|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **C1\_실습과제\_11주** | **이름 :** | **정균모** | **학번 :** | **20145165** |

* **강의 내용**

1. 오른쪽 직각 삼각형

printf("\n====>오른쪽 직각 삼각형\n");

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

for(j=col-(i+1);j<col;j++)

{

dim[i][j]=k++;

}

}

1. ‘ㄹ’자로 채우기

printf("\n=====>ㄹ자로 채우기\n");

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

if(sw==0){

for(j=0;j<col;j++){

dim[i][j]=k++;

}

sw=1;

}

else{

for(j=col-1;j>=0;j--){

dim[i][j]=k++;

}

sw=0;

}

}

1. 포인터 변수 선언 및 초기화

#include <stdio.h>

int main(void){

int data = 100;

char ch = 'A';

int \*ptrint = &data; //포인터 변수 선언 및 초기화

char \*ptrchar = &ch;

printf("간접 참조 출력: %d %c\n", \*ptrint, \*ptrchar);

printf("주소 출력: ptrint=%#p ptrchar=%#p\n", ptrint, &data);

\*ptrint = 200;

\*ptrchar = 'B';

printf("직접 참조 출력: %d %c\n", data, ch);

return 0;

}

1. 일차원 배열과 포인터

#include <stdio.h>

int main(void){

int a[4] = {1, 3, 6, 8};

int \*pa = &a[0]; //\*pa=a; 배열의 시작 주소를 포인터 변수로 초기화

printf("%d %d %d %d\n", \*(pa), \*(pa+1), \*(pa+2), \*(pa+3));

printf("%d %d %d %d\n", pa[0], pa[1], pa[2], pa[3]);

printf("%d ", ++\*pa); //++(\*pa)

printf("%d ", \*++pa); //\*(++pa)

printf("%d ", \*pa++); //\*(pa++)

printf("%d ", --\*pa++); //--(\*(pa++))

printf("%d\n", (\*pa)--);

printf("%d %d %d %d\n", a[0], a[1], a[2], a[3]);

return 0;

}

1. 포인터와 함수

#include <stdio.h>

#define size 4

void dim1\_display(int \*, int);

void dim2\_display(int (\*)[size], int);

int main(void){

int a[] = {8, 2, 8, 1, 3};

int ary[][size] = {5, 7, 6, 2, 7, 8, 1, 3};

dim1\_display(a, sizeof(a)/sizeof(int));

dim2\_display(ary, sizeof(ary)/sizeof(ary[0]));

return 0;}

void dim1\_display(int \*p, int n){

int i;

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

//printf("\*(p + %d) = %d \n", i,\*(p+i));

printf("%d=%d\n", i, \*p++);

}

void dim2\_display(int (\*ptr)[size], int n){

int i, j;

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

for(j=0; j<size;j++)

//printf("\*(ptr[%d] + %d) = %d\n",i, j, \*(ptr[i]+j));

printf("\*(\*(ptr +%d) + %d) = %d\n",i, j, \*(\*(ptr+i)+j));

}

}

❑ 내용 점검

1. 다음에서 서술 내용이 맞으면 0, 틀리면 x 하시오.
2. 어느 변수의 주소 값을 저장하려면 반드시 그 변수의 자료유형과 동일한 포인터 변수에 저장해야 한다. ( O )
3. 포인터 pa에 &a[0]를 저장하면, 연산식 (\*pa+i)로 배열원소를 참조할 수 있다. ( O )
4. 포인터 변수도 자동으로 형변환(cast)이 가능하다. ( X )
5. 연산식 ++\*p는 ++(\*p)으로 포인터 p가 가리키는 값을 1 증가시킨 후 참조한다. ( O )
6. 이차원 배열에서 배열이름인 td는 포인터 상수 td[0]를 가리키는 포인터 상수이다. ( O )
7. 간접연산자 \*는 후위 증감연산자 a++보다 우선순위가 높다. ( O )
8. 일차원 배열 score에서 배열이름 score 자체가 배열 첫 원소의 주소 값이다. ( X )
9. 메모리 공간은 바이트마다 고유한 주소(address)가 있다. ( O )
10. 다음에서 비어있는 부분을 적당히 채우시오.
11. \_\_\_\_\_\_\_\_\_\_&\_\_\_\_\_\_\_\_\_\_는 피연산자인 변수의 메모리 주소를 반환하는 주소연산자이다.
12. \_\_\_\_\_\_\_NULL\_\_\_\_\_\_\_\_\_\_은 헤더파일 stdio.h에 다음과 같이 정의되어 있는 포인터 상수로서 0번지의 주소 값을 의미한다.
13. 포인터 변수가 가리키고 있는 변수를 참조하려면 간접연산자(indirection operator) \_\_\_\_\*\_\_\_\_\_를 사용한다.
14. int형 포인터 pi에 저장된 주소 값이 100이라면 (pi+1)의 주소 값은 \_\_\_\_\_\_\_\_104\_\_\_\_\_\_\_\_\_\_이다.
15. 다음 각각의 문제에서 가장 적절한 것을 하나 선택하시오.
16. 일차원 배열선언 int n[5] = {1, 2, 3, 4, 5}; 문장 이후 사용한 문장이다. 다음 중에서 잘못된 연산식이나 문장은 무엇인가? ( 가,다 )
17. n++
18. n+1
19. \*(n+1)
20. int \*p = n;
21. 배열선언 double a[] = {2, 4, 5, 7, 8, 9};에서 \*a와 \*(a+2)의 참조 값은 각각 무엇인가?

