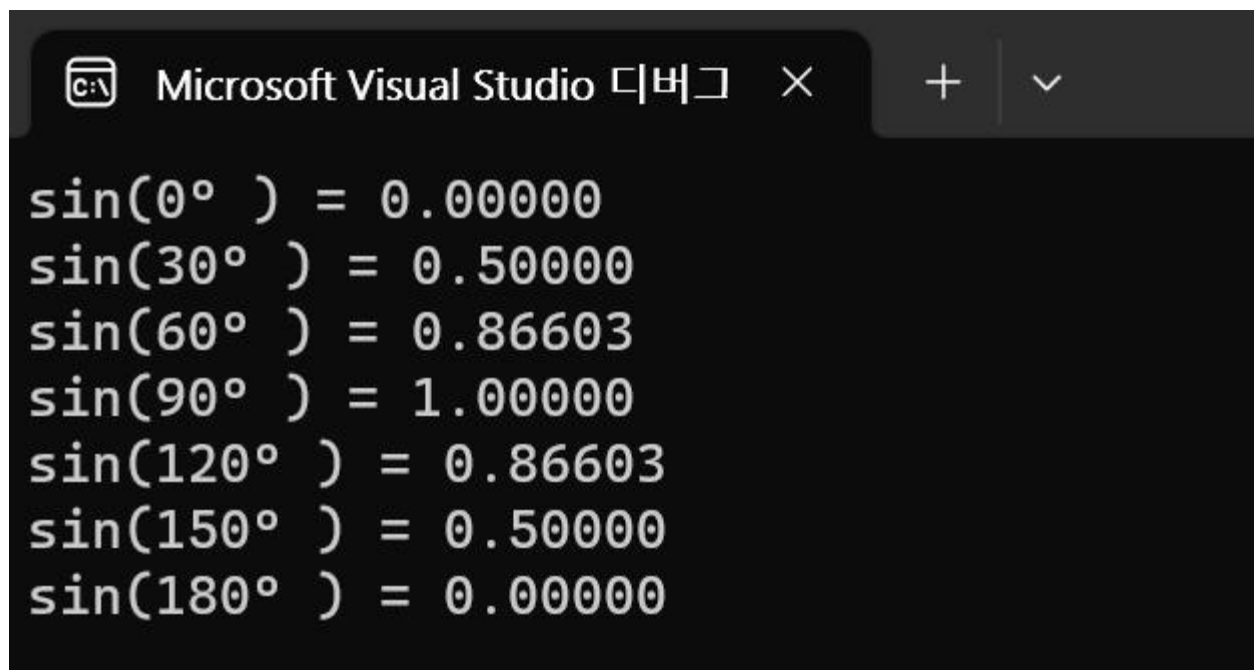


실습 1.

```
#include <stdio.h> // printf 함수를 위한 헤더 파일
#include <math.h>   // sin 함수를 위한 헤더 파일

#define Pi 3.141592
int main() {
    int degree; // 각도 값을 저장할 변수
    double radian; // degree의 라디안 값을 저장할 변수
    for (degree = 0; degree <= 180; degree += 30) {
        radian = (Pi * degree) / 180; // 각도를 라디안 단위로 변환
        printf("sin(%d°) = %.5lf \n", degree, sin(radian));
    }
    return 0;
}
```

실습 1 실행 화면.



The screenshot shows the Microsoft Visual Studio debugger window. The title bar reads "Microsoft Visual Studio 디버그". The output window displays the results of the program execution, showing the sine of angles from 0 to 180 degrees in increments of 30 degrees. The output is as follows:

Angle (degrees)	Sine Value
0°	0.00000
30°	0.50000
60°	0.86603
90°	1.00000
120°	0.86603
150°	0.50000
180°	0.00000

실습 2.

