



---

# 7 장~10 장 실습예제 과제

---

C 프로그래밍



학번: 20200814

이름: 임유빈

학과: 컴퓨터정보공학과

github: <https://github.com/MrLim2589/2020-2-C>

# CHAPTER

07

반복

## 예제 코드

```
// file: repetition.c
#include <stdio.h>

int main(void)
{
    //모두 동일한 문자열의 단순한 반복
    printf("C 언어 재미있네요!\n");
    printf("C 언어 재미있네요!\n");
    printf("C 언어 재미있네요!\n");

    int n = 1;
    //정수값을 1씩 증가시키면서 출력 반복
    printf("%d\n", n++);
    printf("%d\n", n++);
    printf("%d\n", n++);
    printf("%d\n", n++);
    printf("%d\n", n++);

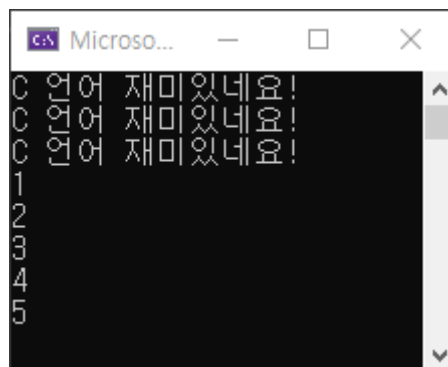
    return 0;
}
```

실습예제 7-1

파일 이름: `repetition.c`

교재: P298~P299

## 결과



```
C:\> Microsoft...
C 언어 재미있네요!
C 언어 재미있네요!
C 언어 재미있네요!
1
2
3
4
5
```

## 예제 코드

```
// file: cel2far3.c
#include <stdio.h>

int main(void)
{
    double celcius = 12.46;

    printf("    섭씨(C)    화씨(F)\n");
    printf("-----Wn");
    printf("%8.2lf %8.2lfWn", celcius, 9.0 / 5 * celcius + 32);
    celcius += 10;
    printf("%8.2lf %8.2lfWn", celcius, 9.0 / 5 * celcius + 32);
    celcius += 10;
    printf("%8.2lf %8.2lfWn", celcius, 9.0 / 5 * celcius + 32);
    celcius += 10;

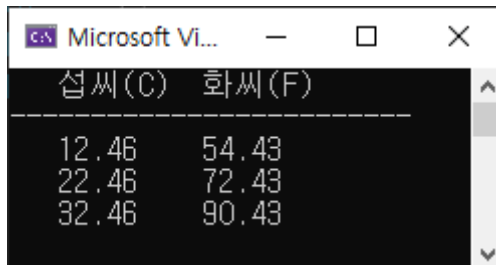
    return 0;
}
```

실습예제 7-2

파일 이름: cel2far3.c

교재: P300

## 결과



```
Microsoft Vi...
섭씨(C) 화씨(F)
-----
12.46 54.43
22.46 72.43
32.46 90.43
```

## 예제 코드

```
// file: onebyte.c
#include <stdio.h>
```

```
int main(void)
{
```

```
    int num = 13;
    printf("정수 %d의 오른쪽 8비트 내부값\n", num);
```

```
    printf("%d", num >> 7 & 1); //오른쪽 8번째 비트값 출력
    printf("%d", num >> 6 & 1);
    printf("%d", num >> 5 & 1);
    printf("%d", num >> 4 & 1);
    printf(" %d", num >> 3 & 1);
    printf("%d", num >> 2 & 1);
    printf("%d", num >> 1 & 1);
    printf("%d\n", num >> 0 & 1); //오른쪽 첫 비트값 출력
```

```
    return 0;
```

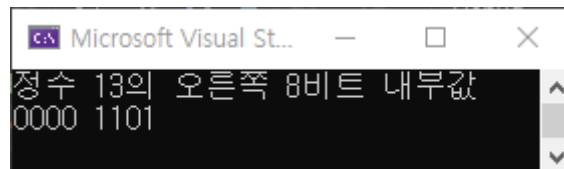
```
}
```

실습예제 7-3

파일 이름: onebyte.c

교재: P301

## 결과



## 예제 코드

```
// file: whilebasic.c
#include <stdio.h>

int main(void)
{
    int count = 1;

    while (count <= 3)
    {
        printf("C 언어 재미있네요!\n");
        count++;
    }
    printf("제어변수 count => %d\n", count);

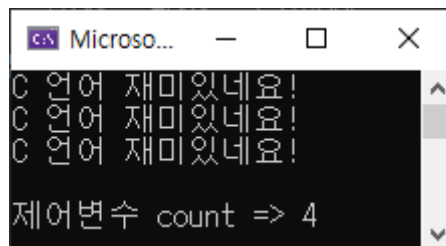
    return 0;
}
```

실습예제 7-4

파일 이름: **whilebasic.c**

교재: P304~P305

## 결과



## 예제 코드

```
// file: whilenumber.c

#include <stdio.h>
#define MAX 5

int main(void)
{
    int n = 1;

    while (n <= MAX)
        printf("%d\n", n++);

    printf("\n제어변수 count => %d\n", n);

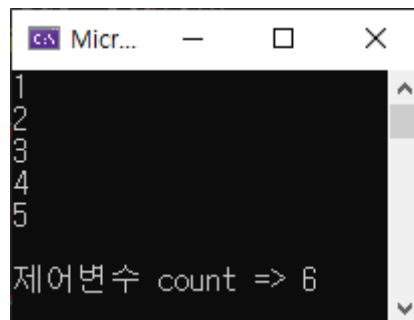
    return 0;
}
```

실습예제 7-5

파일 이름: whilenumber.c

교재 : P306

## 결과



```
c:\n Micr...
1
2
3
4
5
제어변수 count => 6
```

## 예제 코드

```
// fil: whilelab.c

#include <stdio.h>
#define MAX 20

int main(void)
{
    int n = 0;

    while (n < MAX) {
        printf("%4d", n);
        n += 3;
    }
    puts("");

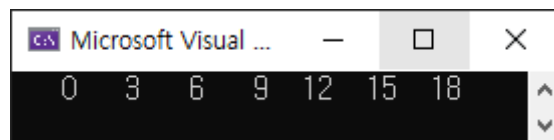
    return 0;
}
```

LAB 7-1

파일 이름: whilelab.c

교재: P307

## 결과





## 예제 코드

```
// file: dowhile.c
#define _CRT_SECURE_NO_WARNINGS //scanf 오류 방지

#include <stdio.h>

int main(void)
{
    int input;

    do
    {
        printf("정수 또는 0(종료)을 입력: ");
        scanf("%d", &input);
    } while (input != 0); //while (intput);

    puts("종료합니다.");

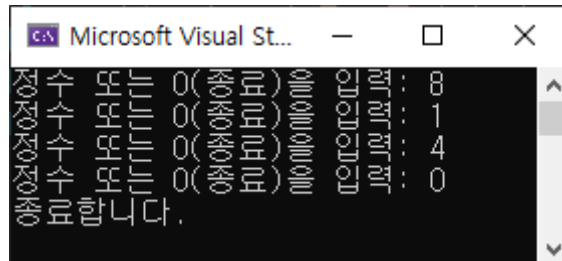
    return 0;
}
```

실습예제 7-6

파일 이름: **dowhile.c**

교재: P309~P310

## 결과



```
Microsoft Visual St...
정수 또는 0(종료)을 입력: 8
정수 또는 0(종료)을 입력: 1
정수 또는 0(종료)을 입력: 4
정수 또는 0(종료)을 입력: 0
종료합니다.
```

## 예제 코드

```
// file: dowhilenumbers.c

#include <stdio.h>
#define MAX 5

int main(void)
{
    int n = 1;

    do
    {
        printf("%d\\n", n++);
    } while (n <= MAX);

    printf("\\n제어변수 n => %d\\n", n);

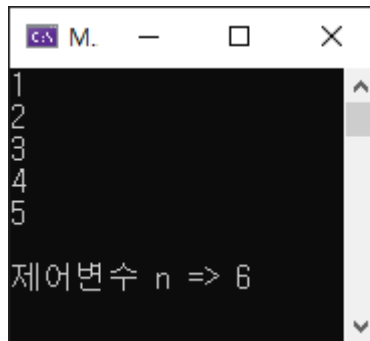
    return 0;
}
```

실습예제 7-7

파일 이름: `dowhilenumbers.c`

교재: P310~P311

## 결과



```
1
2
3
4
5
\\n제어변수 n => 6
```

## 예제 코드

```
// file: dowhilelab.c
#define _CRT_SECURE_NO_WARNINGS

#include <stdio.h>

int main(void)
{
    int input = 0, result = 0, digit = 0;
    int devider = 100;

    printf("양의 정수[100~999] 입력 : ");
    scanf("%d", &input);
    result = input;
    do
    {
        digit = result / devider;
        result %= devider;
        printf("%3d단위 출력: %d\n", devider, digit);
        devider /= 10;
    } while (devider >= 1);

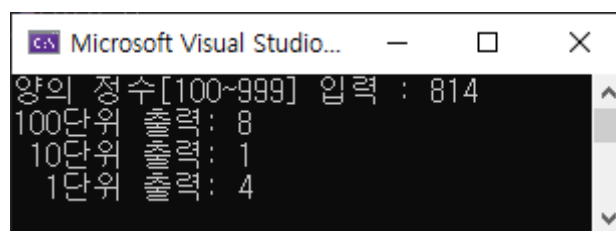
    return 0;
}
```

LAB 7-2

파일 이름: **dowhilelab.c**

교재 : P312~P313

## 결과



```
Microsoft Visual Studio...
양의 정수[100~999] 입력 : 814
100단위 출력: 8
10단위 출력: 1
1단위 출력: 4
```

## 예제 코드

```
// file: forbasic.c

#include <stdio.h>
#define MAX 5

int main(void)
{
    int i;

    for (i = 1; i <= MAX; i++)
    {
        printf("C 언어 재미있네요! %d\n", i);
    }

    printf("\n제어변수 i => %d\n", i);

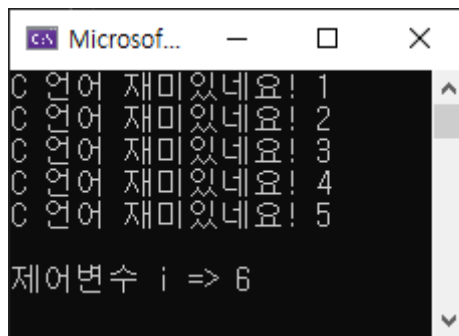
    return 0;
}
```

실습예제 7-8

파일 이름: **forbasic.c**

교재: P316

## 결과



```
c:\n Microsof...
C 언어 재미있네요! 1
C 언어 재미있네요! 2
C 언어 재미있네요! 3
C 언어 재미있네요! 4
C 언어 재미있네요! 5

제어변수 i => 6
```

## 예제 코드

```
// file: force12far3.c

#include <stdio.h>
#define MAX 3
#define INCREMENT 10

int main(void)
{
    double celcius = 12.46;

    printf(" 섭씨(C) 화씨(F)\n");
    printf("-----\n");

    for (int i = 1; i <= MAX; i++, celcius += INCREMENT)
    {
        printf("%8.2lf %8.2lf\n", celcius, 9.0 / 5 * celcius + 32);
    }

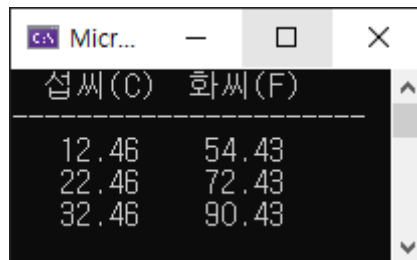
    return 0;
}
```

