

# 프로그래밍 1

1, 2주차(16.03.15) 이론

- C의 장점 : - 간결하다  
- 효율적이다  
- 저수준과 고수준이 모두 가능하다  
- 이식성이 뛰어나다

■ 메모리를 구성하는 장치 // 프로그램이 수행되기 위해서는 메인 메모리에 저장된다.

- CPU
- RAM
- 주기억장치

\* PC(Program Counter) : 메모리에 있는 명령어 중 다음번에 수행할 명령어를 가르키는 지시자

\* 이진수 예제

ex) 10진법 : 55

2	55	
2	27	1
2	13	1
2	6	1
2	3	0
2	1	1
	0	1

0011 0111  
16진법 : 37

\* 컴파일러(Compiler) : 프로그래밍 언어를 기계어로 바꾸어주는 것 (번역하는 사람)

\* 임베디드 시스템 : 특수 목적으로 만들어진 시스템

\* 팩토리얼 공식 :  $n!$  :  $n*(n+1)/2$

■ 알고리즘 : 문제를 해결하는 절차(방법)

- 자연어
- 순서도
- 의사코드

\* test.c(프로그래밍 파일) -> test.obj(오브젝트파일) -> test.exe(실행가능)

○ 소스 프로그램 : 알고리즘을 프로그래밍 언어의 문법에 맞추어 기술한 것

★ IDE(통합 개발 환경) : 코딩, 컴파일, 디버깅 등 프로그램 명령에 필요한 모든 것을 통합한 것(비주얼 스튜디오)

- 오브젝트 파일이름 : test.obj / (에디터, 컴파일러, 디버거를 통합)

\* 컴파일 오류 : 문법 오류 / ex) He go to school (3인칭 문법이 틀림)

○ 링크 : 컴파일된 목적파일과 라이브러리를 연결하여 실행 프로그램을 작성(컴파일러 안에 포함)

- 링크를 수행하는 프로그램 : 링커(linker)[통합 개발 환경에 포함되어 있음]

-

○ 실행 및 디버깅

- 실행 시간 오류 : 0으로 나누는 것 / 잘못된 메모리 주소에 접근한 것
- 논리 오류 : 문법은 틀리지 않았으나 논리적으로 정확하지 않음 (제일 힘든 오류)

- 솔루션 : 문제 해결에 필요한 프로젝트가 들어있는 컨테이너
- 프로젝트 : 하나의 실행 파일을 만드는데 필요한 여러 가지 항목들이 들어 있는 컨테이너

\* 리소스 파일 : 이미지나 동영상 등을 연결하는 파일

- \* 변수 : 변하는 값(데이터 저장 공간) / 상수 : 변하지 않는 값(이름x 상수 : 리터널(literal) 이름o : 기호상수)
- \* # : 전처리기 / 함수 : include(포함)되는 것들 (scanf, printf 등)

	내용		내용
#	전처리기	변수 / 상수	변하는 값 int x; / 변하지 않는 값 x=100;
include	포함한다	함수(function) / 함수 호출	기능 ex) int b; / 함수 안의 문장 실행
<stdio.h>	스탠다드인아웃.헤더	알고리즘	데이터를 처리하는 방법
<limits.h>	최소값과 최대값사용가능 _MAX / _MIN	식별자	변수나 함수의 이름
int	함수의 출력(데이터)타입 (결과) / 자료형	%d / %u / %f / %c	정수 / unsigned / 실수 / 문자
main	함수의 이름 / main함수는 운영체제가 호출	return 0;	자기를 호출한 함수로 돌려줌 (운영체제로 0을 돌려줌)
()	입력		

○ printf 제어문자

제어 문자 이름	제어 문자 표기	값	의미
널문자	\0	0	문자열의 끝을 표시
경고(bell)	\a	7	"뽀"하는 경고 벨소리 발생
백스페이스(backspace)	\b	8	커서를 현재의 위치에서 한 글자 뒤로 옮긴다.
수평탭(horizontal tab)	\t	9	커서의 위치를 현재 라인에서 설정된 다음 탭 위치로 옮긴다.
줄바꿈(newline)	\n	10	커서를 다음 라인의 시작 위치로 옮긴다.
수직탭(vertical tab)	\v	11	설정되어 있는 다음 수직 탭 위치로 커서를 이동
폼피드(form feed)	\f	12	주로 프린터에서 강제적으로 다음 페이지로 넘길 때 사용된다.
캐리지리턴(carriage return)	\r	13	커서를 현재 라인의 시작 위치로 옮긴다.
큰따옴표	\"	34	원래의 큰따옴표 자체
작은따옴표	\'	39	원래의 작은따옴표 자체
역슬래시(back slash)	\\	92	원래의 역슬래시 자체