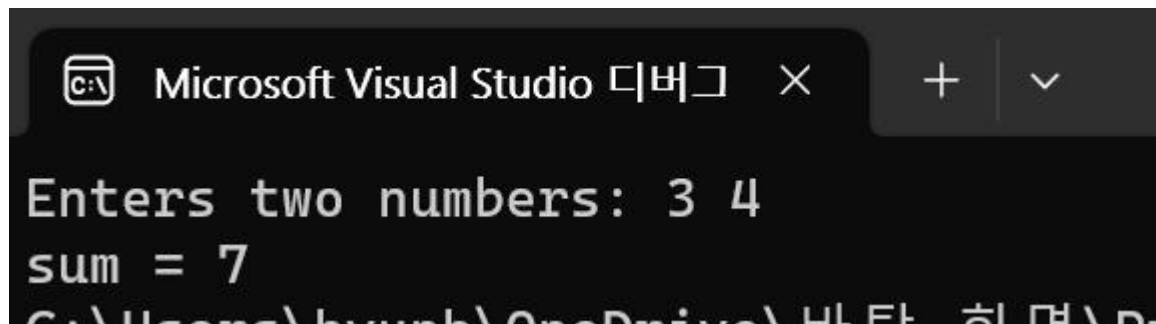
```
#include <stdio.h>
```

```
int addNumbers(int a, int b);           // function prototype
```

```
int main() {  
    int n1, n2, sum;  
    printf("Enters two numbers: ");  
    scanf("%d %d", &n1, &n2);  
  
    sum = addNumbers(n1, n2);           // function call  
    printf("sum = %d", sum);  
    return 0;  
}
```

```
int addNumbers(int a, int b) {         // function definition  
    int result;  
    result = a + b;  
    return result;                     // return statement  
}
```

실습 2 실행 화면.



실습 3.

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

```
/* function declaration */
```

```
void swap(int x, int y);
```

```
int main() {
```

```
    /* local variable definition */
```

```
    int a = 100;
```

```
    int b = 200;
```

```
    printf("Before swap, value of a : %d\n", a);
```

```
    printf("Before swap, value of b : %d\n", b);
```

```
    /* calling a function to swap the values*/
```

```
    swap(a, b);
```

```
    printf("After swap, value of a : %d \n", a);
```

```
    printf("After swap, value of b : %d \n", b);
```

```
    return 0;
```

```
}
```

```
void swap(int x, int y) {
```

```
    int temp;
```

```
    printf("[In SWAP Function] Before swap, value of x & y : %d, %d\n", x, y);
```

```
    temp = x; /* save the value of x*/
```

```
    x = y; /* put y into x */
```

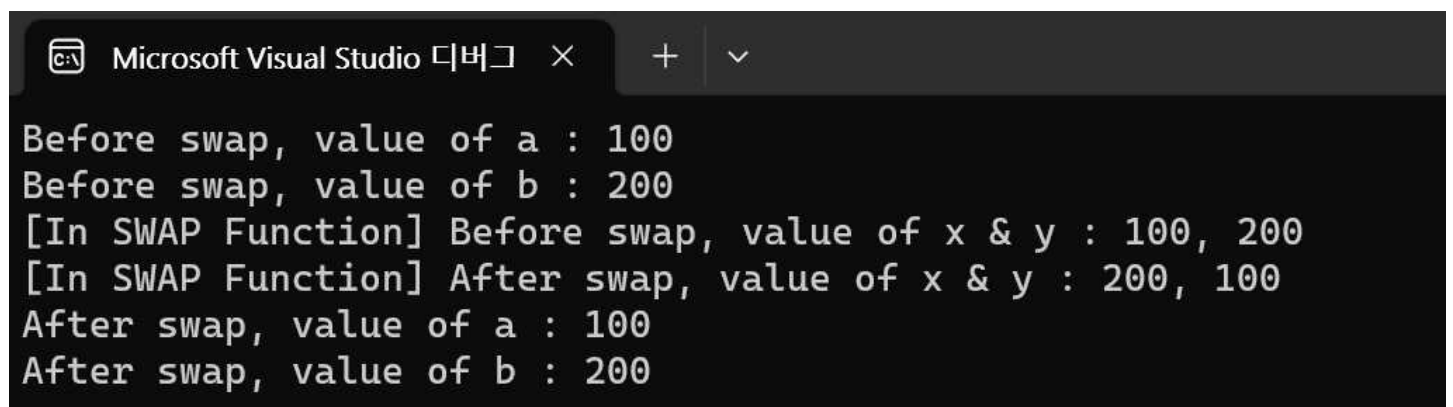
```
    y = temp; /* put temp into y*/
```

```
    printf("[In SWAP Function] After swap, value of x & y : %d, %d \n", x, y);
```

```
    return;
```

```
}
```

