

C프로그래밍

13 SW 개발방법론과 예제풀이

공지사항

• 코딩테스트

- ✓ 6/2(월) 오후 문제 공지 (공지게시판)
- ✓ 6/9(월) 강의시간에 코딩테스트 수행
 - ➔ 현장에서 공지되는 자리에 착석, 테스트 수행
 - ➔ 부정행위 적발 시 F학점 부여

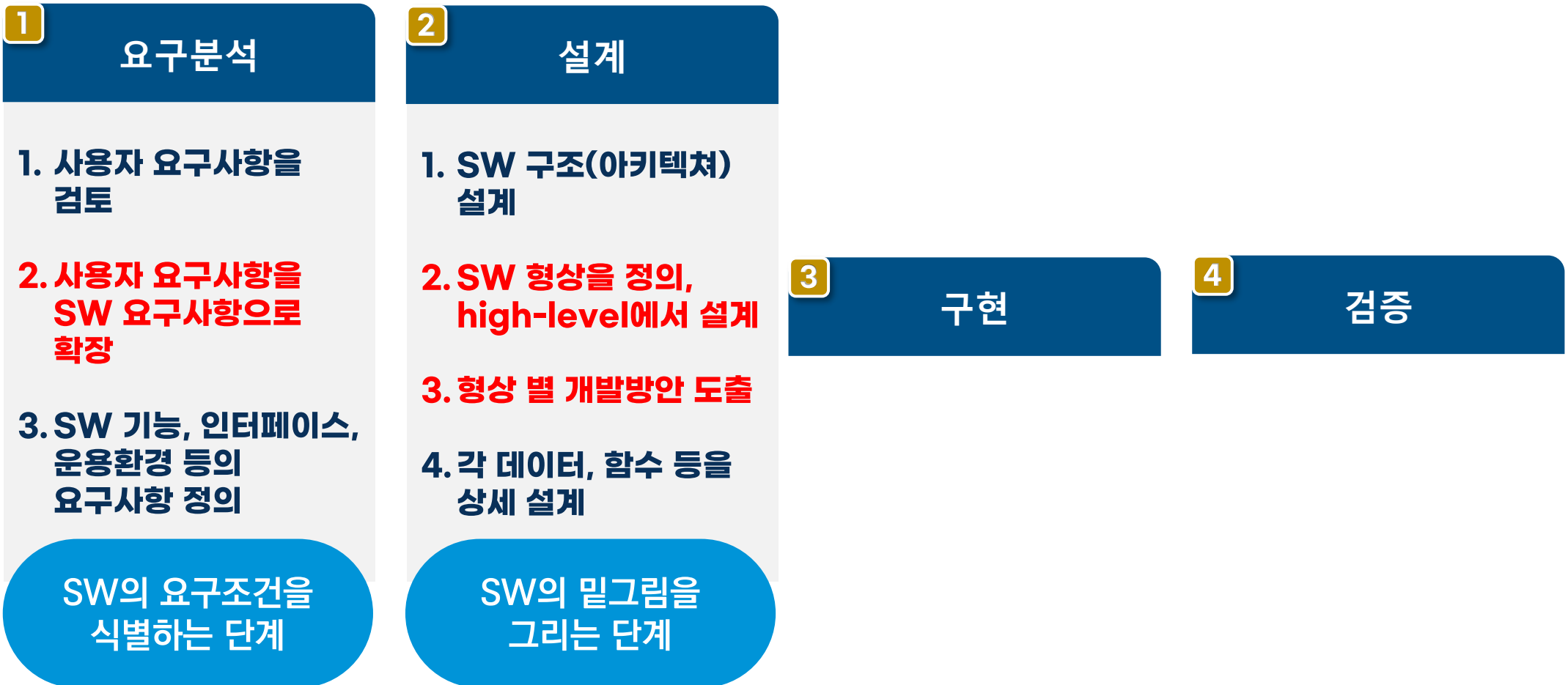
• 기말고사

- ✓ 6/14(토) 13:00 ~ 15:00, 참빛관 지하 B106호
- ✓ 시험범위: 1주차 ~ 13주차 강의내용

SW 개발 방법론 예시

SW 개발 절차 예시

- SW 개발 프로세스에 대한 연구 : 시스템공학, 소프트웨어공학
- 본 설명은 해당 내용을 배우지 않은 상태에서 간단한 프로그램 설계에 활용 할 수 있는 일부분임



