

Jaewon Choi  
Keunsan Park  
Haeun Lee  
(csl.psta@gmail.com)

Systems Software &  
Architecture Lab.

Seoul National University

Spring 2023

# 4190.103A-001: Programming Practice Lab. 3



# Lab. 2 과제 풀이

## Lab. 2) 과제 1 - 문제

130이라는 숫자를 10진수, 8진수, 16진수로 출력해  
봅니다.

Enter를 사용하여 출력 사이에 공백을 둡니다.  
출력은 총 3줄이어야 합니다.

## Lab. 2) 과제 1 - 풀이

```
#include <stdio.h>

int main()
{
    int num = 130;

    printf("%d\n", num);
    printf("%o\n", num);
    printf("%x\n", num);

    return 0;
}
```

```
#include <stdio.h>

int main()
{
    int num = 130;

    printf("%d\n%o\n%x\n", num, num, num);

    return 0;
}
```

## Lab. 2) 과제 2 - 문제

사용자의 성적을 입력 받고,  
성적이 90점 이상이면 A  
90점 미만이고 80점 이상이면 B,  
80점 미만이고 70점 이상이면 C,  
70점 미만이면 D를 출력하는 프로그램을 작성하세요.

```
if (조건 1) {  
    // 코드1  
} else if (조건 2) {  
    // 코드2  
} else {  
    // 코드3  
}
```

형식을 사용하면 됩니다. 필요에 따라 else if 문을 여러 개 사용하세요.

## Lab. 2) 과제 2 - 풀이

```
#include <stdio.h>

int main(void) {
    int score;
    char grade;

    scanf("%d", &score);

    if (score >= 90)
    {
        grade = 'A';
    }
    else if (score >= 80)
    {
        grade = 'B';
    }
    else if (score >= 70)
    {
        grade = 'C';
    }
    else
    {
        grade = 'D';
    }

    printf("%c\n", grade);

    return 0;
}
```

```
#include <stdio.h>

int main(void) {
    int score;
    char grade;

    scanf("%d", &score);

    if (score >= 90)
        grade = 'A';
    else if (score >= 80)
        grade = 'B';
    else if (score >= 70)
        grade = 'C';
    else
        grade = 'D';

    printf("%c\n", grade);

    return 0;
}
```

## Lab. 2) 과제 3 - 문제

문자를 입력받고, 이를 변수에 저장한 뒤 출력합니다.

실수 1673.1832를 입력받고, 이를 float 변수에 저장한 뒤 출력합니다.

실수는 소수점 아래 6자리까지만 출력하도록 합니다.

입출력은 총 4줄 이어야 합니다. 출력 예시는 아래와 같습니다.

A

A

실수

실수

## Lab. 2) 과제 3 - 풀이

```
#include <stdio.h>

int main(void) {
    char c;
    float f;

    scanf("%c", &c);
    printf("%c\n", c);

    scanf("%f", &f);
    printf("%f\n", f);

    return 0;
}
```



주식

# 주석

- 프로그래머의 편의를 위한 메모기능
- 프로그램 수행에는 아무런 영향을 주지 않음
- // : 하나의 행을 주석으로 처리
- /\* ~ \*/ : 시작과 끝의 모든 행을 주석으로 처리

```
/*  
This program is for ...  
blar,blar,blar....  
....  
*/  
  
// if A is positive...  
if (a > 0) {  
    printf("%d is a positive number.\n", a);  

```

# 관계연산자

# 관계연산자

- 두개의 피연산자를 대소 비교 후 참이면 1, 거짓이면 0을 반환

$X == Y$	X와 Y의 값이 같다
$X != Y$	X와 Y의 값이 다르다
$X < Y$	X가 Y보다 작다
$X <= Y$	X가 Y보다 작거나 같다
$X > Y$	X가 Y보다 크다
$X >= Y$	X가 Y보다 크거나 같다

# 관계연산자 - 예시

```
1  #include <stdio.h>
2
3  int main(void) {
4      int a, b;
5
6      scanf("%d %d", &a, &b);
7      printf("a : %d\n", a);
8      printf("b : %d\n", b);
9
10     printf("a == b : %d \n", a == b);
11     printf("a != b : %d \n", a != b);
12     printf("a > b : %d \n", a > b);
13     printf("a < b : %d \n", a < b);
14     printf("a >= b : %d \n", a >= b);
15     printf("a <= b : %d \n", a <= b);
16
17     return 0;
18 }
```

```
1000 1000
a : 1000
b : 1000
a == b : 1
a != b : 0
a > b : 0
a < b : 0
a >= b : 1
a <= b : 1
```