\*a = 0;

\*a = 0;

1. 다음 설명에 대한 변수선언 문장을 작성하시오.
2. float 형 포인터 pf // float \*pf;
3. double 형 이중 포인터 dpd //double dpd;
4. 모든 자료형의 대한 포인터 p; // void \*p;
5. int 형 포인터 3개인 배열 pa // int \*ps[3];
6. 열이 5인 double 형 이차원 배열을 위한 포인터 pd //double (\*pd)[5];
7. short 형 포인터 ps //short \*ps;
8. float 형 포인터 pf // float \*pf;
9. int 형 포인터 addn //int \*addn;
10. 다음 문장에서 문법오류 및 실행오류를 찾아 수정하시오.
11. double d; int \*p = &d; //

double d;

double \*p = &d;

1. int i; int \*pi = &i; int \*\*dpi = &i; //

int i;

int \*pi = &i;

int \*\*dpi = &pi;

1. int a, b, c; int \*pf[3] = {a, b, c}; //

int a, b, c;

int \*pf[3] = {&a, &b, &c};

1. char ch[5]; char \*pc = &ch; //

char ch[5];

char \*pc = &ch[0];

1. double data[2][3]; double (\*pd)[2] = data; //

double data[2][3];

double (\*pd)[3] = data;

1. 다음 일차원 배열과 포인터 선언 문장에 대하여 다음 각각의 연산식 결과를 기술하시오.  
    int score[] = {3, 5, 6, 9, 2, 5, 7, 8, 9, 1};

int \*p = score;

int \*q = p + 4;

1. \*p // 3
2. \*p+1 // 4
3. \*(p+1) // 5
4. ++\*p // 4
5. \*++p // 5
6. \*q // 2
7. q[2] // 7
8. --\*q // 1
9. (\*(q+1))++ // 5
10. \*(q+4) // 9
11. 다음 일차원 배열과 포인터 선언 문장에 대하여 다음 각각의 연산식 결과를 기술하시오.

int score[][3] = {3, 5, 6, 9, 2, 5, 7, 8, 9, 1};

int (\*ps)[3] = score;

int \*p = score[0];

int \*q = score[1];

1. \*ps // 1636164
2. ps[2][2] // 9
3. \*\*score // 3
4. \*p // 3
5. \*q // 9
6. \*score[2] // 7
7. \*(score[2]+1) // 1636192
8. \*(\*(score+1)+1) // 2
9. \*(\*score+5) // 5
10. (\*(score+3))[2] // 0

❑ 프로그래밍 과제

* 예비문제 1. 정수형 변수에 사용자가 입력한 정수를 저장하고 포인터를 이용하여 출력하시오.

int p;

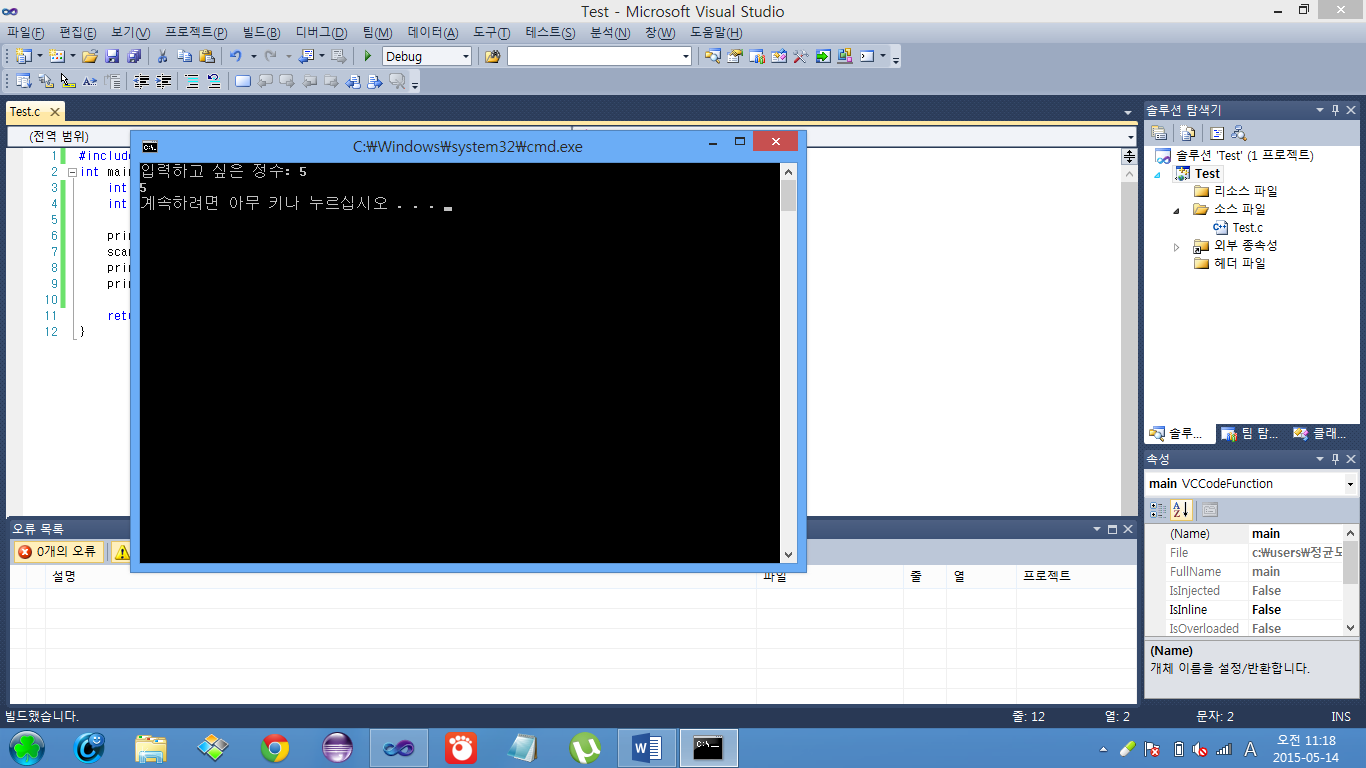
int \*print;