- \* 오버플로어 : 데이터의 크기가 넣는 값의 크기보다 크다. (값보다 상자가 작다)
- \* 언더플로어 : 데이터의 크기가 넣은 값의 크기보다 작다. (값보다 상자가 너무 크다)
  - int(-2147483647~2147483647)에 -2147483649를 넣으면 언더 플로어
- \* unsigned : 음수가 아닌 값만 나타냄(부호가 없음)
- \* 기호 상수
  - 함수 밖 : #define EXCHANGE\_RATE 1120 // define으로 상수의 값 1120 을 포함시킨다.
  - 함수 안 : const int EXCHANGE\_RATE = 1120; // 해당 함수 안에서만 사용

- \* %d로 정수 표현 시 소수점 이하 자리는 없어짐.
- \* **지수적으로 출력하려면 %e** // ex) 5.000e+003 // 10에 3승
- \* %30.25f // 총 30자리, 소수점 25번째 자리, 정수 5번째 자리까지 출력
- \* %5d // 300일 경우 ' 300' 으로 출력

	표현		표현
%d	정수(10진수)	%c	문자
%f	실수 (6째 자리)	%lf	double 실수
%x	16진수	%o	8진수
%e	지수(10의 제곱)		

/\* 4주차 중간점검 : 빛 도달 시간 초->분/초 로 바꾸기 \*/  
#include <stdio.h>

```
void main()
{
    double lights = 3e5;
    double dis = 1496e5;
    double time, num, b;
    int a;

    time = dis / lights;
    num = time / 60.0;
    a = (int)num;
    b = num - (int)num;

    printf("빛의 속도는 %.2fkm/s\n", lights);
    printf("태양과 지구와의 거리 %.2fkm\n", dis);
    printf("도달 시간은 %d분%.2f초\n", a, b);
    printf("도달 시간은 %f초\n", time);
}
```

## 7주차(16.04.19) 실습

## ○ 연산자 기능

연산자의 분류	연산자	의미
대입	=	오른쪽을 왼쪽에 대입
산술	+ - * / %	사칙연산과 나머지 연산
부호	+	
증감	++ --	증가, 감소 연산
관계	> < == != >= <=	오른쪽과 왼쪽을 비교
논리	&&    !	논리적인 AND, OR
조건	?	조건에 따라 선택
coma	,	피연산자들을 순차적으로 실행
비트 단위 연산자	&   ^ ~ << >>	비트별 AND, OR, XOR, 이동, 반전
sizeof 연산자	sizeof	자료형이나 변수의 크기를 바이트 단위로 반환
형변환	(type)	변수나 상수의 자료형을 변환
포인터 연산자	* & []	주소계산, 포인터가 가리키는 곳의 내용 추출
구조체 연산자	. ->	구조체의 멤버 참조

- \* C에서 쓰이는 연산자를 출력하려면 2번 써야함 // ex) %d %% %d => 3 % 3

- \* 나머지 연산 사용 예 // 짝수와 홀수 구분, 5의 배수 판단

```
/* 초 입력받고 초->분/초 로 바꾸기 */
#include <stdio.h>
#define MINUTE 60

void main()
{
    int a; //실수에는 나머지 연산을 적용할 수 없다.

    printf("초 단위 시간을 입력하시오 : ");
    scanf("%d", &a);

    printf("%d초는 %d분 %d초 입니다.\n", a, a/MINUTE, a%MINUTE);
}
```

- \* ++x 와 x++의 차이

- y=++x: 는 y에 증가된 x값(11) / y=x++; 는 y에 증가되지 않은 x값(10)
- nextx = --x + y++; // nextx = 1, nexty = y-- + x++; // nexty = 2
- nextx = y++ + --x + ++y; //x=1, y=1일 경우 nextx = 4, x = 0, y = 3 / 1 + 0 + 3

/\* 두 정수를 입력받아 십진수와 16진수로 출력하는 결과 \*/

#include <stdio.h>

void main()

{

int a, b;

printf("정수를 입력하시오 : ");

scanf("%d", &a);

printf("정수를 입력하시오 : ");

scanf("%d", &b);