실습예제 7-9

파일 이름: force12far3.c

교재: P317~P318

## 결과



섭씨(C)	화씨(F)
12.46	54.43
22.46	72.43
32.46	90.43

## 예제 코드

```
// file: forbit.c
#include <stdio.h>

#define TOTAL_BIT 32

int main(void)
{
    int num = 13;
    printf("정수 %d의 %d비트 내부값:\n", num, TOTAL_BIT);

    for (int i = TOTAL_BIT - 1; i >= 0; i--)
        printf("%d", num >> i & 1);

    printf("\n");

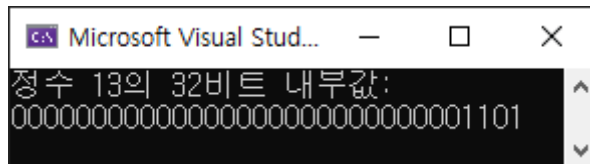
    return 0;
}
```

실습예제 7-10

파일 이름: **forbit.c**

교재: P319~P320

## 결과



## 예제 코드

```
// file: forsum.c
#include <stdio.h>

int main(void)
{
    int i, sum;

    for (i = 1, sum = 0; i <= 10; i++) //++i도 가능
        sum += i;
    printf("1에서 10까지의 합: %d\n", sum);

    for (i = 1, sum = 0; i <= 10; )
        sum += i++;
    printf("1에서 10까지의 합: %d\n", sum);

    for (i = 0, sum = 0; i <= 9; )
        sum += ++i;
    printf("1에서 10까지의 합: %d\n", sum);

    for (i = 1, sum = 0; i <= 10; sum += i++); //반복 몸체가 없는 for 문
    printf("1에서 10까지의 합: %d\n", sum);

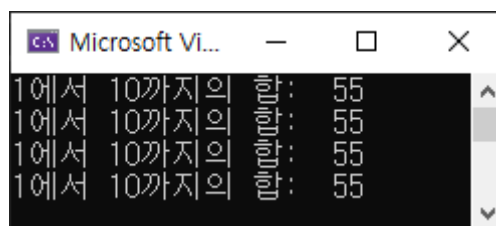
    return 0;
}
```

실습예제 7-11

파일 이름: forsum.c

교재: P321~P322

## 결과



```
Microsoft Vi...
1에서 10까지의 합: 55
1에서 10까지의 합: 55
1에서 10까지의 합: 55
1에서 10까지의 합: 55
```

## 예제 코드

```
// file: inputsum.c
#define _CRT_SECURE_NO_WARNINGS //scanf 오류 방지

#include <stdio.h>

int main(void)
{
    int i, sum, max;

    printf("1에서부터 정수까지의 합을 구할 양의 정수 하나 입력: ");
    scanf("%d", &max);

    for (i = 1, sum = 0; i <= max; i++)
        sum += i;
    printf("for 문으로 구한 1에서 %d까지 합: %3d\n", max, sum);

    i = 1, sum = 0;
    while (i <= max)
    {
        sum += i; //sum = sum + i;
        i++; // ++i도 가능
    }
    printf("while 문으로 구한 1에서 %d까지 합: %3d\n", max, sum);

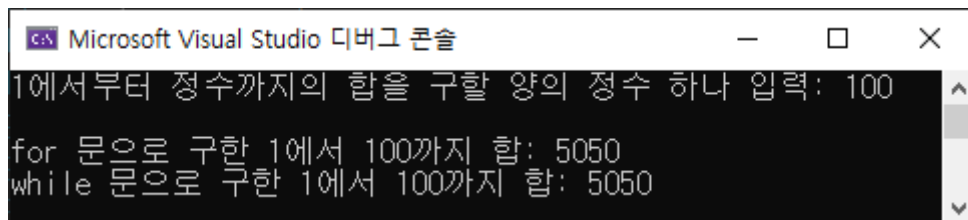
    return 0;
}
```

실습예제 7-12

파일 이름: `inputsum.c`

교재: P323~P324

## 결과



```
Microsoft Visual Studio 디버그 콘솔
1에서부터 정수까지의 합을 구할 양의 정수 하나 입력: 100
for 문으로 구한 1에서 100까지 합: 5050
while 문으로 구한 1에서 100까지 합: 5050
```



## 예제 코드

```
// file: forlab.c

#include <stdio.h>
#define MAX 9

int main(void)
{
    printf("=== 구구단 출력 ===\n");
    for (int i = 2; i <= MAX; i++)
    {
        printf("%6d단 출력\n", i);
    }

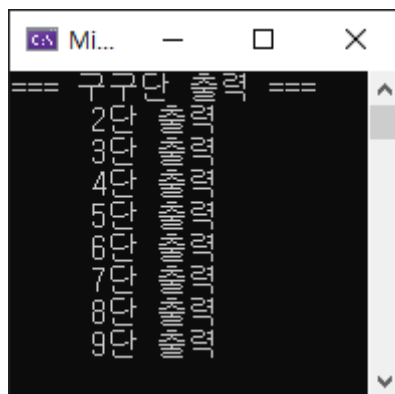
    return 0;
}
```

LAB 7-3

파일 이름: forlab.c

교재: P325

## 결과



```
=== 구구단 출력 ===
  2단 출력
  3단 출력
  4단 출력
  5단 출력
  6단 출력
  7단 출력
  8단 출력
  9단 출력
```

## 예제 코드

```
// file: break.c
#define _CRT_SECURE_NO_WARNINGS //scanf 오류 방지

#include <stdio.h>

int main(void)
{
    int input;

    while (1)
    {
        printf("정수[음수, 0(종료), 양수]를 입력 후 [Enter] : ");
        scanf("%d", &input);
        printf("입력한 정수 %d\n", input);
        if (input == 0)
            break;
    }
    puts("종료합니다.");

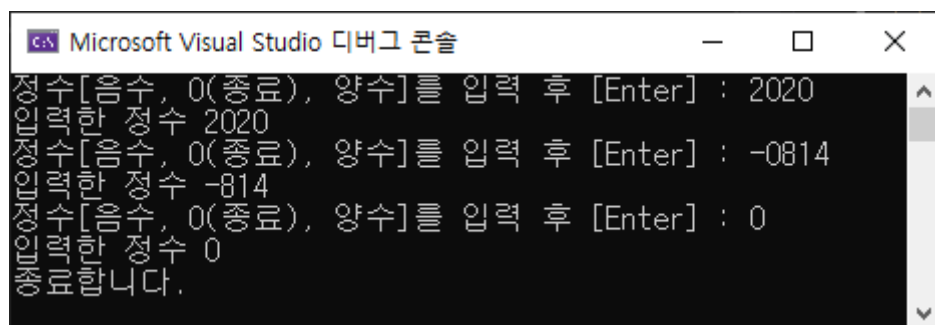
    return 0;
}
```

실습예제 7-13

파일 이름: **break.c**

교재 : P328

## 결과



```
Microsoft Visual Studio 디버그 콘솔
정수[음수, 0(종료), 양수]를 입력 후 [Enter] : 2020
입력한 정수 2020
정수[음수, 0(종료), 양수]를 입력 후 [Enter] : -0814
입력한 정수 -814
정수[음수, 0(종료), 양수]를 입력 후 [Enter] : 0
입력한 정수 0
종료합니다.
```

## 예제 코드

```
// file: continue.c

#include <stdio.h>

int main(void)
{
    const int MAX = 15;

    printf("1에서 %d까지 정수 중에서 3으로 나누어 떨어지지 않는 수\n", MAX);
    for (int i = 1; i <= MAX; i++)
    {
        if (i % 3 == 0) // (!(i % 3))
            continue;
        printf("%3d", i);
    }
    puts("");

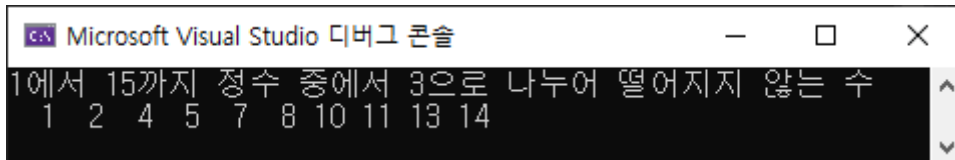
    return 0;
}
```

실습예제 7-14

파일 이름: `continue.c`

교재 : P330

## 결과



Microsoft Visual Studio 디버그 콘솔

```
1에서 15까지 정수 중에서 3으로 나누어 떨어지지 않는 수
1  2  4  5  7  8 10 11 13 14
```

## 예제 코드

```
// file: goto.c

#include <stdio.h>

int main(void)
{
    int count = 1;

loop :
    printf("%3d", count);
    if (++count <= 10)
        goto loop;

    printf("\n프로그램을 종료합니다.\n");

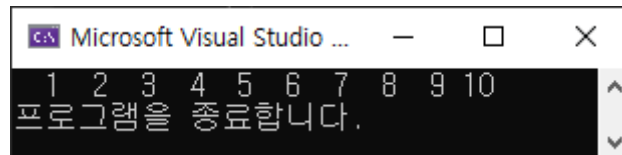
    return 0;
}
```

실습예제 7-15

파일 이름: goto.c

교재 : P331

## 결과



## 예제 코드

```
// file: menu.c
#define _CRT_SECURE_NO_WARNINGS //scanf 오류 방지

#include <stdio.h>

int main(void)
{
    int input;

    do {
        printf("Wt [1] 한식Wn");
        printf("Wt [2] 양식Wn");
        printf("Wt [3] 분식Wn");
        printf("Wt [4] 기타Wn");
        printf("메뉴 번호 선택 후 [Enter] : ");
        scanf("%d", &input);
        printf("선택 메뉴 %dWn", input);

        if (input <= 4 && input >= 1)
            break;
    } while (1);

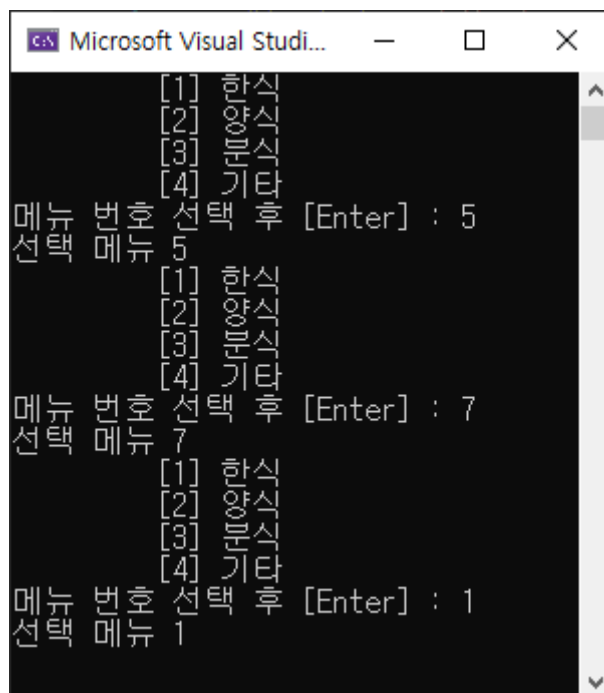
    return 0;
}
```

실습예제 7-16

파일 이름: menu.c

교재 : P332~P333

## 결과



```
Microsoft Visual Studi...
[1] 한식
[2] 양식
[3] 분식
[4] 기타
메뉴 번호 선택 후 [Enter] : 5
선택 메뉴 5
[1] 한식
[2] 양식
[3] 분식
[4] 기타
메뉴 번호 선택 후 [Enter] : 7
선택 메뉴 7
[1] 한식
[2] 양식
[3] 분식
[4] 기타
메뉴 번호 선택 후 [Enter] : 1
선택 메뉴 1
```

## 예제 코드

```
// file: continuelab.c

#include <stdio.h>

int main(void)
{
    const int MAX = 15;

    printf("1에서 %d까지 정수 중에서 5로 나누어 떨어지지 않는 수\n", MAX);
    for (int i = 1; i <= MAX; i++)
    {
        if (!(i % 5))
            continue;
        printf("%3d", i);
    }
    puts("");

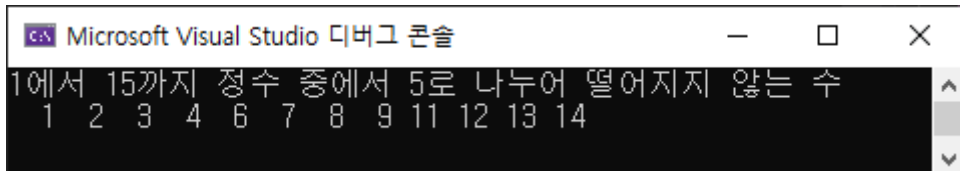
    return 0;
}
```

LAB 7-4

파일 이름: `continuelab.c`

교재 : P334

## 결과



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio 디버그 콘솔". The console output displays the text "1에서 15까지 정수 중에서 5로 나누어 떨어지지 않는 수" followed by a space-separated list of numbers: "1 2 3 4 6 7 8 9 11 12 13 14". The numbers 5 and 10 are omitted from the sequence.