printf("입력하고 싶은 정수: ");

scanf("%d", &p);

print = &p;

printf("%d\n", \*print);



* 예비문제 2. 문자형 변수에 사용자가 입력한 문자를 저장하고 포인터를 이용하여 출력하시오.

char p;

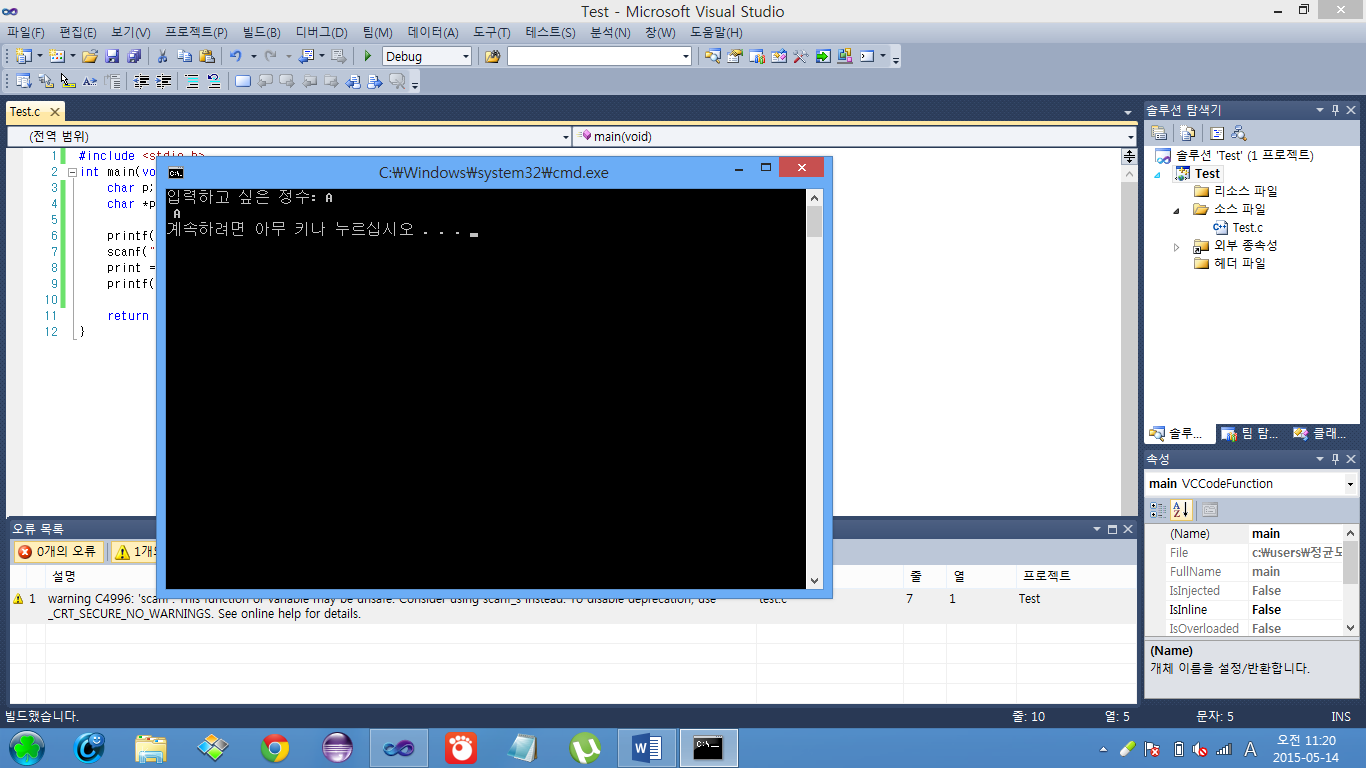
char \*print;

printf("입력하고 싶은 정수: ");

scanf(" %c", &p);

print = &p;

printf(" %c\n", \*print);



