

# CSED101. Programming & Problem solving

## Spring, 2018

### Programming Assignment #1 (30 points)

이원기(wklee@postech.ac.kr)

■ **Due:** 2018.03.19 23:59

■ **Development Environment:** GNU C Compiler (GCC) and Vi Editor (Editor is optional)

#### ■ 제출물

- **C Code files (\*.c)**
  - 프로그램의 소스 코드를 이해하기 쉽도록 반드시 주석을 붙일 것.
- **보고서 파일** (.doc(x) or .hwp) 예) assn1.doc(x) 또는 assn1.hwp
  - AssnReadMe.pdf 를 참조하여 작성할 것.
  - 리눅스 서버에 접속하는 것부터 시작해서 프로그램 컴파일 및 실행하는 과정까지를 화면 캡처하여 보고서에 포함시키고 간단히 설명 할 것!!
  - 명예서약(Honor code): 표지에 다음의 내용을 포함한다. “나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.” 보고서 표지에 명예서약이 없는 경우는 과제를 제출하지 않은 것으로 처리한다.
  - 소스코드와 보고서 파일을 LMS를 이용하여 제출한다.

#### ■ 주의사항

- 각 문제에 해당하는 요구사항을 반드시 지킬 것.
- 모든 문제의 출력 형식은 아래의 예시들과 동일해야 하며, 같지 않을 시는 감점이 된다.
- 각 문제에 제시되어 있는 파일이름으로 제출 할 것. 그 외의 다른 이름으로 제출하면 감점 또는 0점 처리된다.
- 컴파일 & 실행이 안되면 무조건 0점 처리된다.
- 하루 late시 20%가 감점되며, 3일 이상 지나면 받지 않는다. (0점 처리)
- 부정행위에 관한 규정은 POSTECH 전자컴퓨터공학부 학부위원회의 'POSTECH 전자컴퓨터공학부 부정행위 정의'를 따른다. (LMS의 과목 공지사항의 제목 [document about cheating]의 첨부파일인 disciplinary.pdf를 참조할 것.)
- 이번 과제에서는 추가 기능 구현에 대한 추가 점수는 없습니다.

## (15 pts) Problem 1: 초를 시, 분, 초로 출력하기

### (설명)

초를 입력으로 받아 '시 : 분 : 초' 형식으로 ASCII code를 이용해 출력하는 프로그램을 작성하시오.

### (요구사항)

- 0 ~ 359999 사이의 사용자 입력 값을 하나의 정수형 변수(int)에 입력 받는다.  
(예외 처리는 하지 않으며, 잘못된 입력은 고려하지 않는다.)
- 시, 분, 초의 각 출력은 두 자리 정수이다 (1 : 1 : 1(X) → 01 : 01 : 01(O)).  
따라서, 아래의 예와 같이, 시, 분, 초의 각 자리를 따로 저장할 변수를 사용한다.

```
hour1, hour2; // 각각 '시'의 첫번째와 두번째 자리 수를 저장할 변수
min1, min2;   // 각각 '분'의 첫번째와 두번째 자리 수를 저장할 변수
sec1, sec2;   // 각각 '초'의 첫번째와 두번째 자리 수를 저장할 변수
```

- 시, 분, 초의 출력은 모두 **ASCII code 값**을 이용해 아래의 예와 같이 출력한다.

```
printf( "%d%d : %d%d : %d%d\n", hour1, hour2, min1, min2, sec1, sec2 ) // 잘못된 방법
printf( "%2d : %2d : %2d\n", hour, min, sec ); // 잘못된 방법
printf( "%c%c : %c%c : %c%c\n", hour1, hour2, min1, min2, sec1, sec2 ); // 올바른 방법
```

### (주의사항)

- 파일 이름은 **"assn1\_1.c"**로 저장 할 것
- 보고서는 **"assn1.doc"** or **"assn1.hwp"**로 저장 할 것 (보고서는 통합하여 작성)
- 출력은 아래의 "실행예제"와 **동일하도록** 작성 할 것 (모든 공백은 'space' 사용)
- 아직 수업시간에 다루지 않은 if, for, while 문 등은 사용하지 않는다. (사용 시 감점)

(실행예제) 빨간 밑줄은 사용자가 입력하는 부분에 해당됨.

```
[wklee@programming assn1]$ gcc assn1_1.c -o assn1_1.out
[wklee@programming assn1]$ ./assn1_1.out
enter time in seconds(0~359999)> 351244
97 : 34 : 04
[wklee@programming assn1]$
```

## (15 pts) Problem 2: 입력 받은 수들의 다양한 통계 값 구하기

### (설명)

사용자로부터 입력 받은 수들의 총합, 평균, 분산, 표준편차를 계산하여 출력하는 프로그램을 작성하시오.

1. 평균을 구하는 공식은 다음과 같다.

$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{N}$$

2. 편차를 구하는 공식은 다음과 같으며, 특정 데이터가 평균과 얼마나 떨어져 있는지를 나타내는 통계 값이다.

$$d_i = x_i - \bar{x}$$

3. 분산을 구하는 공식은 다음과 같으며, 데이터들이 평균으로부터 얼마나 떨어진 곳에 분포하는지 가늠하는 통계 값이다.

$$\sigma^2 = \frac{d_1^2 + d_2^2 + \dots + d_n^2}{N}$$

4. 표준편차를 구하는 공식은 다음과 같으며, 제곱근을 이용해 분산을 실제 값으로 근접시킨 수치이다.

$$\sigma = \sqrt{\frac{d_1^2 + d_2^2 + \dots + d_n^2}{N}}$$

### (요구사항)

1. 사용자로부터 6개의 수를 입력 받는다.  
(더 많거나, 더 적은 수의 입력은 고려하지 않는다.)
2. 입력 및 출력 모두 실수형(float)변수를 이용한다.
3. 출력은 소수 둘째 자리 까지만 출력한다.
4. <math.h> 헤더를 main문 상단에 include하여, 제곱근 및 거듭제곱 계산을 수행한다.  
라이브러리 사용방법은 다음과 같다.

```
#include<math.h>
int num = 4;
int sqrt_num = sqrt(num); // 4의 제곱근(=2)을 구할 수 있다.
int pow_num = pow(num, 2); // 4의 거듭제곱(=16)을 구할 수 있다.
```

5. Linux환경에서 <math.h>가 포함된 소스코드를 컴파일 하는 방법은 다음과 같다.

```
gcc assn1_2.c -o assn1_2.out -lm
```

(주의사항)

- 파일 이름은 **"assn1\_2.c"**로 저장 할 것
- 보고서는 **"assn1.doc"** or **"assn1.hwp"**로 저장 할 것 (보고서는 통합하여 작성)
- 출력은 아래의 "실행예제"와 **동일하도록** 작성 할 것 (모든 공백은 'space' 사용)
- 아직 수업시간에 다루지 않은 if, for, while 문 등은 사용하지 않는다. (사용 시 감점)

(실행예제) 빨간 밑줄은 사용자가 입력하는 부분에 해당됨.

```
[wklee@programming assn1]$ gcc assn1_2.c -o assn1_2.out -lm
[wklee@programming assn1]$ ./assn1_2.out
Enter 6 numbers> 35.2 -4.12 444 15 -10.43 4441
Sum: 4920.65
Average: 820.11
Variance: 2647623.25
Standard deviation: 1627.15
[wklee@programming assn1]$
```