## 예제 코드

```
// file: nestedloop.c

#include <stdio.h>

int main(void)
{
    int m, n;

    for (m = 1; m <= 5; m++)
    {
        printf("m = %-2d\\n", m);
        for (n = 1; n <= 7; n++)
            printf("n = %-3d", n);
        puts("");
    }

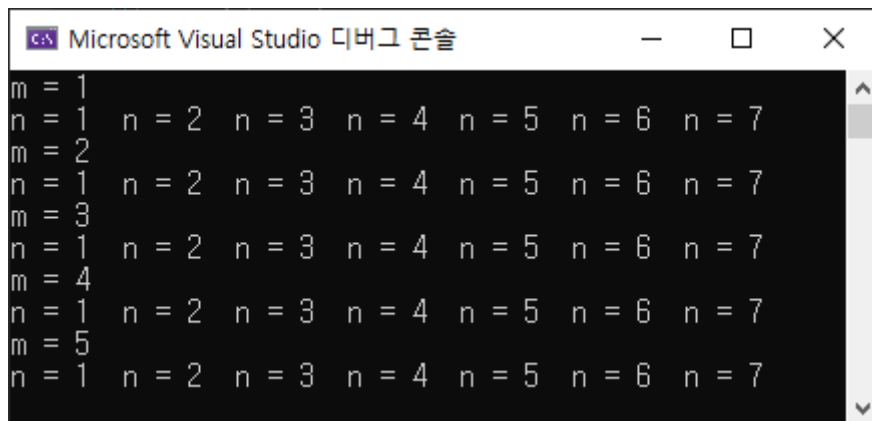
    return 0;
}
```

실습예제 7-17

파일 이름: **nestedloop.c**

교재 : P336~P337

## 결과



```
Microsoft Visual Studio 디버그 콘솔

m = 1
n = 1  n = 2  n = 3  n = 4  n = 5  n = 6  n = 7
m = 2
n = 1  n = 2  n = 3  n = 4  n = 5  n = 6  n = 7
m = 3
n = 1  n = 2  n = 3  n = 4  n = 5  n = 6  n = 7
m = 4
n = 1  n = 2  n = 3  n = 4  n = 5  n = 6  n = 7
m = 5
n = 1  n = 2  n = 3  n = 4  n = 5  n = 6  n = 7
```

## 예제 코드

```
// file: triangle.c

#include <stdio.h>

int main(void)
{
    const int MAX = 5;
    int i, j;

    for (i = 1; i <= MAX; i++)
    {
        for (j = 1; j <= i; j++)
            printf("%*s", j, "");
        puts("");
    }

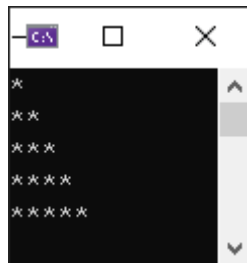
    return 0;
}
```

실습예제 7-18

파일 이름: **triangle.c**

교재: P338

## 결과





## 예제 코드

```
// file: loops.c
#define _CRT_SECURE_NO_WARNINGS

#include <stdio.h>

int main(void)
{
    int input, sum, i, j;

    do
    {
        printf("양의 정수 또는 0(종료)을 입력: ");
        scanf("%d", &input);

        for (i = 1; i <= input; i++)
        {
            for (j = 1, sum = 0; j <= i; j++)
            {
                printf("%d", j);
                j == i ? printf(" = ") : printf(" + ");
                sum += j;
            }
            printf("%d\n", sum);
        }
    } while (input > 0);

    puts("종료합니다.");

    return 0;
}
```

실습예제 7-19

파일 이름: loops.c

교재: P339~P340

## 결과

```
Microsoft Visual Studio 디버그 콘솔
양의 정수 또는 0(종료)을 입력: 10
1 = 1
1 + 2 = 3
1 + 2 + 3 = 6
1 + 2 + 3 + 4 = 10
1 + 2 + 3 + 4 + 5 = 15
1 + 2 + 3 + 4 + 5 + 6 = 21
1 + 2 + 3 + 4 + 5 + 6 + 7 = 28
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 = 36
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 = 55
양의 정수 또는 0(종료)을 입력: 0
종료합니다.
```

## 예제 코드

```
// file: mtable.c

#include <stdio.h>
#define MAX 9

int main(void)
{
    printf("=== 구구단 출력 ===\n");
    for (int i = 2; i <= MAX; i++)
    {
        printf("%d단 출력\n", i);
        for (int j = 2; j <= MAX; j++)
            printf("%d*%d = %d ", i, j, i * j);
        printf("\n");
    }

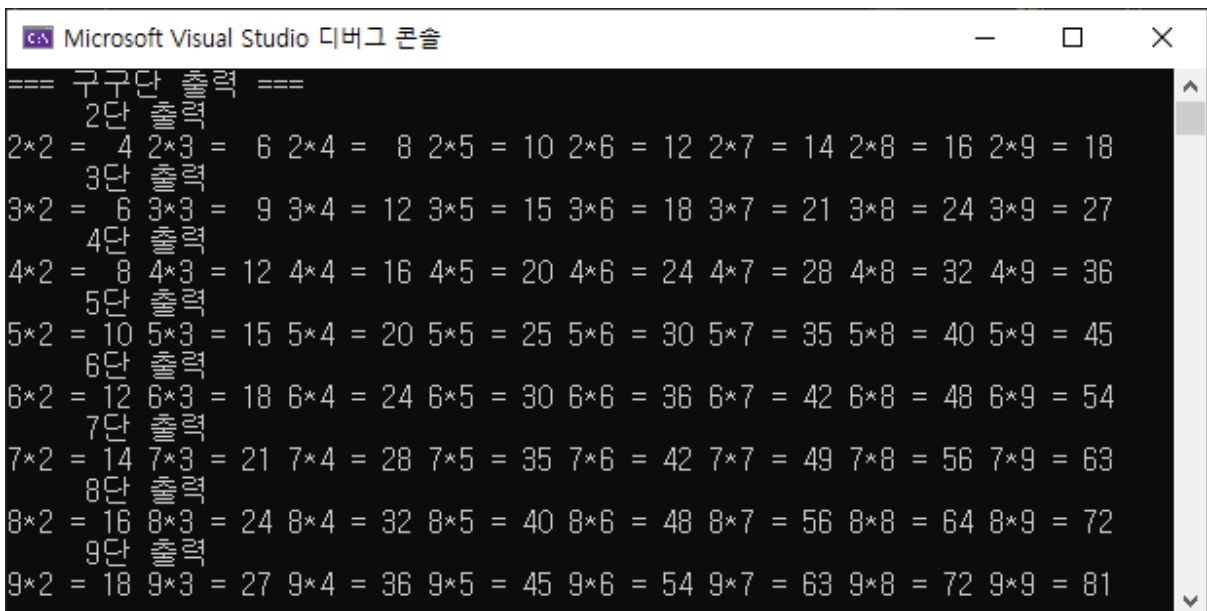
    return 0;
}
```

LAB 7-5

파일 이름: `mtable.c`

교재: P342~P343

## 결과



```
Microsoft Visual Studio 디버그 콘솔

=== 구구단 출력 ===
 2단 출력
2*2 = 4 2*3 = 6 2*4 = 8 2*5 = 10 2*6 = 12 2*7 = 14 2*8 = 16 2*9 = 18
 3단 출력
3*2 = 6 3*3 = 9 3*4 = 12 3*5 = 15 3*6 = 18 3*7 = 21 3*8 = 24 3*9 = 27
 4단 출력
4*2 = 8 4*3 = 12 4*4 = 16 4*5 = 20 4*6 = 24 4*7 = 28 4*8 = 32 4*9 = 36
 5단 출력
5*2 = 10 5*3 = 15 5*4 = 20 5*5 = 25 5*6 = 30 5*7 = 35 5*8 = 40 5*9 = 45
 6단 출력
6*2 = 12 6*3 = 18 6*4 = 24 6*5 = 30 6*6 = 36 6*7 = 42 6*8 = 48 6*9 = 54
 7단 출력
7*2 = 14 7*3 = 21 7*4 = 28 7*5 = 35 7*6 = 42 7*7 = 49 7*8 = 56 7*9 = 63
 8단 출력
8*2 = 16 8*3 = 24 8*4 = 32 8*5 = 40 8*6 = 48 8*7 = 56 8*8 = 64 8*9 = 72
 9단 출력
9*2 = 18 9*3 = 27 9*4 = 36 9*5 = 45 9*6 = 54 9*7 = 63 9*8 = 72 9*9 = 81
```

# CHAPTER

08

포인터 기초

## 예제 코드

```
// file: address.c
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

int main(void)
{
    int input;

    printf("정수 입력: ");
    scanf("%d", &input);
    printf("입력값: %d\n", input);
    printf("주소값: %d(10진수), %p(16진수)\n", (int)&input, &input);
    printf("주소값: %u(10진수), %X(16진수)\n", (unsigned)&input, (int)&input);
    printf("주소값 크기: %d\n", sizeof(&input));

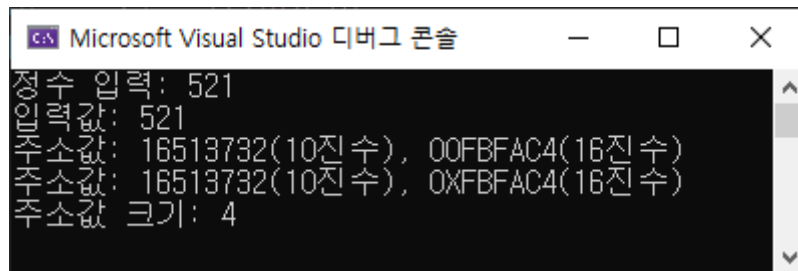
    return 0;
}
```

실습예제 8-1

파일 이름: address.c

교재: P360

## 결과



```
Microsoft Visual Studio 디버그 콘솔
정수 입력: 521
입력값: 521
주소값: 16513732(10진수), 00FBFAC4(16진수)
주소값: 16513732(10진수), 0FBFAC4(16진수)
주소값 크기: 4
```

## 예제 코드

```
// file: pointer.c
#include <stdio.h>

int main(void)
{
    int data = 100;
    int* ptrint;
    ptrint = &data;

    printf("변수명   주소값   저장값\n");
    printf("-----Wn");
    printf("   data  %p  %8dWn", &data, data);
    printf(" ptrint %p  %pWn", &ptrint, ptrint);

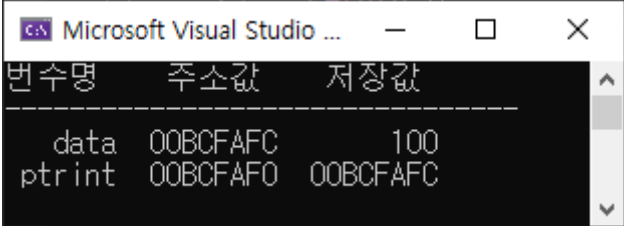
    return 0;
}
```