* 예비문제 3. 정수 배열에 사용자가 입력한 정수 다섯 개를 저장하고 포인터를 이용하여 출력하시오.

int arr[5],i,j;

int \*ptrint;

for(i=0;i<5;i++){

printf("입력하고 싶은 수: ");

scanf("%d", &arr[i]);

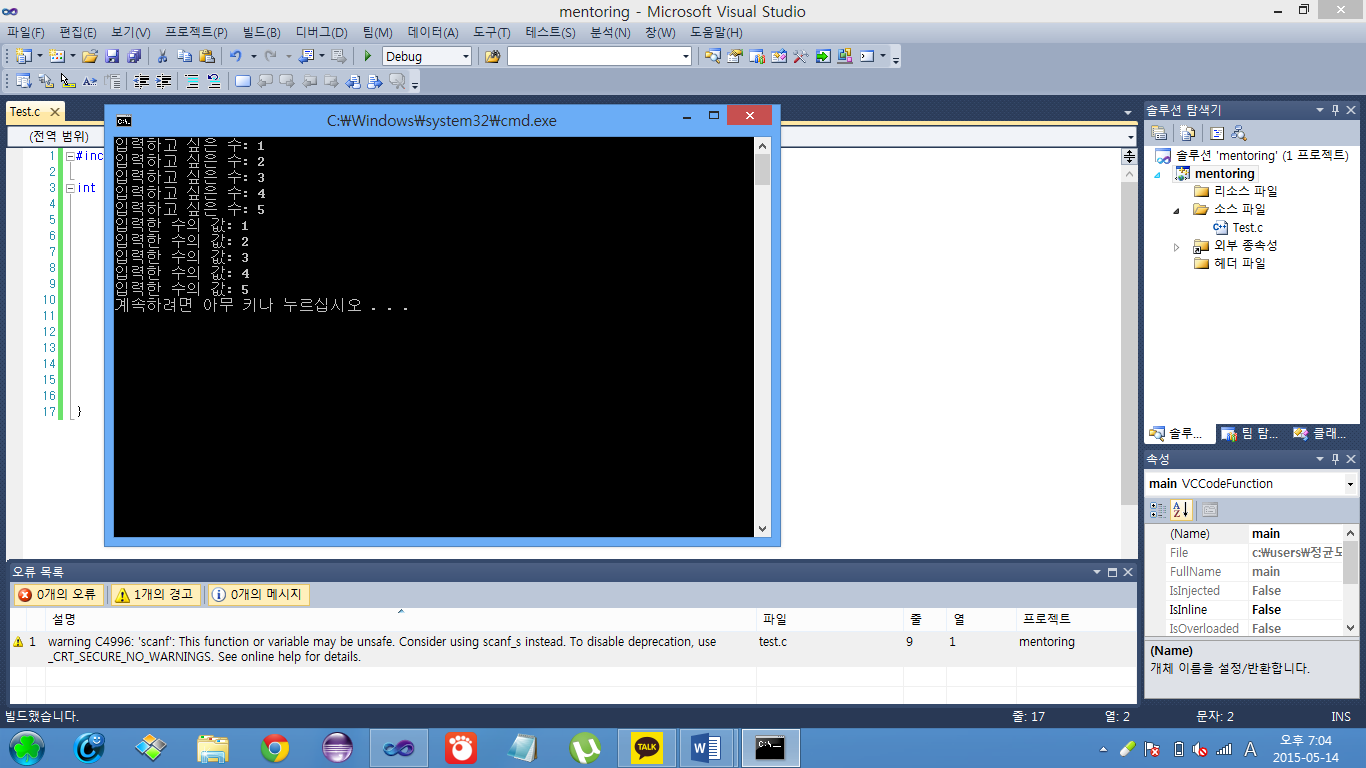
}

ptrint = arr;

for(j=0;j<5;j++){

printf("입력한 수의 값: %d\n",\*(ptrint+j));

}



* 예비문제 4. 문자 배열에 사용자가 입력한 문자 다섯 개를 저장하고 포인터를 이용하여 출력하시오.

char arr[5],i,j;

char \*ptrint;

for(i=0;i<5;i++){

printf("입력하고 싶은 문자: ");

scanf(" %c", &arr[i]);

