 한다	EI	4.3
인사담당자는 사원목록을 부서단위로 조회한다	EO	5.4
인사담당자는 사원목록을 단순 출력한다	EQ	3.8
인사담당자는 일정 금액 이상의 급여 수령자 사원목록을 검색한다	EO	5.4
원화에 대한 미국 달러(USD) 가치를 찾기 위해 A은행 외환DB에서 환율을 검색한다	EO	5.4
인사담당자는 5년 경력 이상이고, 해당 직무 수행경험이 있는 사원목록을 추출한다	EO	5.4
인사담당자는 신입/경력사원 입사 시 사원 파일을 갱신한다	EI	4.3
인사담당자는 외국사원 입사시 급여를 결정하기 위해 H연합회 통화정보를 참조한다	EO	5.4
인사담당자는 사원현황을 엑셀 파일을 업로드 시킨다	EI	4.3

$$FP = \text{데이터 기능점수} + \text{처리 기능점수} = 40.6 + 43.7 = 84.3$$

## 4. 기능 점수의 활용 (1/3)

- IFPUG에서 기능 점수와 프로그래밍 언어별 LOC에 대한 매핑 테이블 제공

프로그래밍 언어		LOC	프로그래밍 언어	LOC
Assembly	Basic	320	FORTRAN	107
	Macro	213		
BASIC		107	HTML 3.0	15
Visual Basic		29	LISP	64
C		128	JAVA	53
C++		53	PL/I	80
COBOL		107	SQL	13
DELPHI		29	Power Builder	16

## 4. 기능 점수의 활용 (2/3)

ex) FP = 46.7 만개  
 생산성 = 1000 LOC / MM  
 1FP = 50 LOC in Java 일때  
 개발자 MM은?  
 $\Rightarrow \frac{1}{2} LOC = 46.7 \times 50 = 2335$   
 $MM = 2335 LOC / 1000 LOC / MM$   
 $= 2.335$

- 기능점수 1포인트당 얼마의 LOC로 고려될 수 있는가?

- 객체지향언어(C++, JAVA) : 53 LOC } → line수추가능
- 절차적 언어(C) : 128 LOC

- 예) 고객 계좌 관리 시스템의 기능 점수 93.1

- C언어로 개발 시  $93.1 \times 128 = 11,916$  LOC

- C++, JAVA 구현 시  $93.1 \times 53 = 4,934$  LOC

- 2020년 한국소프트웨어산업협회 : 기능 점수당 표준 단가는 553,114 원 → 개발비추정

- 개발비 =  $93.1 \times 553,114 = 51,494,913$  원



## 4. 기능 점수의 활용 (3/3)

- 기능점수와 품질 척도
  - 생산성(Productivity)

$$\text{기능 점수당 개발 시간} = \frac{\text{프로젝트에 투입된 총 개발 시간}}{\text{배포된 소프트웨어의 기능 점수}}$$

- 신뢰성(Reliability)

$$\text{신뢰성 Reliability} = \frac{\text{소프트웨어 고장의 횟수}}{\text{총 기능 점수 값}}$$