실습예제 8-2

파일 이름: **pointer.c**

교재: P363

## 결과



```
Microsoft Visual Studio ...
변수명   주소값   저장값
-----
data    00BCFAFC   100
ptrint  00BCFAF0   00BCFAFC
```

## 예제 코드

```
// file: basicpointer.c
#include <stdio.h>

int main(void)
{
    char c = '@';
    char* pc = &c;
    int m = 100;
    int* pm = &m;
    double x = 5.83;
    double* px = &x;

    printf("변수명   주소값   저장값\n");
    printf("-----\n");
    printf("%3s %12p %9c\n", "c", pc, c);
    printf("%3s %12p %9d\n", "m", pm, m);
    printf("%3s %12p %9f\n", "x", px, x);

    return 0;
}
```

LAB 8-1

파일 이름: `basicpointer.c`

교재: P365~P366

## 결과



변수명	주소값	저장값
c	00EFFF7B	@
m	00EFFF60	100
x	00EFFF44	5.830000

## 예제 코드

```
// file: nullpointer.c
#include <stdio.h>

int main(void)
{
    int* ptr1, * ptr2, data = 10;
    ptr1 = NULL;

    printf("%p\n", ptr1);
    //printf("%p\n", ptr2); 초기값이 없어서 컴파일 오류 발생
    printf("%d\n", data);

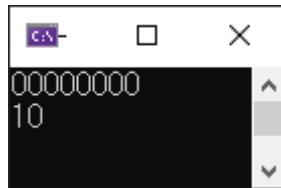
    return 0;
}
```

실습예제 8-3

파일 이름: `nullpointer.c`

교재 : P368

## 결과



## 예제 코드

```
// file: dereference.c
#include <stdio.h>

int main(void)
{
    int data = 100;
    char ch = 'A';
    int* ptring = &data;
    char* ptrchar = &ch;
    printf("간접참조 출력: %d %c\n", *ptring, *ptrchar);

    *ptring = 200; //변수 data를 *ptring로 간접참조하여 그 내용을 수정
    *ptrchar = 'B'; //변수 ch를 *ptrchar로 간접참조하여 그 내용을 수정
    printf("직접참조 출력: %d %c\n", data, ch);

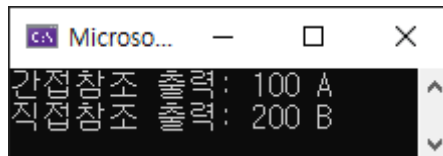
    return 0;
}
```

실습예제 8-4

파일 이름: dereference.c

교재: P370

## 결과



```
Microsoft...
간접참조 출력: 100 A
직접참조 출력: 200 B
```



## 예제 코드

```
// file: calcptr.c
#include <stdio.h>
```

```
int main(void)
{
```

```
    char* pc = (char*)100; //가능하나 잘 이용하지 않음
    int* pi = (int*)100; //가능하나 잘 이용하지 않음
    double* pd = (double*)100; //가능하나 잘 이용하지 않음
    pd = 100; //경고 발생
```

```
    printf("%u      %u      %u\n", (int)(pc - 1), (int)pc, (int)(pc + 1));
    printf("%u      %u      %u\n", (int)(pi - 1), (int)pi, (int)(pi + 1));
    printf("%u      %u      %u\n", (int)(pd - 1), (int)pd, (int)(pd + 1));
```

```
    return 0;
```

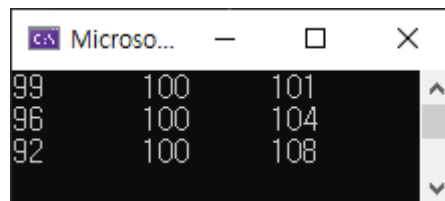
```
}
```

실습예제 8-5

파일 이름: calcptr.c

교재: P372~P373

## 결과



99	100	101
96	100	104
92	100	108

## 예제 코드

```
// file: neighborvar.c
#include <stdio.h>

int main(void)
{
    int a = 1, b = 3, c = 6;

    printf("변수명   저장값   주소값   Wn");
    printf("-----Wn");
    printf("  c       %d     %pWn", c, &c);
    printf("  b       %d     %pWn", b, &b);
    printf("  a       %d     %pWn", a, &a);

    int* p = &c;
    printf("  c       %d     %pWn", *p, p);
    printf("  c       %d     %pWn", *(p + 3), p + 3);
    printf("  c       %d     %pWn", *(p + 6), p + 6);

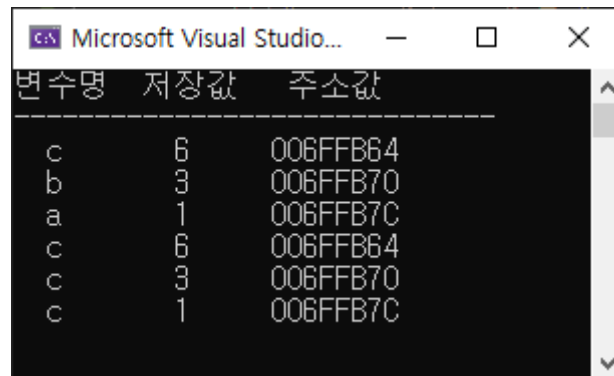
    return 0;
}
```

실습예제 8-6

파일 이름: neighborvar.c

교재: P373~P374

## 결과



변수명	저장값	주소값
c	6	006FFB64
b	3	006FFB70
a	1	006FFB7C
c	6	006FFB64
c	3	006FFB70
c	1	006FFB7C

## 예제 코드

```
// file: swap.c
#include <stdio.h>

int main(void)
{
    int m = 100, n = 200, dummy;
    printf("%d %d\n", m, n);

    //변수 m과 n을 사용하지 않고 두 변수를 서로 교환
    int* p = &m; //포인터 p가 m을 가리키도록
    dummy = *p; //변수 dummy에 m을 저장
    *p = n; //변수 m에 n을 저장
    p = &n; //포인터 p가 n을 가리키도록
    *p = dummy; //변수 n에 dummy 값 저장

    printf("%d %d\n", m, n);

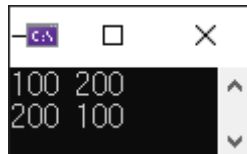
    return 0;
}
```

LAB 8-2

파일 이름: **swap.c**

교재 : P376

## 결과



```
cmd
100 200
200 100
```

## 예제 코드

```
// file: ptrtypecast.c
#include <stdio.h>

int main(void)
{
    int value = 0x61626364;
    int* pi = &value;
    char* pc = (char*)&value;

    printf("변수명   저장값           주소값\n");
    printf("-----Wn");
    printf("value   %0#x   %u\n", value, pi); //정수 출력

    //문자 포인터로 문자 출력 모듈
    for (int i = 0; i <= 3; i++)
    {
        char ch = *(pc + i);
        printf("(pc+%d) %0#6x   %2c   %u\n", i, ch, ch, pc + i);
    }

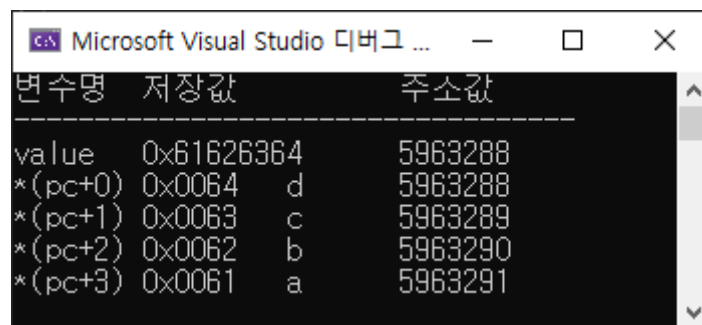
    return 0;
}
```

실습예제 8-7

파일 이름: ptrtypecast.c

교재: P379~P380

## 결과



변수명	저장값		주소값
value	0x61626364		5963288
*(pc+0)	0x0064	d	5963288
*(pc+1)	0x0063	c	5963289
*(pc+2)	0x0062	b	5963290
*(pc+3)	0x0061	a	5963291

## 예제 코드

```
// file: multipointer.c
#include <stdio.h>

int main(void)
{
    int i = 20;
    int* pi = &i; //포인터 선언
    int** dpi = &pi; //이중 포인터 선언

    printf("%p %p %p\n", &i, pi, *dpi);

    *pi = i + 2; // i = i + 2;
    printf("%d %d %d\n", i, *pi, **dpi);

    **dpi = *pi + 2; // i = i + 2;
    printf("%d %d %d\n", i, *pi, **dpi);

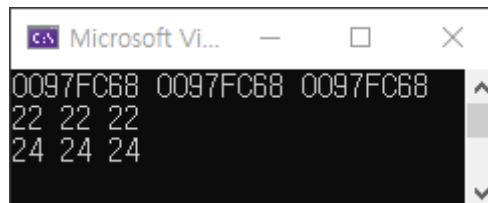
    return 0;
}
```

실습예제 8-8

파일 이름: `multipointer.c`

교재: P382

## 결과



```
Microsoft Vi...
0097FC68 0097FC68 0097FC68
22 22 22
24 24 24
```

## 예제 코드

```
// file: variousop.c
#include <stdio.h>

int main(void)
{
    int i;
    int* pi = &i; //포인터
    int** dpi = &pi; //이중포인터

    *pi = 5;
    *pi += 1; // *pi = *pi + 1과 같음
    printf("%d\n", i);

    // 후위 연산자 pi++는 전위 연산자보다 *pi보다 빠름
    printf("%d\n", (*pi)++); // *pi++는 *(pi++)로 (*pi)++와 다름
    printf("%d\n", *pi);

    *pi = 10;
    printf("%d\n", ++ * pi); // ++*pi == ++(*pi)는 같음
    printf("%d\n", ++ * *dpi); // ++**dpi == ++(**dpi)는 같음
    printf("%d\n", i);

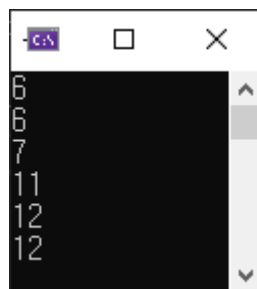
    return 0;
}
```

실습예제 8-9

파일 이름: `variousop.c`

교재 : P383~P384

## 결과



## 예제 코드

```
/* constptr.c */
#include <stdio.h>

int main(void)
{
    int i = 10, j = 20;
    const int* p = &i; /*p 수정 불가
    /*p = 20; 오류
    p = &j;
    printf("%d\n", *p);

    double d = 7.8, e = 2.7;
    double* const pd = &d;
    //pd = &e; 다른 주소를 저장 불가
    *pd = 4.4;
    printf("%f\n", *pd);

    return 0;
}
```

실습예제 8-10

파일 이름: `constptr.c`

교재: P386

## 결과



```
C:\>
20
4.400000
```

## 예제 코드

```
// file: sumpointer.c
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

int main(void)
{
    double x, y;
    double* px = &x;
    double* py = &y;

    //포인터 변수 px와 py를 사용
    printf("두 실수 입력: ");
    scanf("%lf %lf", px, py);
    //합 출력
    printf("%.2f + %.2f = %.2f\n", *px, *py, *px + *py);

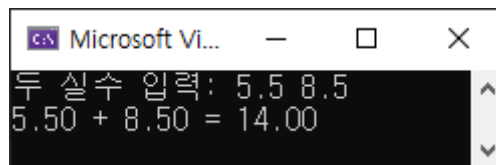
    return 0;
}
```

