**객체지향프로그래밍 LAB #03&04**

**<기초문제>\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1.

#include <iostream>

using namespace std;

int square(int a);

int main() {

int x;

x = 5;

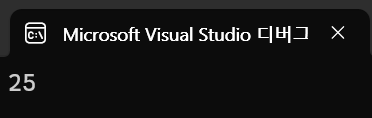
cout << square(x) << endl; // square 함수 사용

}

int square(int a) {

return a \* a;

}



2.

#include <iostream>

using namespace std;

int get\_num();

int myadd(int x, int y);

int mysub(int x, int y);

int mymul(int x, int y);

float mydiv(int x, int y);

int main() {

int x, y;

x = get\_num(); //키보드로부터 값 입력

y = get\_num();

cout << "x = " << x << ", y = " << y << endl;

cout << "x + y = " << myadd(x,y) << endl;

cout << "x - y = " << mysub(x,y) << endl;

cout << "x \* y = " << mymul(x,y) << endl;

cout << "x / y = " << mydiv(x,y) << endl;

return 0;

}

int get\_num() {

int num;

cout << "Enter a number: ";

cin >> num;

return num;

}

int myadd(int x, int y) {

return x + y;

}

int mysub(int x, int y) {

return x - y;

}

int mymul(int x, int y) {

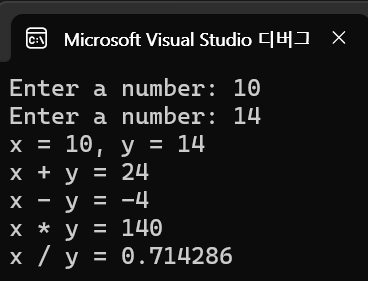
return x \* y;

}

float mydiv(int x, int y) {

return float(x) / float(y); //소수점 출력되도록

}



3.

#include <iostream>

#include <cmath>

using namespace std;

int main() {

double value;

value = 5;

cout << sqrt(value) << endl; //제곱근

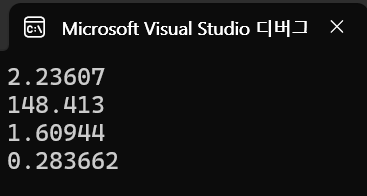
cout << exp(value) << endl; //e의 지수 값 연산

cout << log(value) << endl; //로그연산(밑:자연상수e)

cout << cos(value) << endl; //cosine

return 0;

}



4.

#include <iostream>

using namespace std;

void applePrice(int a = 1000) {

cout << "Price of an apple is " << a << endl;

}

// default augment 중요!

int main() {

applePrice(1500);

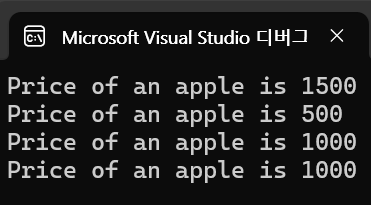
applePrice(500);

applePrice(1000);

applePrice(); // default augments

return 0;

}



5.

#include <iostream>

using namespace std;

int get();

int myadd(int x, int y);

int mysub(int x, int y);

int mymul(int x, int y);

float mydiv(int x, int y);

int addmul(int x, int y, int z); // return value: (x + y) \* z

float muldiv(int x, int y, int z); // return value: (x \* y) / z

int addmuladd(int x, int y, int z); // return value: (x + y) \* (y + z)

float subdivsub(int x, int y, int z); // return value: (x - y) / (y - z)

int main() {

int x, y, z;

x = get();

y = get();

z = get();

cout << "x = " << x << ", y = " << y << ", z = " << z << endl;

cout << "(x + y) \* z = " << addmul(x, y, z) << endl;

cout << "(x \* y) / z = " << muldiv(x, y, z) << endl;

cout << "(x + y) \* (y + z) = " << addmuladd(x, y, z) << endl;

cout << "(x - y) / (y - z) = " << subdivsub(x, y, z) << endl;

}

int get() {

int n;

cout << "Enter a number: ";

cin >> n;

return n;

}

int myadd(int x, int y) {

return x + y;

}

int mysub(int x, int y) {

return x - y;

}

int mymul(int x, int y) {

return x \* y;