실습 3 실행 화면.



```
Microsoft Visual Studio 디버그 × + ∨  
Before swap, value of a : 100  
Before swap, value of b : 200  
[In SWAP Function] Before swap, value of x & y : 100, 200  
[In SWAP Function] After swap, value of x & y : 200, 100  
After swap, value of a : 100  
After swap, value of b : 200
```

실습 4.

#include <stdio.h>

/* function declaration */

void swap(int* x, int* y);

int main() {

/* local variable definition */

int a = 100;

int b = 200;

printf("Before swap, value of a : %d\n", a);

printf("Before swap, value of b : %d\n", b);

/* calling a function to swap the values*/

swap(&a, &b);

printf("After swap, value of a : %d \n", a);

printf("After swap, value of b : %d \n", b);

return 0;

}

void swap(int* x, int* y) {

int temp;

printf("[In SWAP Function] Before swap, value of x & y : %d, %d\n", *x, *y);

temp = *x; /* save the value of x*/

*x = *y; /* put y into x */

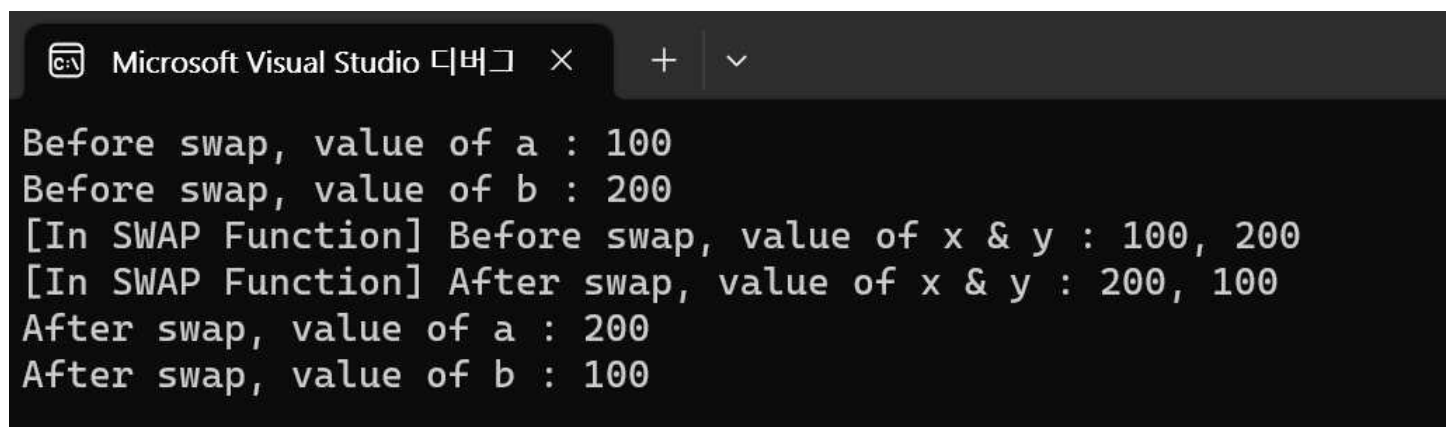
y = temp; / put temp into y*/

printf("[In SWAP Function] After swap, value of x & y : %d, %d \n", *x, *y);

return;

}

실습 4 실행 화면.



```

Microsoft Visual Studio 디버그
Before swap, value of a : 100
Before swap, value of b : 200
[In SWAP Function] Before swap, value of x & y : 100, 200
[In SWAP Function] After swap, value of x & y : 200, 100
After swap, value of a : 200
After swap, value of b : 100

```

실습 5.