LAB 8-3

파일 이름: `sumpointer.c`

교재: P388

## 결과





# CHAPTER

09

배열

## 예제 코드

```
// file: declarearray.c
#include <stdio.h>

#define SIZE 5

int main(void)
{
    //배열 선언
    int score[SIZE];

    //배열 원소에 값 저장
    score[0] = 78;
    score[1] = 97;
    score[2] = 85;
    //score[3]에 쓰레기값 저장
    score[4] = 91;
    //score[5]은 문법 오류는 안 나지만 실행 오류 발생

    //배열 원소 출력
    for (int i = 0; i < SIZE; i++)
        printf("%d ", score[i]);
    printf("\n");

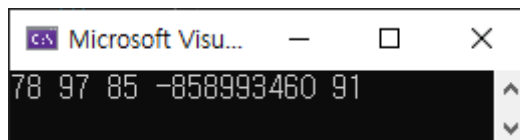
    return 0;
}
```

실습예제 9-1

파일 이름: declarearray.c

교재: P395~P396

## 결과



## 예제 코드

```
// file: initarray.c
#include <stdio.h>
#define SIZE 6

int main(void)
{
    //배열 score의 선언과 초기화
    double score[] = { 89.3, 79.2, 84.83, 76.8, 92.52, 97.4 };
    double sum = 0;

    //for 문을 이용하여 합을 구함
    for (int i = 0; i < SIZE; i++)
    {
        sum += score[i];
        printf("score[%d] = %.2f\n", i, score[i]);
    }
    printf("성적의 합은 %.2f이고 평균은 %.2f이다.\n", sum, sum / SIZE);

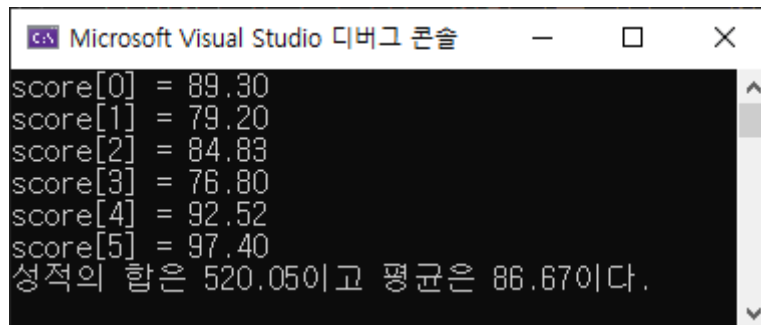
    return 0;
}
```

실습예제 9-2

파일 이름: `initarray.c`

교재 : P398~P399

## 결과



```
Microsoft Visual Studio 디버그 콘솔
score[0] = 89.30
score[1] = 79.20
score[2] = 84.83
score[3] = 76.80
score[4] = 92.52
score[5] = 97.40
성적의 합은 520.05이고 평균은 86.67이다.
```

## 예제 코드

LAB 9-1

파일 이름: `inputarray.c`

교재 : P401

```
// file: inputarray.c
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

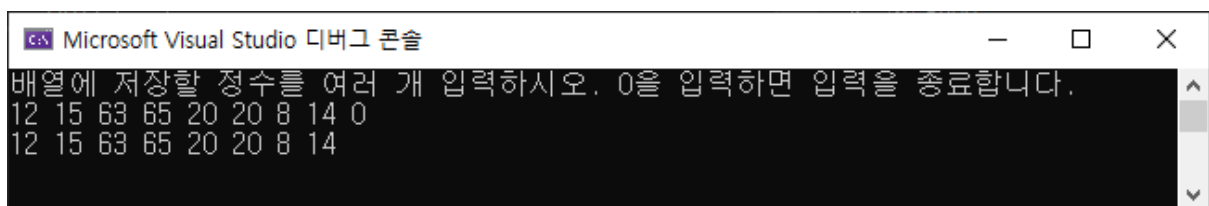
int main(void)
{
    //초기화로 모든 원소에 0을 저장
    int input[20] = { 0 };

    printf("배열에 저장할 정수를 여러 개 입력하시오.");
    printf(" 0을 입력하면 입력을 종료합니다.\n");
    int i = 0;
    do {
        scanf("%d", &input[i]);
    } while (input[i++] != 0);

    i = 0;
    while (input[i] != 0) {
        printf("%d ", input[i++]);
    }
    puts("");

    return 0;
}
```

## 결과



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio 디버그 콘솔". The console output displays the program's execution: it prompts for input, receives the sequence "12 15 63 65 20 20 8 14 0", and then prints the same sequence followed by a newline. The input "0" terminates the loop.

```
Microsoft Visual Studio 디버그 콘솔
배열에 저장할 정수를 여러 개 입력하시오. 0을 입력하면 입력을 종료합니다.
12 15 63 65 20 20 8 14 0
12 15 63 65 20 20 8 14
```

## 예제 코드

```
// file: twodarray.c
#include <stdio.h>

#define ROWSIZE 2
#define COLSIZE 3

int main(void)
{
    //2차원 배열 선언
    int td[ROWSIZE][COLSIZE];

    //2차원 배열 원소에 값 저장
    td[0][0] = 1; td[0][1] = 2; td[0][2] = 3;
    td[1][0] = 4; td[1][1] = 5; td[1][2] = 6;

    printf("반복문 for를 이용하여 출력\n");
    for (int i = 0; i < ROWSIZE; i++)
    {
        for (int j = 0; j < COLSIZE; j++)
            printf("td[%d][%d] == %d ", i, j, td[i][j]);
        printf("\n");
    }

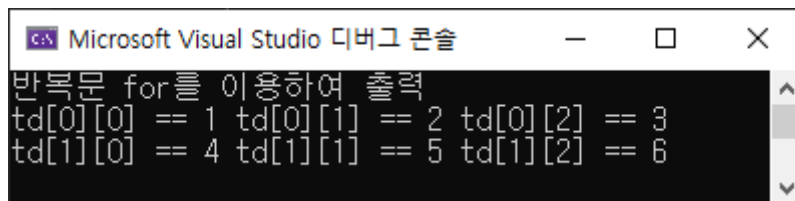
    return 0;
}
```

실습예제 9-3

파일 이름: `twodarray.c`

교재 : P405~P406

## 결과



```
Microsoft Visual Studio 디버그 콘솔
반복문 for를 이용하여 출력
td[0][0] == 1 td[0][1] == 2 td[0][2] == 3
td[1][0] == 4 td[1][1] == 5 td[1][2] == 6
```

## 예제 코드

```
// file: inittwodarray.c
#include <stdio.h>

#define ROWSIZE 2
#define COLSIZE 3

int main(void)
{
    //2차원 배열 초기화
    int td[][3] = { {1}, {1, 2, 3} };

    printf("반복문 for를 이용하여 출력\n");
    for (int i = 0; i < ROWSIZE; i++)
    {
        for (int j = 0; j < COLSIZE; j++)
            printf("%d ", td[i][j]);
        printf("\n");
    }

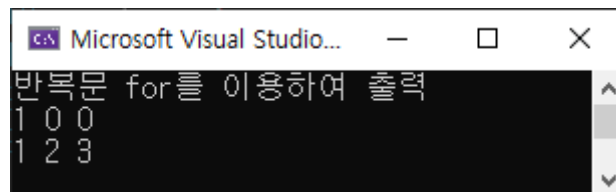
    return 0;
}
```

실습예제 9-4

파일 이름: `inittwodarray.c`

교재: P408

## 결과



```
Microsoft Visual Studio...
반복문 for를 이용하여 출력
1 0 0
1 2 3
```

## 예제 코드

```
// file: tdscore.c
#include <stdio.h>

#define ROWSIZE 4
#define COLSIZE 2

int main(void)
{
    int sum = 0, midsum = 0, finalsum = 0;

    //2차원 배열 초기화
    int score[][COLSIZE] = { 95,85,90,88,86,90,88,78 };

    printf("  중간      기말\n");
    printf("-----\n");
    for (int i = 0; i < ROWSIZE; i++)
    {
        for (int j = 0; j < COLSIZE; j++)
        {
            printf("%10d ", score[i][j]);
            sum += score[i][j];
            if (j == 0) midsum += score[i][j];
            else finalsum += score[i][j];
        }
        puts("");
    }

    printf("-----\n");
    printf("평균: %7.2f  %7.2f\n", (double)midsum / ROWSIZE, (double)finalsum / ROWSIZE);
    printf("\n성적의 합은 %d이고 ", sum);
    printf("평균은 %.2f이다.", (double)sum / (ROWSIZE * COLSIZE));

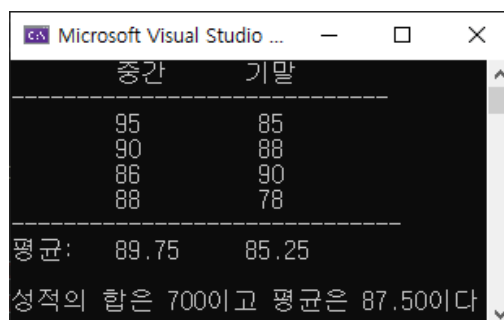
    return 0;
}
```

실습예제 9-5

파일 이름: tdscore.c

교재: P409~P410

## 결과



```
Microsoft Visual Studio ...
  중간      기말
-----
      95      85
      90      88
      86      90
      88      78
-----
평균:   89.75   85.25
성적의 합은 700이고 평균은 87.50이다
```

# 예제 코드

실습예제 9-6

파일 이름: **threedary.c**

교재 : P413~P414

```
// file: threedary.c
#include <stdio.h>

#define ROWSIZE 4
#define COLSIZE 2

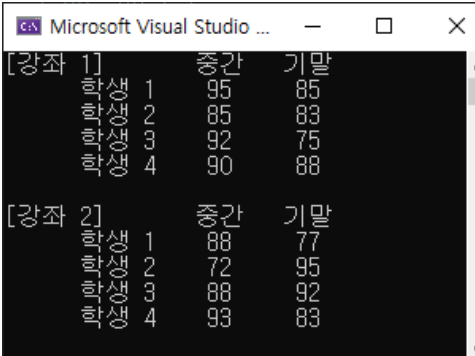
int main(void)
{
    // 3차원 배열 초기화, 첫 번째 크기는 지정하지 않을 수 있음
    int score[ ][ROWSIZE][COLSIZE] = {
        { { 95, 85 },
          { 85, 83 },
          { 92, 75 },
          { 90, 88 } },
        { { 88, 77 },
          { 72, 95 },
          { 88, 92 },
          { 93, 83 } }
    };

    for (int i = 0; i < 2; i++)
    {
        if (i == 0) printf("[강좌 1]");
        else printf("[강좌 2]");
        printf("%11s%7sWn", "중간", "기말");

        for (int j = 0; j < ROWSIZE; j++)
        {
            printf("%10s%2d", "학생", j + 1);
            for (int k = 0; k < COLSIZE; k++)
                printf("%6d ", score[i][j][k]);
            printf("Wn");
        }
        printf("Wn");
    }

    return 0;
}
```

## 결과



[강좌 1]		중간	기말
학생 1	95	85	
학생 2	85	83	
학생 3	92	75	
학생 4	90	88	

[강좌 2]		중간	기말
학생 1	88	77	
학생 2	72	95	
학생 3	88	92	
학생 4	93	83	



## 예제 코드

LAB 9-2

파일 이름: `arrayprint.c`

교재: P415~P416

```
// file: arrayprint.c
#include <stdio.h>

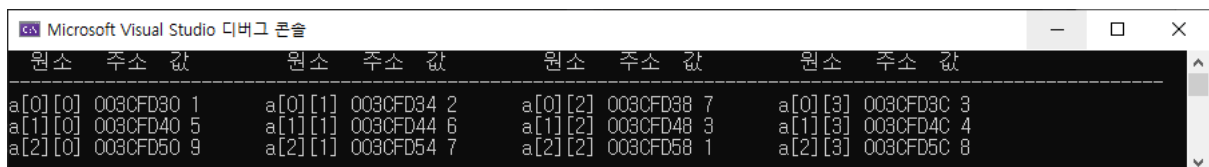
int main(void)
{
    int a[3][4] = {
        { 1,2,7,3 }, //행의 인덱스 값 0
        { 5,6,3,4 }, //행의 인덱스 값 1
        { 9,7,1,8 }  //행의 인덱스 값 2
    };

    printf("%6s %6s %3s      ", "원소", "주소", "값");
    printf("%6s %6s %3s      ", "원소", "주소", "값");
    printf("%6s %6s %3s      ", "원소", "주소", "값");
    printf("%6s %6s %3sWn", "원소", "주소", "값");
    printf("-----");
    printf("-----Wn");

    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 4; j++)
            printf("a[%d][%d] %p %d  ", i, j, &a[i][j], a[i][j]);
        puts("");
    }

    return 0;
}
```