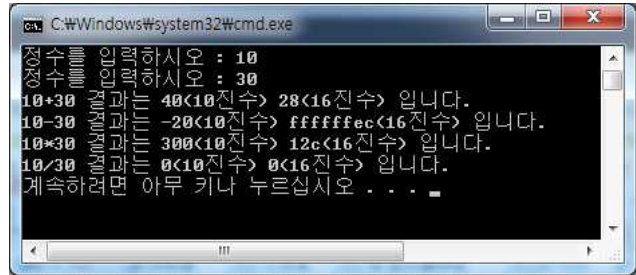
printf("%d+%d 결과는 %d(10진수) %x(16진수) 입니다.\n", a, b, a+b, a+b);

printf("%d-%d 결과는 %d(10진수) %x(16진수) 입니다.\n", a, b, a-b, a-b);

printf("%d\*%d 결과는 %d(10진수) %x(16진수) 입니다.\n", a, b, a\*b, a\*b);

printf("%d/%d 결과는 %d(10진수) %x(16진수) 입니다.\n", a, b, a/b, a/b);

}



/\* 두 실수를 입력받아 십진수와 16진수로 출력하는 실행화면 실수 10, 30 \*/

#include <stdio.h>

void main()

{

double a, b;

printf("두 개의 실수를 입력하시오 : ");

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

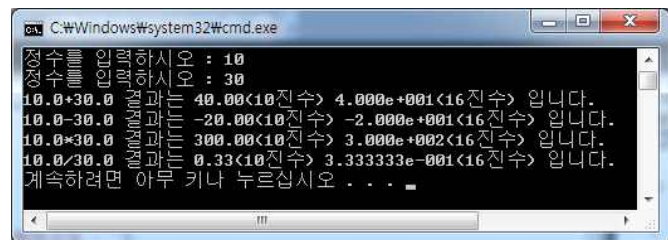
printf("%.1lf+%.1lf 결과는 %.2lf(10진수) %.3e(지수적표현) 입니다.\n", a, b, a+b, a+b);

printf("%.1lf-%.1lf 결과는 %.2lf(10진수) %.3e(지수적표현) 입니다.\n", a, b, a-b, a-b);

printf("%.1lf\*%.1lf 결과는 %.2lf(10진수) %.3e(지수적표현) 입니다.\n", a, b, a\*b, a\*b);

printf("%.1lf/%.1lf 결과는 %.2lf(10진수) %e(지수적표현) 입니다.\n", a, b, a/b, a/b);

}



## ■ 1주차 중간점검 (시험에 나올 수 있음)

- 프로그램 개발 과정을 순서대로 정리해보자  
: **요구사항분석 - 알고리즘의개발 - 코딩 - 컴파일과 링크 - 실행과 디버깅 - 유지보수**
- 소스 파일의 이름으로 test.txt는 올바른가? : txt는 사용할 수 없음.
- 소스 파일, 오브젝트 파일, 실행 파일의 차이점을 설명하라.  
: - **소스 파일은 내가 소스를 작성한 것**  
- **오브젝트 파일은 컴파일 시키며 라이브러리에 링크 시킨 것**  
- **실행파일은 링크를 완료하여 실행할 수 있게 만든 것**
- 소스 파일이 test.c라면 컴파일 과정을 거친 후에 생성되는 오브젝트 파일과 실행 파일의 이름은 어떻게 되는가?  
: 오브젝트 파일 test.obj / 실행 파일 tset.exe
- 컴파일과 링크 과정을 거쳐서 실행 파일을 만든 다음에 소스 파일과 오브젝트 파일을 보관해야 하는가?  
: **소스 파일은 당연히 보관해야함 / 오브젝트 파일은 보관 안해도됨.**
- 디버깅이란 무엇인가? : 버그를 잡는 것

## ■ 2주차 중간점검 (시험에 나올 수 있음)

- 에디터, 컴파일러, 링커, 실행, 디버깅 등의 기능이 하나의 프로그램 안에 들어 있는 것을 무엇이라하는가?  
: **IDE(통합 개발 환경)**
- Visual C++에서 새로운 프로젝트를 생성하는 메뉴는 무엇인가?  
: **파일 - 새로만들기(빈프로젝트)**
- Visual C++에서 프로젝트에 속하는 소스 파일을 컴파일하여 실행 파일을 생성하는 메뉴는?  
: **소스파일 우클릭 - 추가 - 새항목**
- C 언어에서는 대문자와 소문자를 구별하는가?
- Visual C++를 이용하여서 sample.c라는 소스 파일을 컴파일하였을 때 생성되는 파일들은 무엇인가?  
: **오브젝트 파일(sample.obj)**
- Visual C++를 사용하여 소스 프로그램을 편집하는 경우, 메모장 같은 다른 텍스트 에디터를 사용하여도 되는가?  
: **사용 안된다.**

■ 3주차 중간점검 (시험에 나올 수 있음)