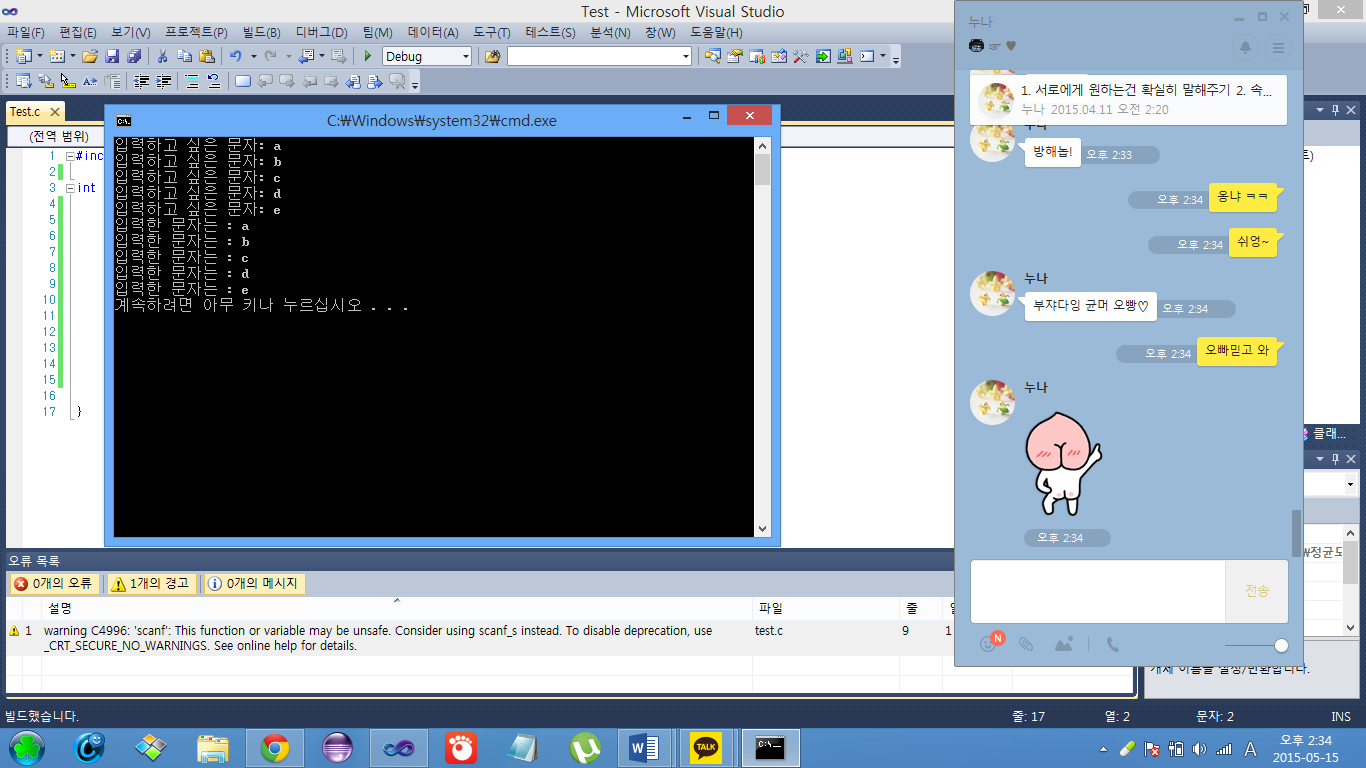
}

ptrint = arr;

for(j=0;j<5;j++){

printf("입력한 문자는 : %c\n",\*(ptrint+j));

}



* 예비문제 5. 3행3열 이차원 실수 배열에 사용자가 입력한 실수를 저장하고 포인터를 이용하여 출력하시오.

double a[3][3];

int i, j;

double (\*b)[3] = {NULL};

printf("실수 입력:");

for(j=0; j<3; j++){

for(i=0; i<3; i++){

scanf("%lf",&a[j][i]);

}

}

b = a;