[6주차 예제 04] 별표(*) 출력

- **사용자로부터 입력 받은 숫자만큼 별표(*)를 출력**

- ✓ 사용자가 0을 입력 시 다시 입력 받음
- ✓ 사용자가 11 이상 입력 시 “그렇게 많은 별표를 출력할 수 없습니다.”라는 에러 메시지와 함께 프로그램을 종료

<프로그램 구현 힌트>

- **무한 반복문/중첩 반복문 사용**
- **continue와 break 활용**

요구분석과 설계 예시 (1/6)

- **6-2 수업 시 구현해본 “제한된 별표 출력 ” 예제**
- **사용자 요구사항을 검토**
 1. 사용자로부터 입력 받은 숫자만큼 화면에 별을 출력
 2. 사용자가 0을 입력하면 다시 입력 받음
 3. 사용자가 11 이상을 입력하면 에러 메시지를 출력하고 프로그램을 종료
 4. 무한 반복문/중첩 반복문 사용
 5. continue와 break를 사용하여 작성

요구분석과 설계 예시 (2/6)

- **SW 요구사항으로 확장**

1. 사용자로부터 입력 받은 숫자만큼 화면에 별을 출력

- 1-1 사용자로부터 숫자를 입력 받아라

- 1-2 주어진 숫자만큼 화면에 별을 출력하라

2. 사용자가 0을 입력하면 다시 입력 받음

- 2-1 사용자가 0을 입력하는지 검사하라

- 2-2 0을 입력하면 화면에 별을 출력하지 말고, 숫자를 입력 받아라 → 1-1

요구분석과 설계 예시 (3/6)

• SW 요구사항으로 확장

3. 사용자가 11 이상을 입력하면 에러 메시지를 출력하고 프로그램을 종료

3-1 사용자가 11 이상을 입력하는지 검사하라

3-2 11 이상을 입력하면 에러 메시지를 출력하라

3-3 에러 메시지를 출력하면 프로그램을 종료하라 → 유일한 종료조건

3-4 11 미만을 입력하면 프로그램을 종료하지 마라 → 1과 2의 무한 반복

4. 무한 반복문/중첩 반복문 사용

4-1 무한 반복문을 사용하라 → 특정 조건에 의해 종료시까지 무한 반복

4-2 중첩 반복문을 사용하라 → 반복문의 중첩되는 형태를 활용

5. continue와 break를 사용하여 작성

5-1 continue를 사용하라 → 특정 조건이 충족되면 반복문의 처음으로 돌아가라

5-2 break를 사용하라 → 특정 조건이 충족되면 반복문을 종료하라

요구분석과 설계 예시 (4/6)

- **high-level에서의 설계 - 1단계 : 요구사항을 동작 순서에 맞추어 배치**
 - ✓ 다양한 방법이 있지만, 지금 단계에서는 자연어(pseudo-code)로 하는 것을 추천