## 결과



원소	주소	값	원소	주소	값	원소	주소	값	원소	주소	값
a[0][0]	003CFD30	1	a[0][1]	003CFD34	2	a[0][2]	003CFD38	7	a[0][3]	003CFD3C	3
a[1][0]	003CFD40	5	a[1][1]	003CFD44	6	a[1][2]	003CFD48	3	a[1][3]	003CFD4C	4
a[2][0]	003CFD50	9	a[2][1]	003CFD54	7	a[2][2]	003CFD58	1	a[2][3]	003CFD5C	8

## 예제 코드

```
// file: array.c
#include <stdio.h>
#define SIZE 3

int main(void)
{
    int score[] = { 89,98,76 };

    //배열 이름 score는 첫 번째 원소의 주소
    printf("score: %p, &score[0]: %p\n", score, &score);

    //배열 이름 score는 첫 번째 값
    printf("*score: %d, score[0]: %d\n\n", *score, score[0]);

    printf("첨자    주소    저장값\n");
    //배열 이름 score를 사용한 주소와 원소값 참조
    for (int i = 0; i < SIZE; i++)
        printf("%2d %11p %6d\n", i, (score + i), *(score + i));

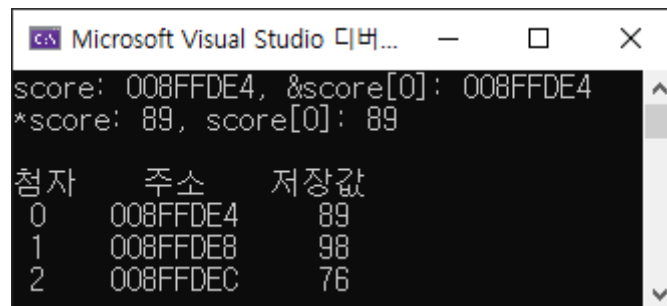
    return 0;
}
```

실습예제 9-7

파일 이름: **array.c**

교재: P418~P419

## 결과



```
Microsoft Visual Studio 디버...
score: 008FFDE4, &score[0]: 008FFDE4
*score: 89, score[0]: 89

첨자    주소    저장값
0      008FFDE4    89
1      008FFDE8    98
2      008FFDEC    76
```

## 예제 코드

실습예제 9-8

파일 이름: ptrtoary.c

교재 : P421~P422

```
// file: ptrtoary.c
#include <stdio.h>

int main(void)
{
    int a[4] = { 1,3,6,8 };
    int* pa = a; //a == &a[0]

    //새로 선언한 포인터 변수를 사용
    printf("%d      %d      %d      %d\n", *(pa), *(pa + 1), *(pa + 2), *(pa + 3));
    //새로 선언한 포인터 변수를 배열과 같이 사용 가능
    printf("%d      %d      %d      %d\n", pa[0], pa[1], pa[2], pa[3]);
    //원래 배열로도 사용 가능
    printf("%d      %d      %d      %d\n", a[0], a[1], a[2], a[3]);
    //원래 배열 주소의 간접참조로도 가능
    printf("%d      %d      %d      %d\n", *a, *(a + 1), *(a + 2), *(a + 3));

    pa = &a[1]; // &a[1] = a+1
    //a[1]을 하나 증가시켜 a[2]를 가리키도록
    printf("%d      ", ++ * pa);
    //a[1]를 출력한 후 pa를 하나 증가시켜 a[2]를 가리키도록
    printf("%d      ", *pa++);
    //a[2]를 하나 감소시켜 출력
    printf("%d      ", -- * pa);
    //a[2]를 출력하고 하나 감소시킴
    printf("%d\n", (*pa)--);
    //현재 포인터 변수 pa는 a[2]를 가리키고 있으며 다음으로 배열 모두 출력
    printf("%d      %d      %d      %d\n", *(pa - 2), *(pa - 1), *pa, *(pa + 1));

    return 0;
}
```

## 결과

```
C:\> Microsoft Visual Stu...
1      3      6      8
1      3      6      8
1      3      6      8
1      3      6      8
4      4      5      5
1      4      4      8
```

## 예제 코드

실습예제 9-9

파일 이름: ptypecast.c

교재: P424

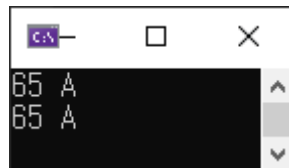
```
// file: ptypecast.c
#include <stdio.h>

int main(void)
{
    char c[4] = { 'A', 'WO', 'WO', 'WO' }; //문자 'A' 코드값: 65
    //int *pi = &c[0]; 경고
    int* pi = (int*)&c[0];

    printf("%d %c\n", (int)c[0], c[0]);
    printf("%d %c\n", *pi, (char)*pi);

    return 0;
}
```

## 결과



## 예제 코드

```
// file: tdaryptr.c
#include <stdio.h>

#define ROW 2
#define COL 3

int main(void)
{
    int td[][COL] = { {8,5,4}, {2,7,6} };

    **td = 10; //td[0][0] = 10;
    *td[1] = 20; //td[1][0] = 20;

    for (int i = 0, cnt = 0; i < ROW; i++)
    {
        for (int j = 0; j < COL; j++, cnt++)
        {
            printf("%d %d %d, ", *(td + cnt), *(td[i] + j), (*(td + i) + j));
        }
        printf("\n");
    }

    printf("%d, %d, %d\n", sizeof(td), sizeof(td[0]), sizeof(td[1]));
    printf("%p %p %p\n", td, td[0], td[1]);
    printf("%p %p\n", &td[0][0], &td[1][0]);

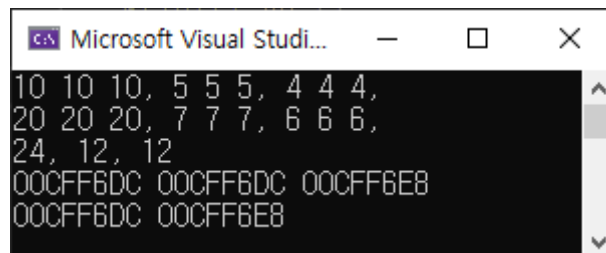
    return 0;
}
```

실습예제 9-10

파일 이름: **tdaryptr.c**

교재: P426~P427

## 결과



```
Microsoft Visual Studi...
10 10 10, 5 5 5, 4 4 4,
20 20 20, 7 7 7, 6 6 6,
24, 12, 12
00CFF6DC 00CFF6DC 00CFF6E8
00CFF6DC 00CFF6E8
```

## 예제 코드

```
// file: doubletoint.c
#include <stdio.h>

int main(void)
{
    double dint[4] = { 0.0 };

    int* p = (int*)dint;
    p[0] = 1;
    p[1] = 2;
    p[2] = 3;
    p[3] = 4;

    for (int i = 0; i < 4; i++)
        printf("%p %d\n", p + i, *(p + i));

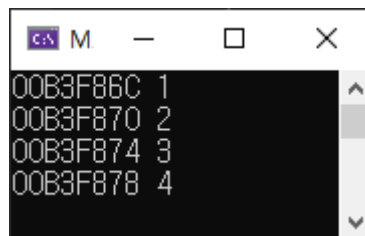
    return 0;
}
```

LAB 9-3

파일 이름: doubletoint.c

교재 : P428~P429

## 결과



## 예제 코드

실습예제 9-11

파일 이름: `pointerarray.c`

교재: P431~P432

```
// file: pointerarray.c
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

#define SIZE 3

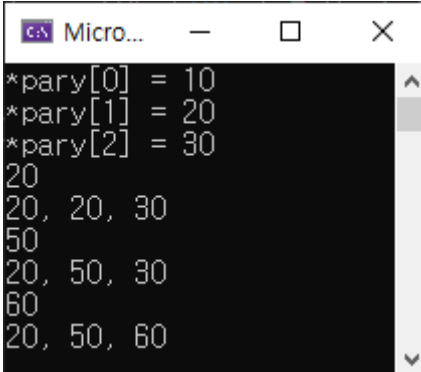
int main(void)
{
    //포인터 배열 변수 선언
    int* pary[SIZE] = { NULL };
    int a = 10, b = 20, c = 30;

    pary[0] = &a;
    pary[1] = &b;
    pary[2] = &c;
    for (int i = 0; i < SIZE; i++)
        printf("*pary[%d] = %d\n", i, *pary[i]);

    for (int i = 0; i < SIZE; i++)
    {
        scanf("%d", pary[i]);
        printf("%d, %d, %d\n", a, b, c);
    }

    return 0;
}
```

## 결과



```
cs Micro...
*pary[0] = 10
*pary[1] = 20
*pary[2] = 30
20
20, 20, 30
50
20, 50, 30
60
20, 50, 60
```

## 예제 코드

실습예제 9-12

파일 이름: `tdarypointer.c`

교재: P435~P437

```
// file: tdarypointer.c
#include <stdio.h>

int main(void)
{
    int a[] = { 8,2,8,1,3 };
    int* p = a;

    printf("%2d, %2d\n", *(p + 1), *(p + 4)); //배열 원소 a[1], a[4] 참조
    printf("%2d, %2d\n", p[1], p[4]); //배열 원소 a[1], a[4] 참조
    printf("sizeof(a) = %d, sizeof(p) = %d\n", sizeof(a), sizeof(p));

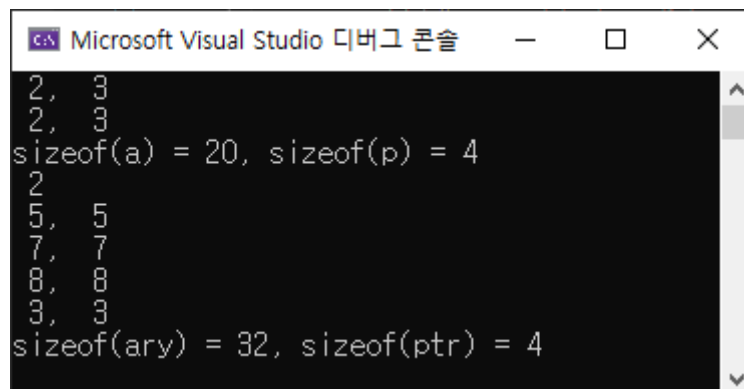
    printf("%2d\n", **p); //배열 두 번째 원소 참조
    //printf("%2d\n", **a); //오류

    int ary[][4] = { 5,7,6,2,7,8,1,3 }; //2행 4열
    int(*ptr)[4] = ary; //열이 4인 배열을 가리키는 포인터
    //int* ptr[4] = ary; //포인터 배열

    printf("%2d, %2d\n", **ary, **ptr++); //첫 번째 원소, ptr은 1 증가됨
    printf("%2d, %2d\n", **ary, **ptr); //두 번째 원소, ptr은 1 증가됨
    ptr = ary; //다시 ptr이 배열 ary처음을 가리키도록
    printf("%2d, %2d\n", *(ary[1] + 1), *(ptr[1] + 1)); //2행 2열, 8
    printf("%2d, %2d\n", *(ary + 1) + 3, *(ptr + 1) + 3); //2행 4열, 3
    printf("sizeof(ary) = %d, sizeof(ptr) = %d\n", sizeof(ary), sizeof(ptr));

    return 0;
}
```