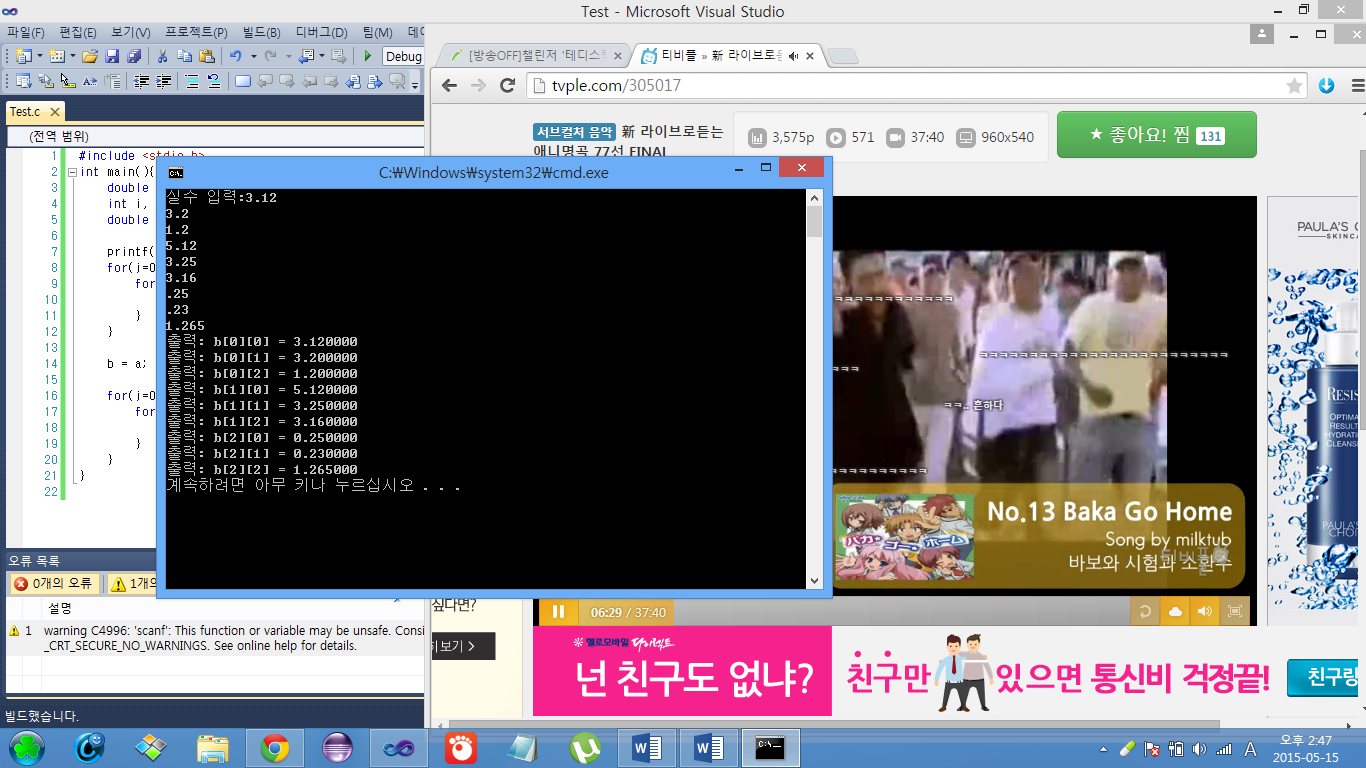
for(j=0; j<3; j++){

for(i=0; i<3; i++){

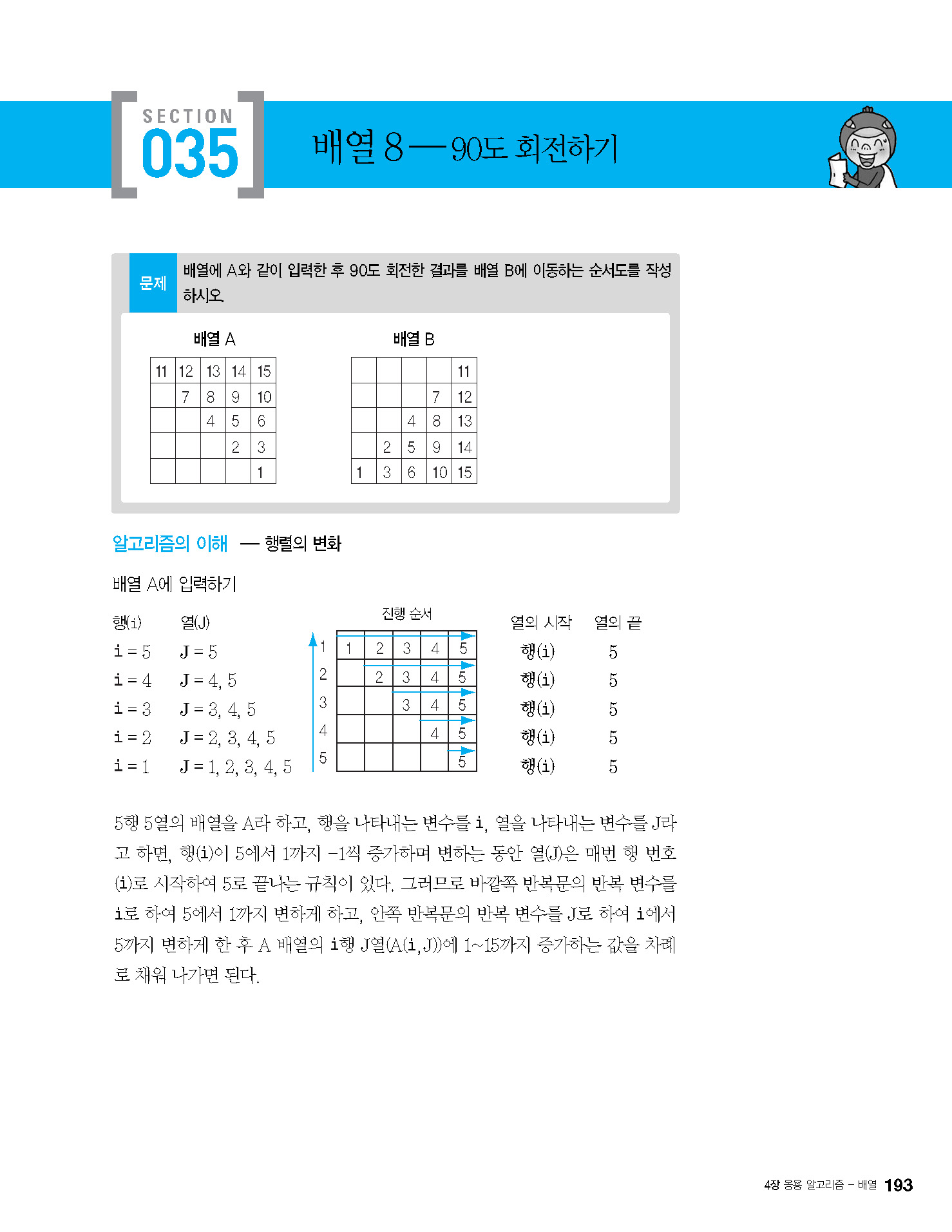
printf("출력: b[%d][%d] = %lf\n", j, i, \*(b[j]+i));