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

```
double array_avg(int arr[], int n);    // 함수의 원형 선언
```

```
int main() {
```

```
    int pen[4] = { 4500, 5370, 4920, 6090 };
```

```
    int monthly_stock[12] = { 505, 409, 389, 257, 450, 501, 500, 621, 480, 350, 389, 250 };
```

```
    double average;
```

```
    average = array_avg(pen, 4);    // 연 평균 판매수 구하기
```

```
    printf("펜 평균 판매수: %.1lf \n", average);
```

```
    average = array_avg(monthly_stock, 12); // 연 평균 재고량 구하기
```

```
    printf("평균 재고량: %.1lf \n", average);
```

```
    return 0;
```

```
}
```

```
// 배열 원소가 n개 있는 arr 배열의 평균을 구하는 함수
```

```
double array_avg(int arr[], int n) {
```

```
    int i, sum = 0;
```

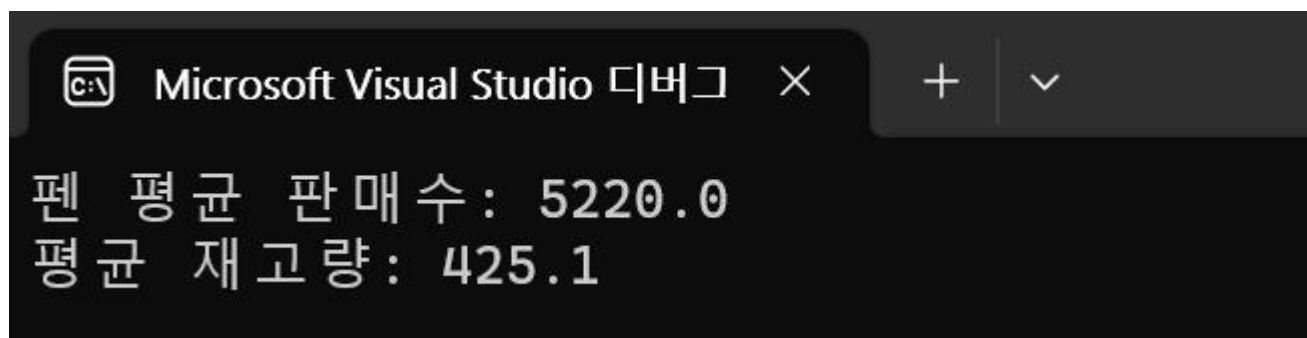
```
    for (i = 0; i < n; i++)
```

```
        sum = sum + arr[i];
```

```
    return (double)sum / n;
```

```
}
```

실습 5 실행 화면.



실습 6.

mymath.h 코드

```
#pragma once
```

```
int fact(int n);
```

```
int power(int base, int exp);
```

power.c 코드

```
#include "mymath.h"
```

```
int power(int base, int exp) {  
    int result = 1;  
    int i;  
    for (i = 1; i <= exp; ++i) {  
        result *= base;  
    }  
    return result;  
}
```

fact.c 코드

```
#include "mymath.h"
```

```
int fact(int n) {  
    if (n == 0) {  
        return 1;  
    }  
    else {  
        return n * fact(n - 1);  
    }  
}
```

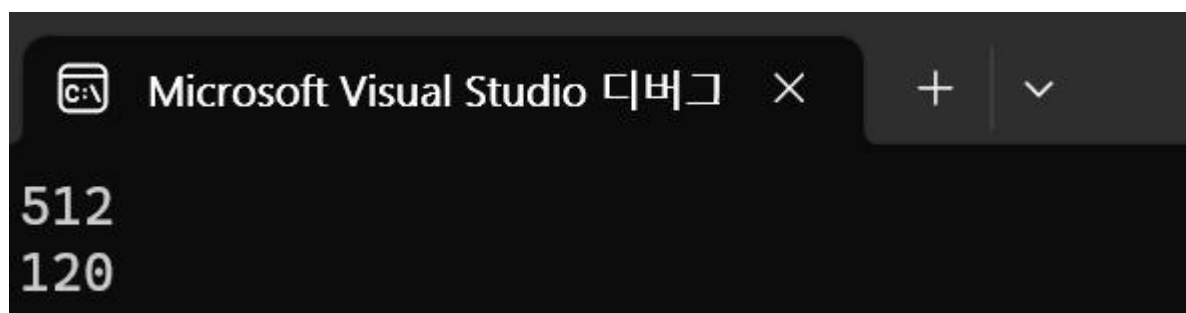
Lab08_6 코드

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

```
#include "mymath.h"
```

```
int main() {  
    printf("%d\n", power(2, 9));  
    printf("%d\n", fact(5));  
  
    return 0;  
}
```

실습 6 실행 화면.