- 주석은 `/*/*.....*/`와 같이 중첩해서 쓸 수 있는가? : 중첩해서 쓸 수 없다.
- 주석은 한 줄 이상 될 수 있는가? : 있다.
- 주석에는 어떤 내용을 쓰면 좋은가? : 필기할 내용
- 주석은 프로그램의 동작에 어떤 영향을 끼치는가? : 영향을 끼치지 않는다.

ex) 환율 구하기

```
#include <stdio.h>          // #은 전처리기 / standard input output : 표준입출력.헤더파일

int main()                  //(int: 결과) void main()을 쓰면 끝에 return0;을 쓰면 안된다
{                           //\t 은 tab만큼 띄워쓰기
    double a, b;
    int c;

    printf("한화당 환율을 입력하시오 : ");
    scanf("%lf", &a);

    printf("바꿀 한화를 입력하시오. : ");
    scanf("%d", &c);

    b = c / a;

    printf("%d당 달러는 %.2lf원입니다.\n", c, b); // %.2f는 소숫점 2자리까지 표현

    return 0; // 결과를 돌려줄 때 사용
}
```

시험 화요일 11시~12시

중간고사 시험 범위 : ~ 복합대입연산자 까지

(16. 11. 04) 실습

○ **fgets** 기능

- fgets(char배열이름, sizeof(배열이름), stdin); : \n(엔터)가 나올 때까지 문자열을 입력받는다.
- 배열이름[strlen(배열이름)-1] = '\0'; : 마지막 char배열은 공백이 들어가므로 -1에 '\0'을 넣는다.

○ **fflush** 기능

- fflush(stdin); : 버퍼를 지움 (문자열을 비움)

중간고사 파트 끝.

# 프로그래밍 1 - 1학기 기말

9주차(16.05.03)

- 형 변환
  - 명시적인 형 변환 : (int)a 식으로 작성
- 비교시 주의사항 :  $(1e32 + 0.01) > 1e32$  일 경우 같은 것으로 간주한다.

- 연산자 우선 순위 : 단항>산술>관계>논리>대입>coma
  - \* 단항 : ++ --
  - \* 산술 : + - \* / %
  - \* 관계 : > < >= <=
  - \* 논리 : && ||
  - \* 대입 : =

10주차(16.05.10)

- 제어문 - 조건문(if), 반복문(while, for)
  - \* 복합문 : if문 안의 문장이 2개 이상인 것 {} 사용
  - \* else문은 가장 가까운 if문과 매칭 됨

11주차(16.05.17)

- while 문 : 조건이 만족할 때 까지 // ex) 10kg빠질 때 까지
  - while(scanf("%d", &a) != EOF) : 입력한 숫자가 EOF(-1)가 아니면 반복문 종료 / Ctrl+z
- for 문 : 몇 번 반복하는지 지정해 줘야함. // ex) 100번 반복하라
- 제곱근 : #include <Math.h> / sqrt(스퀘어 루트)사용 (double형)
- 정수형중의 최대값 : INT\_MAX / 최소값 : INT\_MIN;

12주차(16.05.24)

- rand() : #include <stdlib> / rand() % num 사용
  - rand() % 101 // 0~100 까지 랜덤 값
  - rand() % 100 + 1 // 1~100까지 랜덤 값
- time : #include <time.h>
  - srand((unsigned int)time(NULL));
- srand((unsigned int)time(NULL)); // srand 는 rand의 seed값 ()
- 무한루프 : while(1) / for(;;)
- continue : 반복문의 continue 밑을 스킵하고 반복문 실행

13주차(16.05.31)

- 함수종류 : 사용자 정의 함수 / 라이브러리 함수(print)
  - 반환형 / 함수 헤더 / 함수 몸체 //ex) int add(int a){} 이면 return이 int형인 add() 함수
- 함수호출 : 반환형 함수명(); //ex) void add(){ } 이면 add();
- 함수반환 : return;
  - \* 매개 변수 : 함수 안에 입력받는 값
- 함수 원형 선언 : 함수가 메인함수 뒤에 있을 경우 선언 // ex)double get\_double();

## ○ 9주차 실습

//화씨온도 -> 섭씨온도로 바꾸기//

#include <stdio.h>

void main()

{

double a;

int b;

printf("화씨 입력 :");

scanf("%d", &b);

a = 5.0 / 9 \* (b-32);

printf("섭씨온도 %d, 화씨온도 %.2lf",b, a);

}

//아스키코드 값 구하기//

#include <stdio.h>

void main()