}

}



1. 배열 A에 그림과 같이 초기화 한 후 90도 회전한 결과를 배열 B에 이동하는 프로그램을 작성하시오



#include <stdio.h>

int main(void){

int arr[5][5] = {{11,12,13,14,15}, {0,7,8,9,10},{0,0,4,5,6},{0,0,0,2,3},{0,0,0,0,1}};

int arr1[5][5];

int i, j;

int a=0,b=0;

for(i=0; i<5; i++){

for(j=0; j<5; j++){

printf("%3d",arr[i][j]);

}

printf("\n");

}

printf("\n");

for(i=0;i<5;i++){

for(j=5;j>0;j--){

arr1[a][b]=arr[j-1][i];

printf("%3d " ,arr1[a][b]);

b++;

}

b=0;

a++;

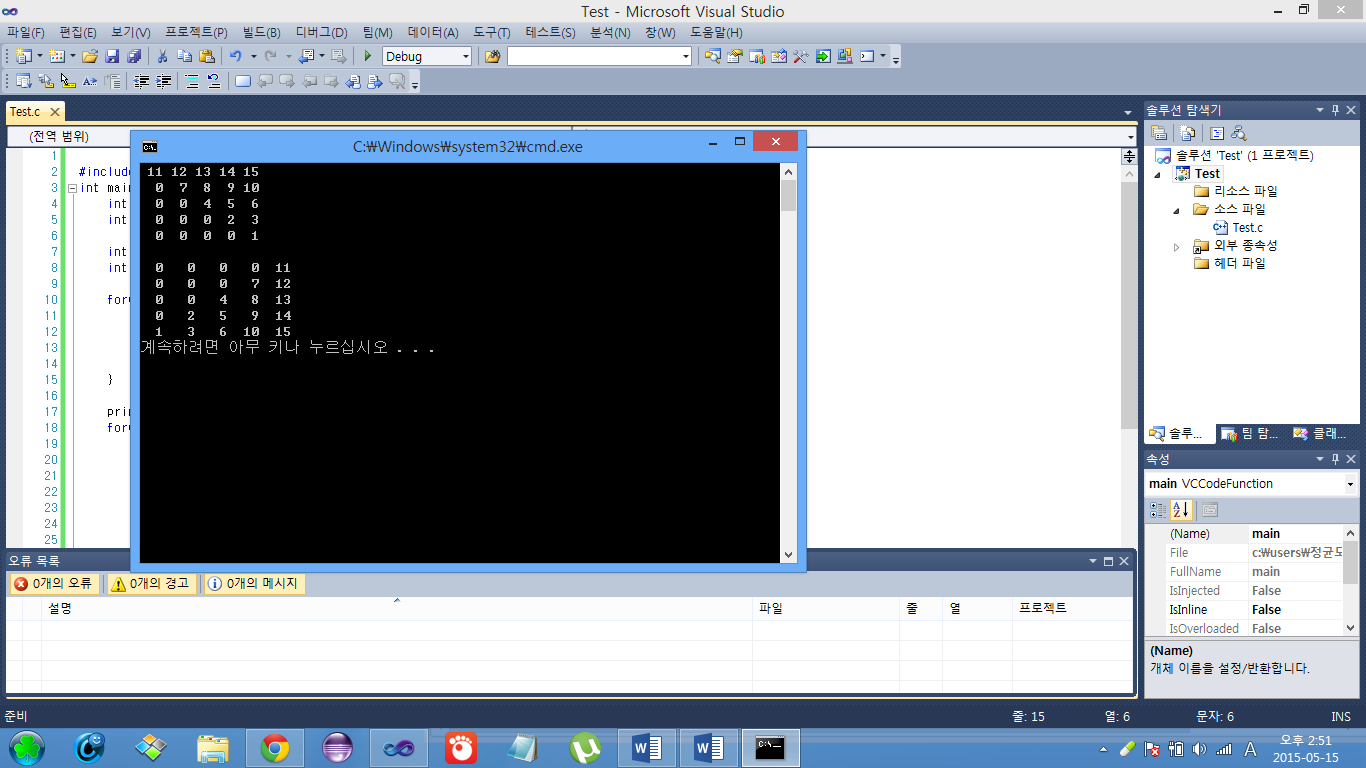
printf("\n");

}

return 0;

return 0;

}



1. 자료유형 double형 1차원 배열을 다음과 같이 초기화하고, 첫 번째 인자인 배열 source을 두 번째 인자인 배열 target에 복사하는 함수를 만들어 결과를 알아보는 프로그램을 포인터를 이용하여 작성하시오.