실습 7.

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

```
int fact2(int n); /* Function Definition */
```

```
void main() {
```

```
    int num, res;
```

```
    printf("Enter positive integer: ");
```

```
    scanf("%d", &num);
```

```
    res = fact2(num); /* Normal Function Call */
```

```
    printf("%d! = %d", num, res);
```

```
}
```

```
int fact2(int n) { /* Function Definition */
```

```
    int f = 1;
```

```
    if (n <= 0) {
```

```
        return(1);
```

```
    }
```

```
    else {
```

```
        // Recursive Function Call as fact( ) calls itself
```

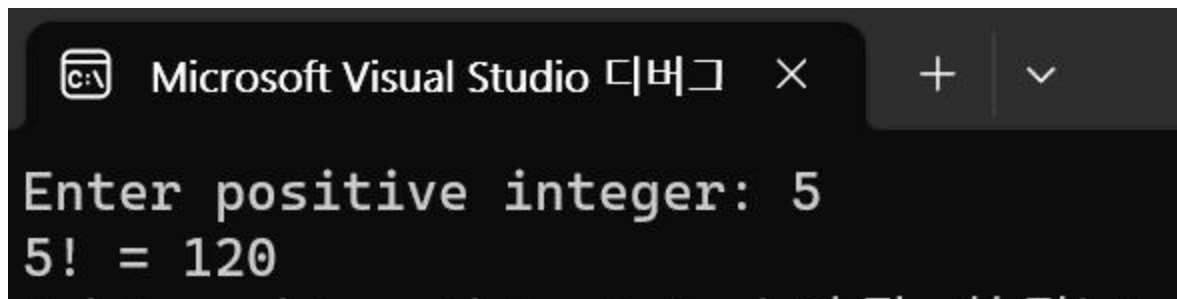
```
        f = n * fact2(n - 1);
```

```
        return(f);
```

```
    }
```

```
}
```

실습 7 실행 화면.



```
Microsoft Visual Studio 디버그 × + ▾  
Enter positive integer: 5  
5! = 120
```

실습 8.

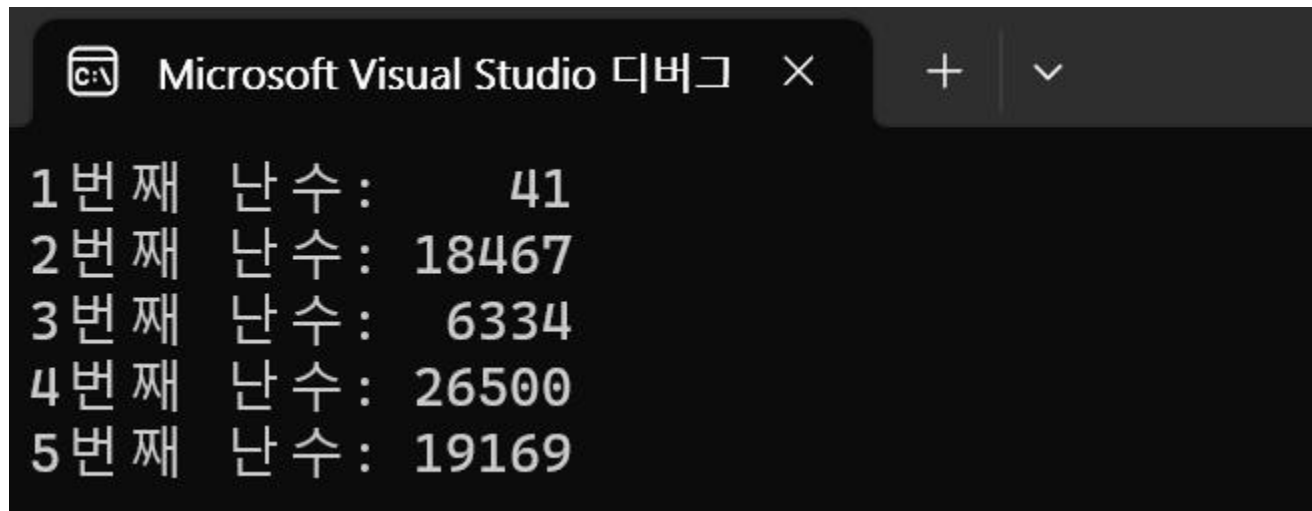
```
#include <stdio.h>
#include <stdlib.h> // rand 함수를 위한 헤더 파일

int main() {
    int i, random;

    for (i = 1; i <= 5; i++) {
        random = rand(); //난수를 얻어 random 변수에 저장
        printf("%d번째 난수: %5d \n", i, random);
    }

    return 0;
}
```