{

char c;

printf("문자 숫자를 입력하시오<0-9> : ");

scanf("%c", &c);

printf("읽어들인 문자 %c의 아스키 코드 값은 %d 입니다.\n", c, c);

printf("문자 %c 를 숫자 %d 로 변환하였습니다.\n", c, (c-'0'));

}

//과제1. ISBN 체크 숫자 구하기//

#include <stdio.h>

void main()

{

int a[9];

int x=0, y;

printf("ISBN의 처음 아홉자리 숫자를 연속적으로 입력하시오\n");

scanf\_s("%d %d %d %d %d %d %d %d %d", &a[0], &a[1], &a[2], &a[3], &a[4], &a[5], &a[6], &a[7], &a[8]);

for (int i = 0; i < 9; i++)

{

x += a[i] \* (i+1);

}

y = x % 11;

printf("입력된 ISBN의 체크 숫자는 %d입니다.\n", y);

}

## ○ 10주차 실습

```
//윤년 구하기//
#include <stdio.h>

void main()
{
    int year;

    printf("연도를 입력 : ");
    scanf("%d", &year);

    if ((year%4==0) && (year%100 != 0) || (year%400 ==0))
        printf("윤년임 \n");
    else printf("윤년 아님 \n");
}

//BMI 구하기//
#include <stdio.h>
void main()
{
    int kg, cm;
    double BMI;

    printf("키와 몸무게를 입력하시오 : ");
    scanf("%d %d", &cm, &kg);

    BMI = kg / ((double)cm*cm0.0001);

    if(BMI <= 18.5)
        printf("저체중입니다. 건강에 유의하세요.\n");
    else if(BMI <= 24.9)
        printf("정상체중입니다. 현재 상태를 유지하십시오.");
    else if(BMI <= 29.9)
        printf("과체중입니다. 적절한 운동을 하십시오.");
    else
        printf("비만입니다. 다이어트와 적절한 운동을 하십시오.");
}
```

## ○ 11주차 실습

```
//방사능 남은양 구하기//
#include <stdio.h>

void main()
{
    int sum=0, count=0, grade=0;
    double average;
    printf("입력 종료 시 음수 값 입력\n");
    while(grade >=0){
        sum = sum+grade;
        scanf("%d", &grade);
        count++;
    }
    average = (double)sum / (count-1);
    printf("평균은 = %lf\n", average);
}

//방사능 남은양 구하기//
#include <stdio.h>

void main()
{
    int year;
    double bangsa=100.0;
    double c=bangsa;
    printf("반감기를 입력하시오(년):");
    scanf("%d", &year);

    while(1)
    {
        bangsa = bangsa/2;
        printf("%d년 후에 남은 양=%lf\n", year, bangsa);
        if (bangsa <= c/10)
        {
            printf("1/10이하로 되기 까지 걸린 시간 = %d\n", year);
            break;
        }
        else
            year+=10;
    }
}
```



## ○ 12주차 실습

//랜드값을 써서 숫자 맞추기//

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
```

```
void main(){

    int answer, input, count=0;

    srand((unsigned int)time(NULL));

    answer = rand() % 101;
    printf("answer = %d\n", answer);
    do
    {
        printf("숫자 입력 : ");
        scanf("%d", &input);
        if(input < answer )
            printf("더 큰 숫자 입력 \n");
        else if(input > answer )
            printf("더 작은 숫자 입력 \n");
        count++;
    }
    while( answer != input );

    printf("정답, %d 회 만에 정답\n", count);
}
```

//\*을 계단식으로 출력(공백 추가)//

```
#include <stdio.h>
```

```
void main(){

    int num;

    printf("숫자 입력 : ");
    scanf("%d", &num);

    for(int i=1; i<=num; i++)
    {
        for(int k=num-i; k>0; k--)
        {
            printf(" ");
        }
        for(int j=0; j<i; j++)
        {
            printf("*");
        }
        printf("\n");
    }
}
```

//2,3,5의 배수가 아닌 수 출력//

```
#include <stdio.h>
```

```
void main()
{
    int num;

    while(1)
    {
        printf("숫자 입력 : ");
        scanf("%d", &num);
        if( num < 0)
        {
            break; //음수 입력시 종료
        }

        for(int i=1; i<=num; i++)
        {
            if(i%2!=0 && i%3!=0 && i%5!=0)
            {
                printf(" %d ", i);
            }
        }
        printf("\n");
    }
}
```

## ○ 12주차 과제

/\* 종이를 한번 접으면 면적이 1/2로 줄어든다. 종이를 몇 번 접어야 원래 면적의 1/100로 줄어드는가?  
역시 로그 함수나 지수 함수를 사용하지 말고 반복 구조를 이용하여서 해결하여 보자. \*/