* double ary[5] = {3.12, 5.14, 7.25, 7.48, 5.91};
* void copyarray(double \*source, double \*target, int size);
* void print(double \*x, int size)

#include <stdio.h>

void copyarray(double \*source, double \*target, int size);

void print(double \*x, int size);

int main(void){

int i;

double ary1[5] = {3.12, 5.14, 7.25, 7.48, 5.91};

double ary2[5] = {0};

int size = sizeof(ary1)/sizeof(ary1[0]);

copyarray(ary1, ary2, size);

print(ary2, size);

}

void copyarray(double \*source, double \*target, int size){

int i;

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

target[i] = source[i];

}

}

void print(double \*x, int size){

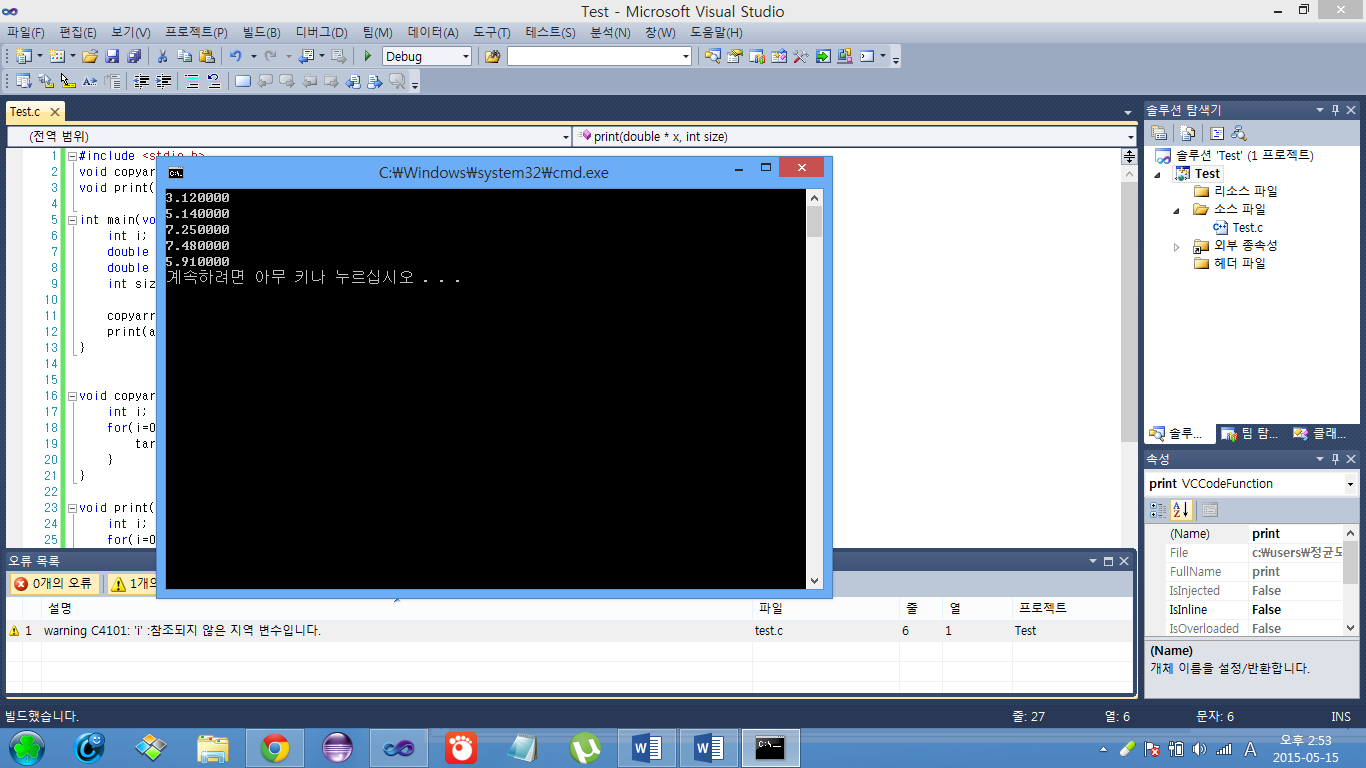
int i;

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

printf("%lf\n",x[i]);

}

}



1. 배열에서 모든 원소의 값을 모두 n씩 증가시키는 프로그램을 포인터를 이용하여 작성하시오.

* 배열 int data[] = {3, 21, 35, 57, 24, 82, 8};, n은 표준입력으로 처리
* 함수 increment()는 배열 p에서 배열크기 size만큼 모든 원소의 값을 모두 n씩 증가시키는 함수: void increment(int \*p, int size, int n)
* 배열의 모든 원소를 출력하는 함수 : void print(int \*pi, int size)

#include <stdio.h>

void increment(int \*p, int size, int n);

void print(int \*pi, int size);

int main(void){

int data[] = {3, 21, 35, 57, 24, 82, 8};

int size = sizeof(data)/sizeof(data[0]);

int n, i;

printf("원래의 값:");

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

printf("%d ",data[i]);

}

printf("\n몇 씩 증가시킬까요??");

scanf("%d",&n);

increment(data, size, n);

print(data, size);

}

void increment(int \*p, int size, int n){

int i;

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

p[i] = p[i]+n;

}

}

void print(int \*pi, int size){

int i;

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

printf("%d ",pi[i]);

}

}