}

float mydiv(int x, int y) {

return float(x) / float(y); //소수점 출력되도록

}

int addmul(int x, int y, int z) {

return myadd(x, y) \* z; // return value: (x + y) \* z

}

float muldiv(int x, int y, int z) {

return mydiv(mymul(x, y), z); // return value: (x \* y) / z

}

int addmuladd(int x, int y, int z) {

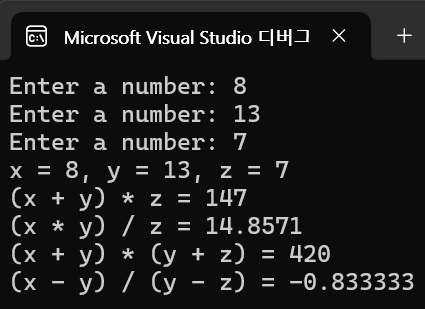
return myadd(x, y) \* myadd(y, z); // return value: (x + y) \* (y + z)

}

float subdivsub(int x, int y, int z) {

return mydiv(mysub(x, y), mysub(y, z)); // return value: (x - y) / (y - z)

}



6.

#include <iostream>

using namespace std;

void print\_DOB(int year = 1900, int month = 1, int day = 1) {

cout << "생년월일은 " << year << "년 " << month << "월 " << day << "일입니다." << endl;

}

int main() {

print\_DOB();

int year, month, day;

cout << "year 입력 : ";

cin >> year;

cout << "month 입력 : ";

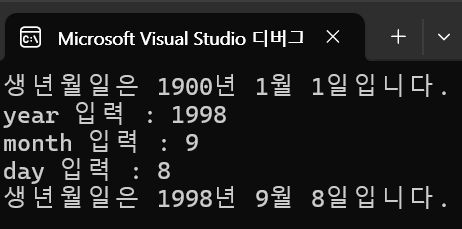
cin >> month;

cout << "day 입력 : ";

cin >> day;

print\_DOB(year, month, day);

}



7.

#include <iostream>

#include <cmath>

using namespace std;

int main() {

double x, y;

cout << "x = ";

cin >> x;

cout << "y = ";

cin >> y;

cout << "Rounding up number of " << x << ": " << ceil(x)<< endl;

cout << "Rounding down number of " << x << ": " << floor(x)<< endl;

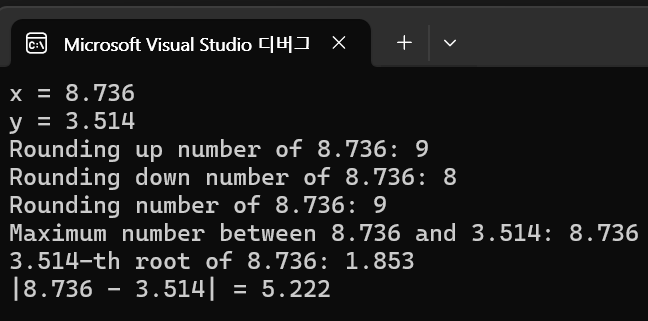
cout << "Rounding number of " << x << ": " << round(x)<< endl;

cout << "Maximum number between " << x << " and " << y << ": " << max(x, y) << endl;

cout << y << "-th root of " << x << ": " << pow(x, 1/y) << endl;

cout << '|' << x << " - " << y << "| = " << abs(x - y) << endl;

}



8.

#include <iostream>

using namespace std;

void get\_data(int& x, int& y) {

cout << "x 입력 : ";

cin >> x;

cout << "y 입력 : ";

cin >> y;

cout << endl;

}

void swap\_call\_by\_value(int a,int b) {

int temp = a;

a = b;

b = temp;

}

void swap\_call\_by\_reference(int& a, int& b) {

int temp = a;

a = b;

b = temp;

}

int main() {

int x, y;

get\_data(x, y);

cout << "swap\_call\_by\_value 함수 사용 전" << endl;

cout << "x = " << x << ", y = " << y << endl;

swap\_call\_by\_value(x, y);

cout << "swap\_call\_by\_value 함수 사용 후" << endl;

cout << "x = " << x << ", y = " << y << endl;

cout << endl;

cout << "swap\_call\_by\_reference 함수 사용 전" << endl;

cout << "x = " << x << ", y = " << y << endl;

swap\_call\_by\_reference(x, y);

cout << "swap\_call\_by\_reference 함수 사용 후" << endl;

cout << "x = " << x << ", y = " << y << endl;

}