```
#include <stdio.h>
```

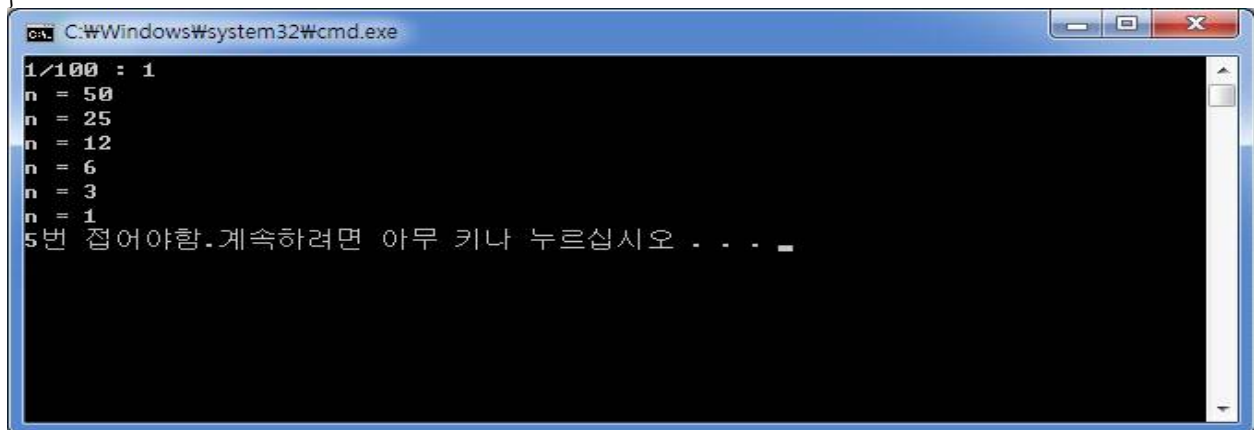
```
void main()
{
    int count=0; //몇번 접었는지 카운트
    int num=100; // 입력받는 면적
    int n; // 줄어들 숫자

    printf("1/100 : %d\n", num/100);

    n = num;

    while(1)
    {
        n = n/2;
        printf("n = %d \n", n);
        if( n <= num/100)
            break;
        count++;
    }

    printf("%d번 접어야함.", count);
}
```



```
C:\Windows\system32\cmd.exe
1/100 : 1
n = 50
n = 25
n = 12
n = 6
n = 3
n = 1
5번 접어야함. 계속하려면 아무 키나 누르십시오 . . .
```

/\*세균이 1시간마다 4배씩 증가한다고 가정하자. 이 세균 10마리를 배양하면 7시간 후의 세균의 수는 얼마나 될까?  
역시 지수 함수나 로그 함수를 이용하지 말고 반복 구조만을 사용하여서 해결하여 보자. \*/

```
#include <stdio.h>
```

```
void main()
{
    int virus;
    while (1)
    {
        printf_s("세균 개체 수 : ");
        scanf_s("%d", &virus);
        if (virus < 0) break;