## 결과



```
Microsoft Visual Studio 디버그 콘솔
2, 3
2, 3
sizeof(a) = 20, sizeof(p) = 4
2, 5
7, 7
8, 8
3, 3
sizeof(ary) = 32, sizeof(ptr) = 4
```



## 예제 코드

```
// file: arraysize.c
#include <stdio.h>

int main(void)
{
    int data[] = { 3,4,5,7,9 };

    printf("%zd %zdWn", sizeof(data), sizeof(data[0]));
    printf("일차원 배열: 배열 크기 == %zdWn", sizeof(data), sizeof(data[0]));

    //4x3 행렬
    double x[][3] = { {1,2,3}, {7,8,9}, {4,5,6}, {10,11,12} };

    printf("%zd %zd %zd %zdWn", sizeof(x), sizeof(x[0]), sizeof(x[1]), sizeof(x[0][0]));
    int rowsize = sizeof(x) / sizeof(x[0]);
    int colsize = sizeof(x[0]) / sizeof(x[0][0]);
    printf("이차원 배열: 행수 == %d 열수 == %dWn", rowsize, colsize);
    printf("이차원 배열: 전체 원소 수 == %zdWn", sizeof(x) / sizeof(x[0][0]));

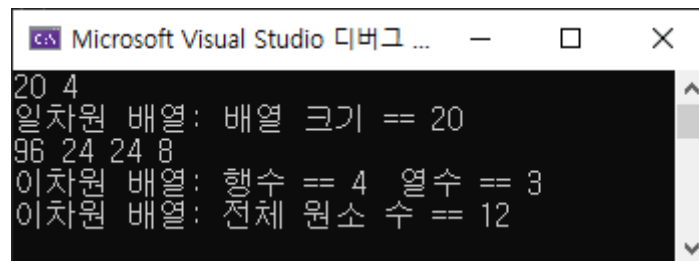
    return 0;
}
```

실습예제 9-13

파일 이름: **arraysize.c**

교재: P438~P439

## 결과



```
Microsoft Visual Studio 디버그 ...
20 4
일차원 배열: 배열 크기 == 20
96 24 24 8
이차원 배열: 행수 == 4 열수 == 3
이차원 배열: 전체 원소 수 == 12
```

## 예제 코드

```
// file: twoarysample.c
#include <stdio.h>

int main(void)
{
    int abc[4][3] = {
        {1,2,3},
        {5,6,7},
        {9,10,11},
        {13,14,15}
    };

    int rowsize = sizeof(abc) / sizeof(abc[0]);
    int colsize = sizeof(abc[0]) / sizeof(abc[0][0]);

    printf("각 행의 첫 주소 출력: Wn");
    for (int i = 0; i < rowsize; i++)
        printf("%p ", abc[i]);
    printf("WnWn");

    printf("2행 원소의 주소와 값 출력: Wn");
    int* p = abc[1];
    for (int i = 0; i < colsize; i++)
    {
        printf("%p ", p);
        printf("%dWn", *p++);
    }

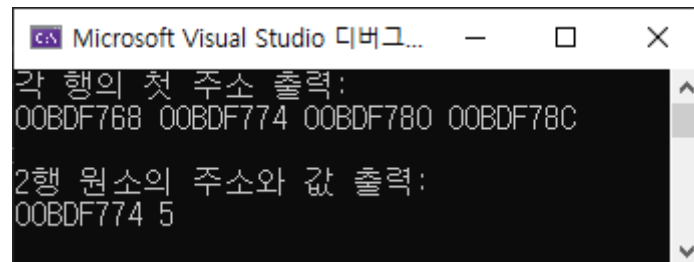
    return 0;
}
```

LAB 9-4

파일 이름: twoarysample.c

교재: P441~P442

## 결과



```
Microsoft Visual Studio 디버그...
각 행의 첫 주소 출력:
00BDF768 00BDF774 00BDF780 00BDF78C

2행 원소의 주소와 값 출력:
00BDF774 5
```

# CHAPTER

10

함수 기초

## 예제 코드

실습예제 10-1

파일 이름: `functionadd2.c`

교재: P454~P455

```
// file: functionadd2.c
#include <stdio.h>

//int add2(int a, int b); //이 위치도 가능

int main(void)
{
    int a = 3, b = 5;
    int add2(int a, int b); //int add2(int, int)도 가능

    //위 함수원형이 없으면 함수호출에서 오류
    int sum = add2(a, b);
    printf("합: %d\n", sum);

    return 0;
}

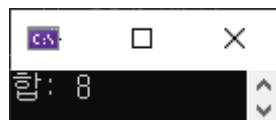
//함수 add2의 함수구현 또는 함수정의 부분
int add2(int a, int b)
{
    int sum = a + b;

    return (sum); //괄호 생략 가능
}

//위 main() 함수에서 호출이 없으므로 이 함수구현은 실행x
int findMin2(int x, int y)
{
    int min = x < y ? x : y;

    return (min);
}
```

## 결과



## 예제 코드

```
// file: getsum.c
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

int getsum(int); //함수원형

int main(void)
{
    int max = 0;

    printf("1에서 n까지의 합을 구할 n을 입력하시오. >> ");
    scanf("%d", &max);

    printf("1에서 %d까지의 합: %d\n", max, getsum(max)); //함수 호출

    return 0;
}

int getsum(int n)
{
    int sum = 0;
    for (int i = 1; i <= n; i++)
        sum += i;

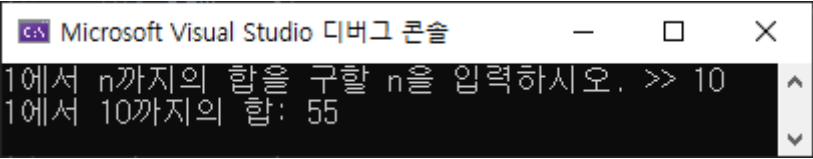
    return sum;
}
```

LAB 10-1

파일 이름: **getsum.c**

교재: P457~P458

## 결과



```
cs Microsoft Visual Studio 디버그 콘솔
1에서 n까지의 합을 구할 n을 입력하시오. >> 10
1에서 10까지의 합: 55
```

# 예제 코드

```
// file: functioncall.c
#include <stdio.h>

int add2(int a, int b); //int add2(int, int)도 가능
int findMax2(int, int); //int findMax2(int a, int b)도 가능
void printMin(int, int); //int printMin(int a, int b)도 가능

int main(void)
{
    int a = 3, b = 5;

    int max = findMax2(a, b);
    printf("최대: %d\n", max);
    printf("합: %d\n", add2(a, b));

    //반환값이 없는 함수호출은 일반 문장처럼 사용
    printMin(2, 5);

    return 0;
}

//이하 함수 add2, findMax2, findMin2, printMin 구현
int add2(int a, int b)
{
    int sum = a + b;

    return (sum);
}

int findMax2(int a, int b)
{
    int max = a > b ? a : b;

    return max;
}

int findMin2(int x, int y)
{
    int min = x < y ? x : y;

    return (min);
}

void printMin(int a, int b)
{
    int min = a < b ? a : b;
    printf("최소: %d\n", min);

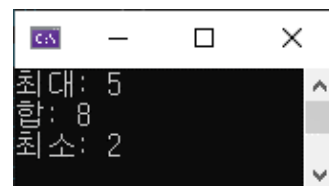
    return; //생략 가능
}
```

실습예제 10-2

파일 이름: `functioncall.c`

교재: P460~P462

## 결과



## 예제 코드

실습예제 10-3

파일 이름: `arrayparam.c`

교재: P465~P456

```
// file: arrayparam.c
#include <stdio.h>

//int sumaryf(int ary[], int SIZE);
int sumary(int* ary, int SIZE);

int main(void)
{
    int point[] = { 95,88,76,54,85,33,65,78,99,82 };
    int* address = point;
    int aryLength = sizeof(point) / sizeof(int);

    int sum = 0;
    for (int i = 0; i < aryLength; i++)
        sum += *(point + i);
        //sum += *(point++); 오류
        //sum += *(address++); 가능

    printf("메인에서 구한 합은 %d\n", sum);
    address = point;
    printf("함수 sumary() 호출로 구한 합은 %d\n", sumary(point, aryLength));
    printf("함수 sumary() 호출로 구한 합은 %d\n", sumary(&point[0], aryLength));
    printf("함수 sumary() 호출로 구한 합은 %d\n", sumary(address, aryLength));

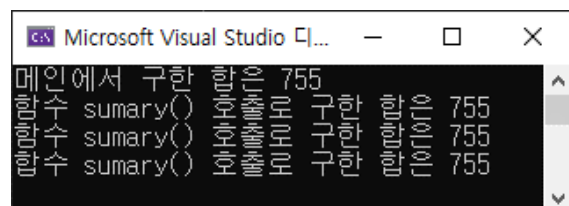
    return 0;
}

//int sumary(int ary[], int SIZE)도 가능
int sumary(int* ary, int SIZE)
{
    int sum = 0;

    for (int i = 0; i < SIZE; i++)
    {
        //sum += ary[i]; 가능
        //sum += *(ary + i); 가능
        sum += *ary++;
        //sum += *(ary++); 가능
    }

    return sum;
}
```

## 결과



```
Microsoft Visual Studio 디...
메인에서 구한 합은 755
함수 sumary() 호출로 구한 합은 755
함수 sumary() 호출로 구한 합은 755
함수 sumary() 호출로 구한 합은 755
```

## 예제 코드

```
// file: twodarrayfunction.c
#include <stdio.h>

//2차원 배열값을 모두 더하는 함수원형
double sum(double data[][3], int, int);
//2차원 배열값을 모두 출력하는 함수원형
void printarray(double data[][3], int, int);

int main(void)
{
    //4 x 3 행렬을 위한 이차원 배열 선언 및 초기화
    double x[][3] = { {1,2,3}, {7,8,9}, {4,5,6}, {10,11,12} };

    int rowsize = sizeof(x) / sizeof(x[0]);
    int colsize = sizeof(x[0]) / sizeof(x[0][0]);
    printf("2차원 배열의 자료값은 다음과 같습니다.\n");
    printarray(x, rowsize, colsize);
    printf("2차원 배열 원소합은 %.3lf 입니다.\n", sum(x, rowsize, colsize));

    return 0;
}

//배열값을 모두 더하는 함수
double sum(double (*data)[3], int rowsize, int colsize)
//double sum(double data[][3], int rowsize, int rowsize)
{
    double total = 0;

    for (int i = 0; i < rowsize; i++)
        for (int j = 0; j < colsize; j++)
            total += data[i][j];

    return total;
}

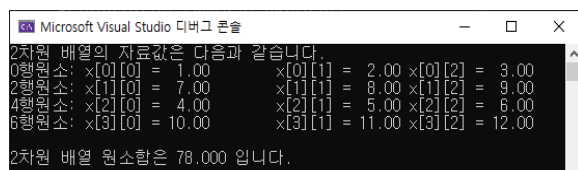
//배열값을 모두 출력하는 함수
void printarray(double (*data)[3], int rowsize, int colsize)
//void printarray(double data[][3], int rowsize, int colsize)
{
    for (int i = 0; i < rowsize; i++)
    {
        printf("%d행원소: ", i + 1);
        for (int j = 0; j < colsize; j++)
            printf("x[%d][%d] = %.2lf", i, j, data[i][j]);
        printf("\n");
    }
    printf("\n");
}
```