// 1-1 사용자로부터 숫자를 입력 받아라
사용자로부터 숫자를 입력 받음

// 2-1 사용자가 0을 입력하는지 검사하라

// 2-2 0을 입력하면 화면에 별을 출력하지 말고, 숫자를 입력 받아라 → 1-1
숫자가 0이면

다시 숫자를 입력 받음

// 3-1 사용자가 11 이상을 입력하는지 검사하라

// 3-2 11 이상을 입력하면 에러 메시지를 출력하라

// 3-3 에러 메시지를 출력하면 프로그램을 종료하라 → 유일한 종료조건

숫자가 0이 아니면서 11 이상이면

에러메시지를 출력

프로그램을 종료

// 1-2 주어진 숫자만큼 화면에 별을 출력하라

숫자가 0이 아니면서 11 미만이면

숫자 크기 만큼의 별을 출력

// 3-4 11 미만을 입력하면 프로그램을 종료하지 마라 → 1과 2의 무한 반복

다시 사용자로부터 숫자를 입력 받음

요구분석과 설계 예시 (5/6)

- high-level에서의 설계 - 2단계

: 프로그래밍 언어 문법에 맞추어 업그레이드

1. 무한 반복함 // 4-1 무한 반복문을 사용하라
 2. 사용자로부터 숫자를 입력 받음
3. 숫자가 0인지 검사함
 - 3.1 숫자가 0 이면 1로 돌아감
// 5-1 continue를 사용하라
4. 숫자가 11 이상인지 검사함
 - 4.1 에러메시지를 출력
 - 4.2 숫자가 11 이상이면 6으로 넘어감
// 5-2 break를 사용하라
5. 숫자 크기만큼 반복함 //4-1 중첩 반복문을 사용하라
 - 5.1 화면에 별을 출력
6. 프로그램을 종료

요구분석과 설계 예시 (6/6)

- high-level 설계 내용별 개발방안 도출

1. 무한 반복함 → while(1)문 사용
 2. 사용자로부터 숫자를 입력 받음 → 입력 변수 선언, scanf()
3. 숫자가 0인지 검사함 → if문
 - 3.1 숫자가 0 이면 1로 돌아감 → continue
4. 숫자가 11 이상인지 검사함 → if문
 - 4.1 에러메시지를 출력 → printf(“그렇게 많은… ”)
 - 4.2 숫자가 11 이상이면 6으로 넘어감 → break
5. 숫자 크기만큼 반복함 → for문
 - 5.1 화면에 별을 출력 → printf(“*”)
6. 프로그램을 종료 → return

예 제

[예제 01] 성적관리 프로그램 작성 review

영희, 철수, 영철, 영수 등 4명의 학생의 국어, 영어, 수학, 과학 점수(1~9점)를 입력 받아 배열에 저장하고, 각 학생의 총점과 과목별 총점을 계산하여 저장하고, 출력하는 프로그램을 작성하시요.

<1단계> 점수를 입력 받아서 저장

	국어	영어	수학	과학	학생별 총점
영희	2	3	4	5	
철수	5	5	4	7	
영철	3	7	2	4	
영수	8	6	2	9	
과목별 총점					

<2단계> 총점을 계산/저장하고 출력

	국어	영어	수학	과학	학생별 총점
영희	2	3	4	5	14
철수	5	5	4	7	21
영철	3	7	2	4	16
영수	8	6	2	9	25
과목별 총점	18	21	12	25	

1. 입력 받은 점수와 총점은 5X5 2차원 배열에 저장하여 출력하시요.

[예제 01] 성적관리 프로그램 v2.0

main()를 포함하여 총 4개의 함수로 구성되는 프로그램으로 업그레이드

함수를 직접 설계

- (1) 학생의 이름과 배열을 전달받아 학생의 점수를 입력 받고, 배열에 저장하는 `input_score()`를 만든다.
- (2) 학생들의 점수가 저장된 배열을 전달받아 학생 별, 과목 별 총점을 계산하고 배열에 저장하는 `calc_total()`를 만든다.
- (3) 학생들의 점수와 총점이 저장된 배열을 전달받아 각 학생 별, 과목 별 총점을 출력하는 `print_score()`를 만든다.
- (4) `main()`는 배열을 생성하고, 필요한 함수들을 호출하여 프로그램을 완성한다.