        for (int i = 1; i <= 7; i++)
        {
            virus *= 4;
        }

        printf("7시간 후의 세균의 수 : %d\n", virus);
    }
}
```



```
C:\WINDOWS\system32\cmd.exe
세균 개체 수 : 50
7시간 후의 세균의 수 : 819200
세균 개체 수 : 70
7시간 후의 세균의 수 : 1146880
세균 개체 수 : 100
7시간 후의 세균의 수 : 1638400
세균 개체 수 : -1
계속하려면 아무 키나 누르십시오 . . .
```

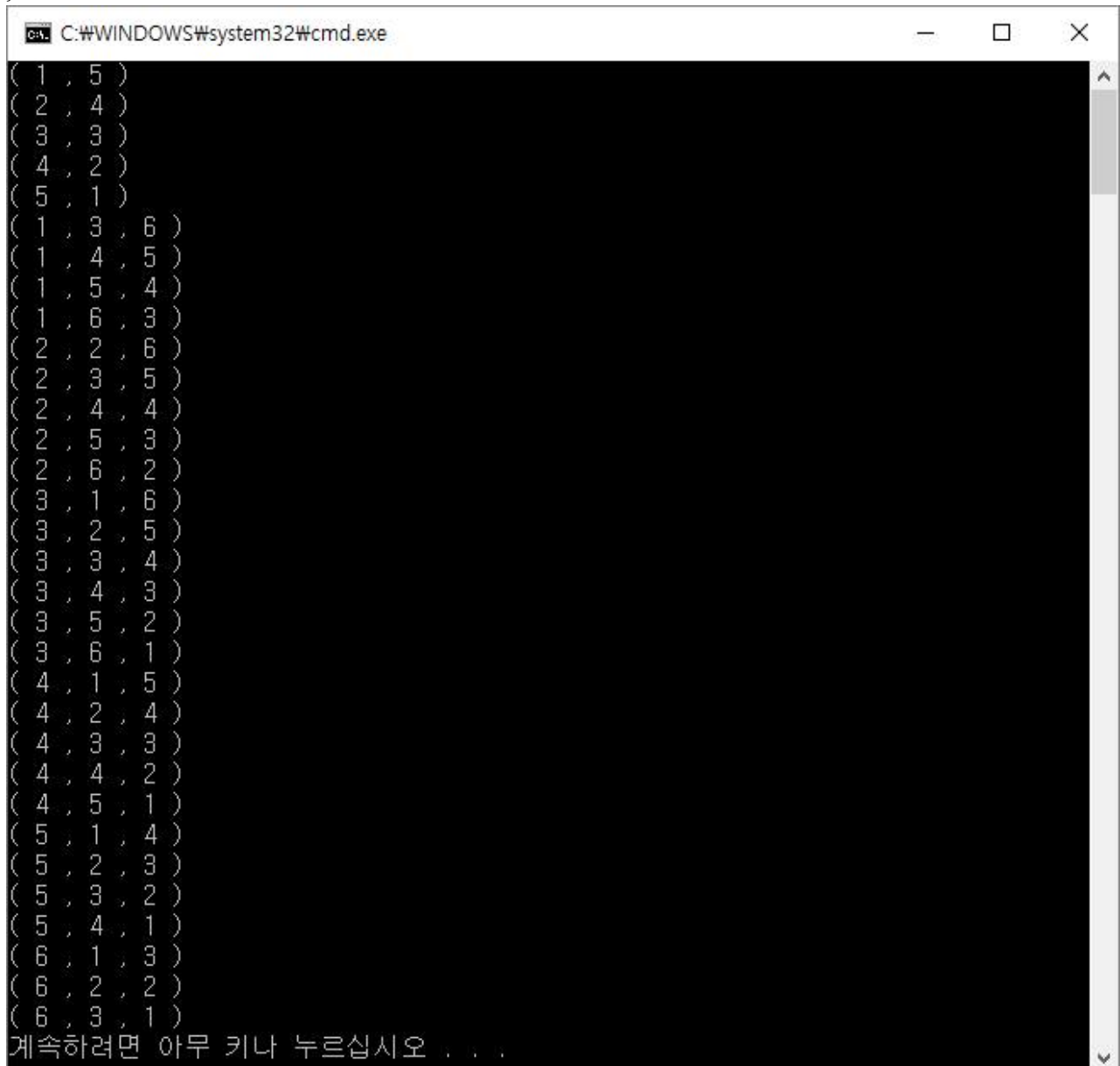
## ○ 12주차 과제

/\*라스베가스와 같은 도박장에 가면 주사위 게임이 있다.

주사위 2개를 던졌을 때, 합이 6이 되는 경우를 전부 출력하여 보자. 예를 들어서 (1,5),(2,4)...와 같이 출력되면 된다.  
또 주사위 3개를 사용하여 합이 10이 되는 경우를 전부 출력하여 보자. \*/

```
#include <stdio.h>
```

```
void main(){
    for (int i = 1; i <= 6; i++)
    {
        for (int j = 1; j <= 6; j++){
            if (i + j == 6){
                printf("( %d , %d ) \n", i, j);
            }
        }
    }
    for (int i = 1; i <= 6; i++)
    {
        for (int j = 1; j <= 6; j++){
            for (int k = 1; k <= 6; k++){
                if (i + j + k == 10){
                    printf("( %d , %d , %d ) \n", i, j, k);
                }
            }
        }
    }
}
```