실습예제 10-4

파일 이름: twodarrayfunction.c

교재: P468~P469

## 결과



```
Microsoft Visual Studio 디버그 콘솔
2차원 배열의 자료값은 다음과 같습니다.
0행원소: x[0][0] = 1.00 x[0][1] = 2.00 x[0][2] = 3.00
1행원소: x[1][0] = 7.00 x[1][1] = 8.00 x[1][2] = 9.00
2행원소: x[2][0] = 4.00 x[2][1] = 5.00 x[2][2] = 6.00
3행원소: x[3][0] = 10.00 x[3][1] = 11.00 x[3][2] = 12.00
2차원 배열 원소합은 78.000 입니다.
```



# 예제 코드

LAB 10-2

파일 이름: `incrementary.c`

교재: P472~P473

```
// file: incrementary.c
#include <stdio.h>

//함수 incrementary의 함수원형
void incrementary(int ary[], int n, int SIZE);
//함수 printary의 함수원형
void printary(int* data, int SIZE);

int main(void)
{
    int data[] = { 4,7,2,3,5 };
    int aryLength = sizeof(data) / sizeof(int);

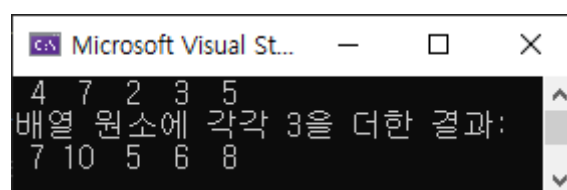
    //배열 출력을 위해 printary() 함수 호출
    printary(data, aryLength);
    //배열 원소를 모두 3씩 증가시키기 위해 incrementary() 함수 호출
    incrementary(data, 3, aryLength);
    printf("배열 원소에 각각 3을 더한 결과: \n");
    //결과를 알아보기 printary() 함수 호출
    printary(data, aryLength);

    return 0;
}

//배열 크기가 SIZE인 배열 ary의 모든 원소를 n만큼 증가시키는 함수
void incrementary(int ary[], int n, int SIZE)
{
    for (int i = 0; i < SIZE; i++)
        *(ary + i) += n; //또는 ary[i] += n;
}

//배열 크기가 SIZE인 배열 ary의 모든 원소를 출력하는 함수
void printary(int* data, int SIZE)
{
    for (int i = 0; i < SIZE; i++)
        printf("%2d ", data[i]);
    printf("\n");
}
```

## 결과



```
Microsoft Visual St...
4 7 2 3 5
배열 원소에 각각 3을 더한 결과:
7 10 5 6 8
```

## 예제 코드

실습예제 10-5

파일 이름: **factorial.c**

교재: P475

```
// file: factorial.c
#include <stdio.h>

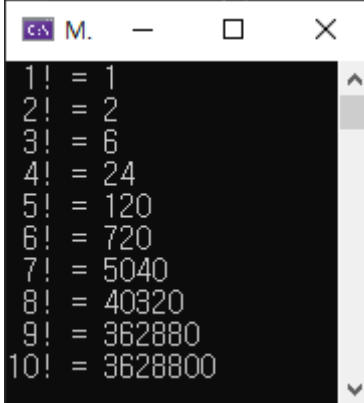
int factorial(int); //함수원형

int main(void)
{
    for (int i = 1; i <= 10; i++)
        printf("%2d! = %d\n", i, factorial(i));

    return 0;
}

// n! 구하는 함수
int factorial(int number)
{
    if (number <= 1)
        return 1;
    else
        return (number * factorial(number - 1));
}
```

## 결과



```
1! = 1
2! = 2
3! = 6
4! = 24
5! = 120
6! = 720
7! = 5040
8! = 40320
9! = 362880
10! = 3628800
```

## 예제 코드

```
// file: rand.c
#include <stdio.h>

#include <stdlib.h> //rand()를 위한 헤더파일 포함

int main(void)
{
    printf("0 ~ %5d 사이의 난수 5개: rand()\n", RAND_MAX);
    for (int i = 0; i < 5; i++)
        printf("%5d ", rand());
    puts("");

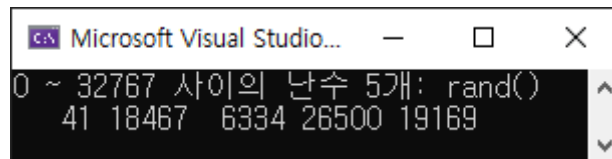
    return 0;
}
```

실습예제 10-6

파일 이름: **rand.c**

교재: P478

## 결과



```
Microsoft Visual Studio...
0 ~ 32767 사이의 난수 5개: rand()
41 18467 6334 26500 19169
```

## 예제 코드

```
// file: srand.c
#include <stdio.h>
#include <stdlib.h> //rand(), srand()를 위한 헤더파일 포함

#include <time.h> //time()을 위한 헤더파일 포함

#define MAX 100

int main(void)
{
    long seconds = (long)time(NULL);
    srand(seconds);

    printf("1 ~ %5d 사이의 난수 5개:\n", MAX);
    for (int i = 0; i < 5; i++)
        printf("%5d ", rand() % MAX + 1);
    puts("");

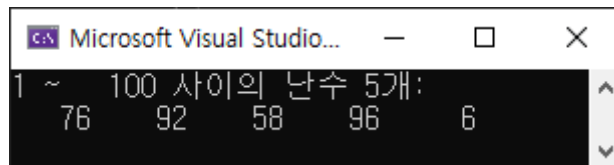
    return 0;
}
```

실습예제 10-7

파일 이름: **srand.c**

교재: P479~P480

## 결과



## 예제 코드

```
// file: math.c
#include <stdio.h>

#include <math.h> //수학 관련 다양한 함수헤더 포함 헤더파일

int main(void)
{
    printf("   i           i제곱      i세제곱      제곱근(sqrt)Wn");
    printf("-----Wn");
    for (int i = 3; i < 7; i++)
        printf("%3d      %7.1f      %9.1f      %9.1fWn", i, pow(i, 2), pow(i, 3), sqrt(i));
    printf("Wn");

    printf("exp(1.0) == %5.2f, ", exp(1.0));
    printf("pow(2.72, 1.0) == %5.2f, ", pow(2.72, 1.0));
    printf("sqrt(49) == %5.2fWn", sqrt(49));
    printf("abs(-10) == %3d, ", abs(-10));
    printf("ceil(7.1) == %5.2f, ", ceil(7.1));
    printf("floor(6.9) == %5.2fWn", floor(6.9));

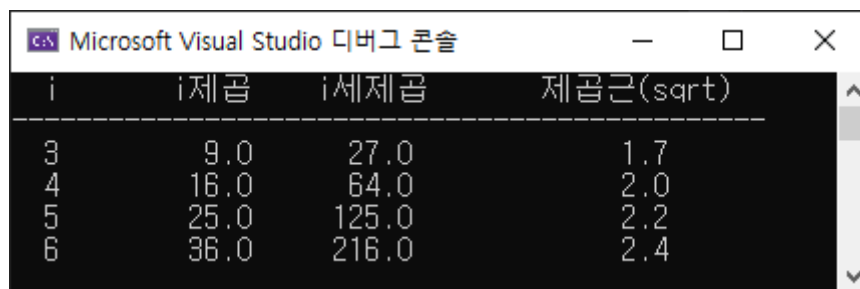
    return 0;
}
```

실습예제 10-8

파일 이름: **math.c**

교재: P481~P482

## 결과



i	i제곱	i세제곱	제곱근(sqrt)
3	9.0	27.0	1.7
4	16.0	64.0	2.0
5	25.0	125.0	2.2
6	36.0	216.0	2.4

## 예제 코드

```
// file: char.c
#define _CRT_SECURE_NO_WARNINGS //scanf() 오류 방지
#include <stdio.h>

#include <ctype.h> //문자 관련 함수는 헤더파일 ctype.h에 매크로로 정의

void print2char(char);

int main(void)
{
    char ch;

    printf("알파벳(종료x) 또는 다른 문자 입력하세요.\n");
    do
    {
        printf("문자 입력 후 Enter: ");
        scanf("%c", &ch);
        getchar(); //enter 키 입력 받음
        if (isalpha(ch))
            print2char(ch);
        else
            printf("입력: %c\n", ch);
    } while (ch != 'x' && ch != 'X'); //입력이 x 또는 X이면 종료

    return 0;
}

void print2char(char ch)
{
    if (isupper(ch))
        printf("입력: %c, 변환: %c\n", ch, tolower(ch));
    else
        printf("입력: %c, 변환: %c\n", ch, toupper(ch));

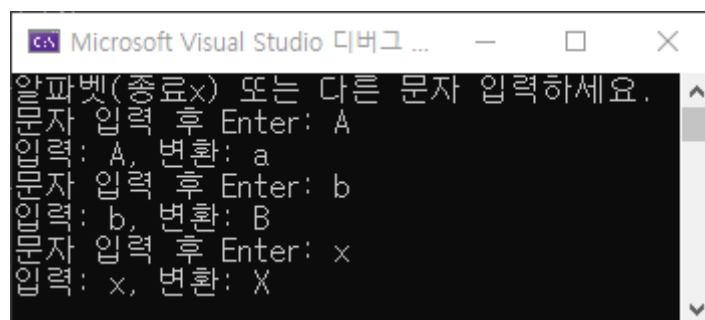
    return;
}
```

실습예제 10-9

파일 이름: char.c

교재: P483~P485

## 결과



```
Microsoft Visual Studio 디버그 ...
알파벳(종료x) 또는 다른 문자 입력하세요.
문자 입력 후 Enter: A
입력: A, 변환: a
문자 입력 후 Enter: b
입력: b, 변환: B
문자 입력 후 Enter: x
입력: x, 변환: X
```

## 예제 코드

```
// file: numberguess.c
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

#include <stdlib.h> //rand(), srand()를 위한 헤더파일 포함
#include <time.h> //time()을 위한 헤더파일 포함

#define MAX 100

int main(void)
{
    int guess, input;

    srand((long)time(NULL));
    guess = rand() % MAX + 1;

    printf("1에서 %d 사이에서 한 정수가 결정되었습니다.\n", MAX);
    printf("이 정수는 무엇일까요? 입력해 보세요. : ");

    while (scanf("%d", &input)) {
        if (input > guess)
            printf("입력한 수보다 작습니다. 다시 입력하세요. : ");
        else if (input < guess)
            printf("입력한 수보다 큼니다, 다시 입력하세요. : ");
        else
        {
            puts("정답입니다.");
            break;
        }
    }

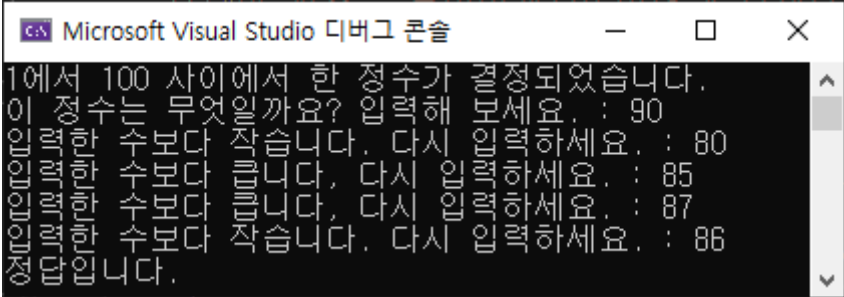
    return 0;
}
```

LAB 10-3

파일 이름: **numberguess.c**

교재 : P486~P487

## 결과



```
Microsoft Visual Studio 디버그 콘솔
1에서 100 사이에서 한 정수가 결정되었습니다.
이 정수는 무엇일까요? 입력해 보세요. : 90
입력한 수보다 작습니다. 다시 입력하세요. : 80
입력한 수보다 큼니다, 다시 입력하세요. : 85
입력한 수보다 큼니다, 다시 입력하세요. : 87
입력한 수보다 작습니다. 다시 입력하세요. : 86
정답입니다.
```