실습 8 실행 화면.



```
Microsoft Visual Studio 디버깅 × + ▾

1 번째 난수:    41
2 번째 난수: 18467
3 번째 난수:  6334
4 번째 난수: 26500
5 번째 난수: 19169
```

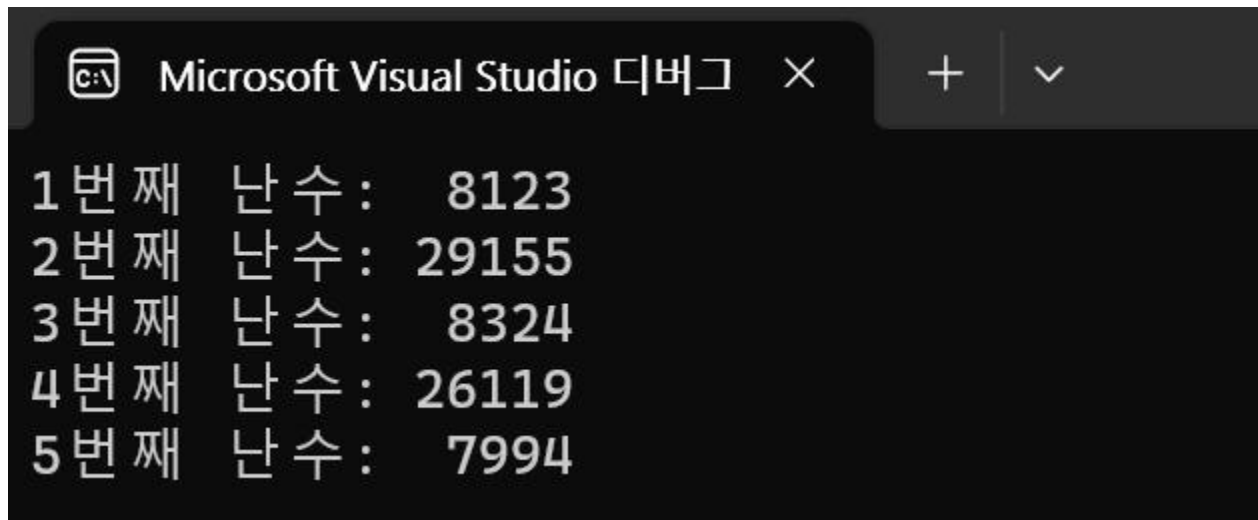

실습 9.

```
#include <stdio.h>
#include <stdlib.h>    // rand 함수를 위한 헤더 파일
#include <time.h>      // time 함수를 위한 헤더 파일

int main() {
    int i, random;

    srand(time(NULL));
    // time 함수를 실행할 때 컴퓨터의 현재 시간을 rand 함수의 씨드로 설정하기
    // srand(2);
    for (i = 1; i <= 5; i++) {
        random = rand();    // 난수를 얻어 random 변수에 저장
        printf("%d번째 난수: %5d \n", i, random);
    }
    return 0;
}
```

실습 9 실행 화면.



```
Microsoft Visual Studio 디버그 × + ▾

1번째 난수: 8123
2번째 난수: 29155
3번째 난수: 8324
4번째 난수: 26119
5번째 난수: 7994
```