# 관계연산자 - 우선순위

- 산술연산자의 우선순위
  - +, - 보다 \*, %, / 가 우선
  - $a+b*2$  :  $a+(b*2)$
- 산술연산자와 관계연산자의 우선순위
  - 관계연산자보다 산술연산자의 우선순위가 더 높음
  - $a + b <= a * b$  :  $(a+b) <= (a * b)$

```
int a, b;

scanf("%d %d", &a, &b);

printf("a + b * 2 : %d\n", a + b * 2);
printf("a + (b * 2) : %d\n", a + (b * 2));
printf("(a + b) * 2 : %d\n", (a + b) * 2);
printf("a + b <= a * b : %d\n", a + b <= a * b);
printf("(a + b) <= (a * b) : %d\n", (a + b) <= (a * b));
printf("a + (b <= a) * b : %d\n", a + (b <= a) * b);
```

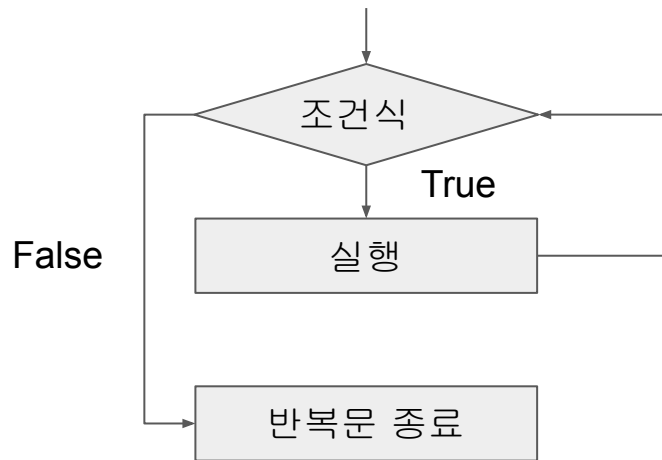
```
2 2
a + b * 2 : 6
a + (b * 2) : 6
(a + b) * 2 : 8
a + b <= a * b : 1
(a + b) <= (a * b) : 1
a + (b <= a) * b : 4
```

반복문

# 반복문 - while문

- while문
  - 조건식이 참일 경우 계속 반복하는 문법

```
while(조건식) {  
    실행문 1;  
    실행문 2;  
    ...  
}
```





# 반복문 - while문

- 입력받은 숫자만큼 \* 을 출력하는 프로그램

```
#include<stdio.h>

int main(void){
    int i, number;
    scanf("%d", &number);

    i = 0;
    while(i < number)
    {
        printf("*");
        i = i + 1;
    }

    printf("\ni : %d", i);

    return 0;
}
```

```
/* 코드가 실행되는 중입니다... */
10
*****
i : 10
/* 코드 실행이 완료되었습니다! */
```

# 반복문 - while문

- 횟수를 증가시키는 부분을 ++, -- 연산자로 대체가능
  - $i = i + 1 \rightarrow i++$
  - $i = i - 1 \rightarrow i--$

```
#include<stdio.h>

int main(void){
    int i, number;
    scanf("%d", &number);

    i = 0;
    while(i < number)
    {
        printf("*");
        i++;
    }

    printf("\ni : %d", i);

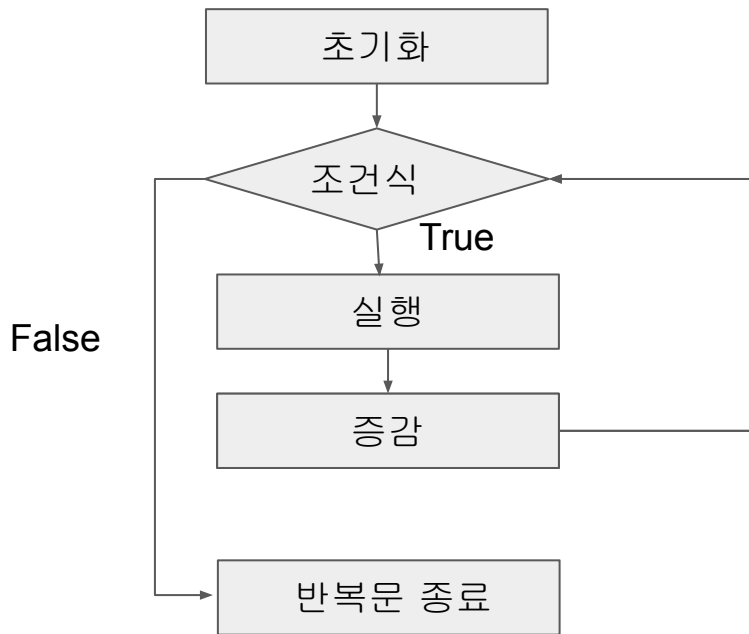
    return 0;
}
```

```
/* 코드가 실행되는 중입니다... */
10
*****
i : 10
/* 코드 실행이 완료되었습니다! */
```