[예제 01] 성적관리 프로그램 v2.0

(1) 학생의 이름과 배열을 전달받아 학생의 점수를 입력 받고, 배열에 저장하는 `input_score()`를 만든다.

✓ 학생의 이름을 전달받아야 함

이름은 문자열 → 문자 배열 또는 문자형 포인터

`void input_score(char* name);` 또는 `void input_score(char name[]);`

✓ 점수를 저장할 2차원 배열을 전달받아야 함

2차원 배열을 전달하는 방법은?

`void input_score(int score[5][5]);`

이를 종합하면?

`void input_score(char *name, int score[5][5]);`

<도전>

과목의 이름도
필요하지 않을까?

[예제 01] 성적관리 프로그램 v2.0

✓ main()에서 input_score()를 사용하는 방법

1. 이름/과목을 저장하는 배열과 점수를 저장하는 배열을 생성한다.

```
char name[ ][5] = {"영희", "철수", "영철", "영수"};
```

```
// name[0] → "영희", name[1] → "철수"
```

```
char subject[ ][5] = {"국어", "영어", "수학", "과학"};
```

```
int score[5][5] = {0};
```

2. "영희"부터 점수를 입력 받기 위해 input_score()를 호출한다.

```
input_score(name[0], score);
```

반복문을 사용해서 input_score()를
호출하자.

3. 다른 학생에 대해서도 반복적으로 점수를 입력 받기 위해 input_score()를 호출한다.

[예제 02] 입력 받은 숫자를 정렬하는 프로그램 review

예제 설명

사용자로부터 10개의 숫자열을 입력 받아 정렬하여 출력하는 프로그램을 만드세요.

실행 결과

입력 숫자열 : 6 2 4 5 8 2 0 9 1 5
정렬 숫자열 : 9 8 6 5 5 4 2 2 1 0

(1) 매개변수로 받은 숫자열의 값들을 내림차순으로 정렬하는

void sort_num(int* num)를 만들어 사용

(2) 숫자열의 입력과 출력은 모두 main()에서 수행

[예제 02] 입력 받은 숫자를 정렬하는 프로그램 v2.0

예제 설명

사용자로부터 100개 이내의 숫자열과 원하는 정렬 방식을 입력 받고, 정렬 결과를 출력하는 프로그램을 작성하시요.

실행 결과

입력 숫자열 : 6 2 4 5 8 2 6 9

정렬 방식 : 오름차순

정렬 숫자열 : 2 2 4 5 6 6 8 9

- (1) 사용자로부터 입력을 받는 `input_num()`를 만들어 사용한다.
- (2) 매개변수로 받은 숫자열의 값들을 정렬하는 `sort_num()`를 만들어 사용한다.
- (3) 사용자가 숫자 0을 입력하면 더 이상 숫자를 입력 받지 않는다. 숫자 0은 입력 숫자열에서 제외 한다. 정렬 방식은 “오름차순“, “내림차순 ” 이 있다.
- (4) `main()`는 필요한 배열을 생성하고, 함수들을 호출하여 프로그램을 완성한다.

[예제 03] 성적관리 프로그램 v3.0

최대 10명의 학생들에 대해 국어, 영어, 수학, 과학 과목의 점수를 입력받고, 총점을 출력하는 프로그램을 작성

- (1) 학생의 이름을 저장할 배열을 전달받아 학생의 숫자와 각 학생의 이름을 입력받고, 배열에 저장하는 `input_name()` 작성
- (2) 한명의 학생에 대해 과목별 점수를 모두 입력받아 저장하는 `input_score()` 작성
- (3) 학생들의 모든 과목별 점수 또는 과목의 모든 학생들 점수를 입력받아 총점을 계산, 반환하는 `calc_total()`를 작성
- (4) 학생의 이름 또는 과목명과 총점을 입력받아 그 내용을 출력하는 `print_score()`를 작성
- (5) `main()`은 필요한 배열을 생성하고, 필요한 함수들을 호출하여 프로그램을 완성

Q & A