```
C:\WINDOWS\system32\cmd.exe
( 1 , 5 )
( 2 , 4 )
( 3 , 3 )
( 4 , 2 )
( 5 , 1 )
( 1 , 3 , 6 )
( 1 , 4 , 5 )
( 1 , 5 , 4 )
( 1 , 6 , 3 )
( 2 , 2 , 6 )
( 2 , 3 , 5 )
( 2 , 4 , 4 )
( 2 , 5 , 3 )
( 2 , 6 , 2 )
( 3 , 1 , 6 )
( 3 , 2 , 5 )
( 3 , 3 , 4 )
( 3 , 4 , 3 )
( 3 , 5 , 2 )
( 3 , 6 , 1 )
( 4 , 1 , 5 )
( 4 , 2 , 4 )
( 4 , 3 , 3 )
( 4 , 4 , 2 )
( 4 , 5 , 1 )
( 5 , 1 , 4 )
( 5 , 2 , 3 )
( 5 , 3 , 2 )
( 5 , 4 , 1 )
( 6 , 1 , 3 )
( 6 , 2 , 2 )
( 6 , 3 , 1 )
계속하려면 아무 키나 누르십시오 . . .
```

## ○ 13주차 실습

/\* 함수 실습 : 함수호출 및 함수 원형 선언 \*/

#include <stdio.h>

double get\_double(); // 함수 원형 선언

double add(double x, double y);

double get\_max(double x, double y);

void print\_star();

void main(){

double num1, num2;

num1 = get\_double();

num2 = get\_double();

print\_star();

printf("두 실수의 합 : %.2f \n", add(num1, num2));

printf("두 실수 중 큰 값 : %.2f \n", get\_max(num1, num2));

print\_star();

}

double add( double x, double y){

return x+y;

}

double get\_double(){

double num;

printf("실수 두개 입력 : ");

scanf("%lf", &num);

return num;

}

double get\_max(double x, double y){

if(x>y) return x;

else return y;

}

void print\_star(){

int i;

for(i=0; i<30; i++){

printf("\*");

printf("\n");

}

```

C:\Windows\system32\cmd.exe
실수 두개 입력 : 5.5
실수 두개 입력 : 6.3
*****
두 실수의 합 : 11.80
두 실수 중 큰 값 : 6.30
*****
계속하려면 아무 키나 누르십시오 . . .
  
```

/\* 소수구하기 \*/

#include <stdio.h>

int get\_num();

int is\_prime(int num);

int prime(int num);

void main(){

int num = get\_num();

is\_prime(num);

printf("\n");

prime(num);

}

int get\_num(){

int num;

printf("숫자 입력 : ");

scanf("%d", &num );

return num;

}

int is\_prime(int num){

int count=0;

int i, j;

for(i=1; i<=num; i++){

for(j=1; j<i; j++){

if(i%j == 0) count++;

if(count ==1) printf("[%d] ", i);

count =0;

}

}

int prime(int num){

int i, count=0;

for( i=2; i<num; i++){

if (num % i ==0) count++;

}

if (count==0) printf("%d은(는) 소수입니다.\n", num);

else printf("%d은(는) 소수가 아닙니다.\n", num);

}

```

C:\Windows\system32\cmd.exe
숫자 입력 : 50
[2] [3] [5] [7] [11] [13] [17] [19] [23] [29] [31] [37] [41] [43] [47]
50은(는) 소수가 아닙니다.
계속하려면 아무 키나 누르십시오 . . .
  
```

```

/* 기말고사
2문제
1. 조건문, 반복문 총2개 (개당 10점 / 총 20점)
2. 함수 문제 총4개(개당 10점 / 총 40점)
*/
#include <stdio.h>

int get_integer();
int get_sum(int n);
void print_pattern(int n);
int is_prime(int n);

void main(){
    int n;
    n=get_integer();
    printf("1에서 %d까지의 합계는 %d입니다. \n", n, get_sum(n));
    print_pattern(n);
    if(is_prime(n))
        printf("%d은(는) 소수입니다.\n", n);
    else
        printf("%d은(는) 소수가 아닙니다. \n", n);
}

int get_integer(){
    int num=0;
    printf("정수를 입력하시오 : ");
    scanf_s("%d", &num);
    return num;
}

int get_sum(int n){
    int sum=0;
    for(int i=1; i<=n; i++){
        sum+=i;
    }
    return sum;
}

void print_pattern(int n){
    for(int i=0; i<=n; i++){
        for(int j=0; j<=i; j++){
            printf("*");
        }
        printf("\n");
    }
}

int is_prime(int n){
    int count=0;
    int num2=10;

    for(int i=1; i<=n; i++){
        if(n%i==0){
            count++;
        }
    }

    if(count == 2){
        return true;
    }
    else
        return false;
}

/*
void main(){
    int sum=0;
    int sum2=0;

    for(int i=1; i<101; i++){
        if(i%2!=0){
            sum+=i;
        }
        if(i%3!=0){
            sum2+=i;
        }
    }

    printf("1에서 100까지 홀수의 합 : %d\n",sum);
    printf("1에서 100까지 3의 배수만 제외한 정수의 합 : %d\n", sum2);
}
*/

```

기말고사 파트 끝.

=====