# 반복문 - for문

- for문
  - while문과 동일한 용도로 반복을 하기 위한 문법
  - 초기화 및 증감식을 함께 선언할 수 있음

```
for (초기화 ; 조건식 ; 증감) {  
    실행문 1;  
    실행문 2;  
    ...  
}
```



# 반복문 - for문

- 입력받은 숫자만큼 \* 을 출력하는 프로그램

```
#include<stdio.h>

int main(void){
    int i, number;
    scanf("%d", &number);

    for (i = 0; i < number; i++)
    {
        printf("*");
    }

    printf("\ni : %d", i);

    return 0;
}
```

```
/* 코드가 실행되는 중입니다... */
10
*****
i : 10
/* 코드가 실행이 완료되었습니다! */
```

# 반복문 - for문

- 증감식은 2개 이상의 식을 선언할 수도 있음

```
#include <stdio.h>

int main()
{
    // Write your code here
    int number;
    int i = 0;
    int j = 0;

    scanf("%d", &number);

    for (i = 0; i < number; i++, j += 2)
    {
        printf("i : %d, j : %d\n", i, j);
    }

    return 0;
}
```

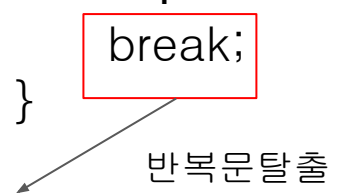
```
/* 코드가 실행되는 중입니다... */
5
i : 0, j : 0
i : 1, j : 2
i : 2, j : 4
i : 3, j : 6
i : 4, j : 8

/* 코드가 실행이 완료되었습니다! */
```

# 반복문 - break문

- break문
  - 반복문을 탈출하기 위함
  - for, while문 안에서 사용 가능

```
while(조건식) {  
    실행문 1;  
    실행문 2  
    ...  
    break;  
}
```



반복문탈출

# 반복문 - break문

- break를 통한 while문 탈출

```
#include<stdio.h>

int main(void){
    int i, number;
    scanf("%d", &number);

    i = 0;
    while(1)
    {
        if (i >= number)
            break;

        printf("*");
        i++;
    }

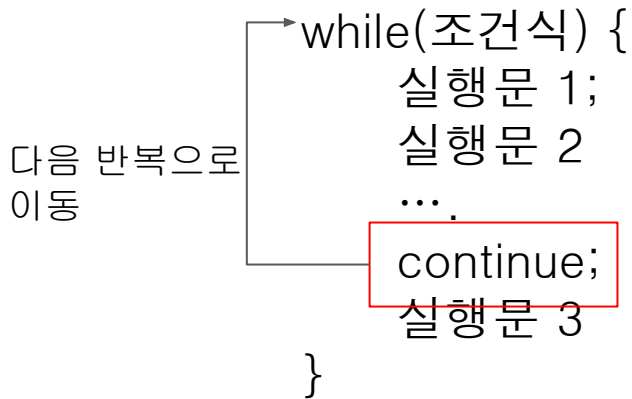
    printf("\ni : %d", i);

    return 0;
}
```

```
/* 코드가 실행되는 중입니다... */
10
*****
i : 10
/* 코드 실행이 완료되었습니다! */
```

# 반복문 - continue문

- continue문
  - 다음 반복으로 이동하기 위함, 조건을 다시 판단
  - for, while문 안에서 사용 가능
    - for문의 경우 증감식 수행







# 실습 & 과제

# 과제 / 실습

- 과제 내용
  - 당일 진행한 실습과 관련된 문제
- 제출 기한
  - 실습 : 당일 자정까지, 다음날 자정까지 지각제출 허용 (단, 점수 -30%)
  - 과제 : 다음 실습 수업 전날 23:59 까지
- 제출 방법
  - Elice의 Submit 기능 활용