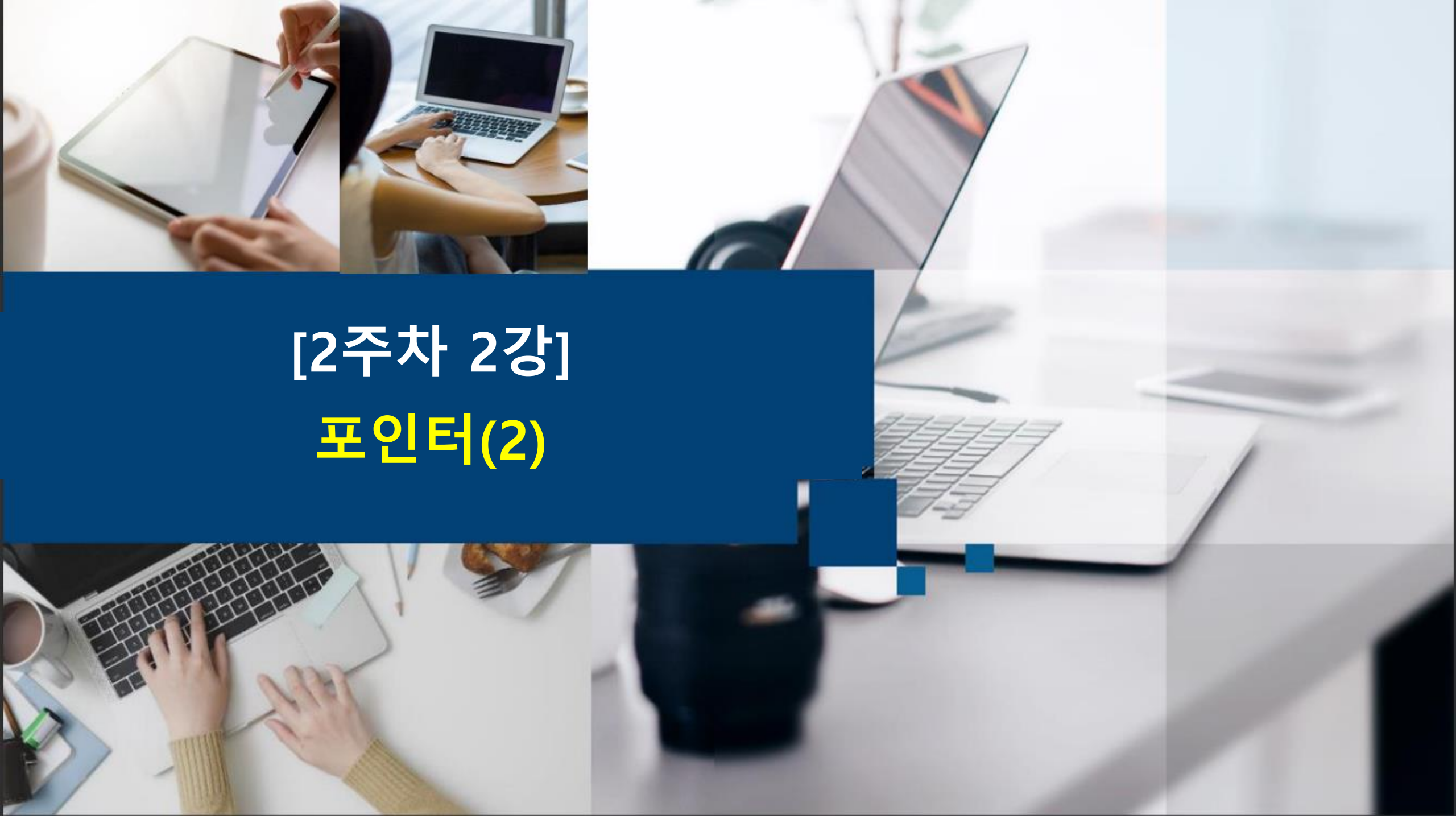
C프로그래밍및실습

담당교수:최희식
dali3054@ssu.ac.kr



[2주차 2강]

포인터(2)

1

포인터 연산

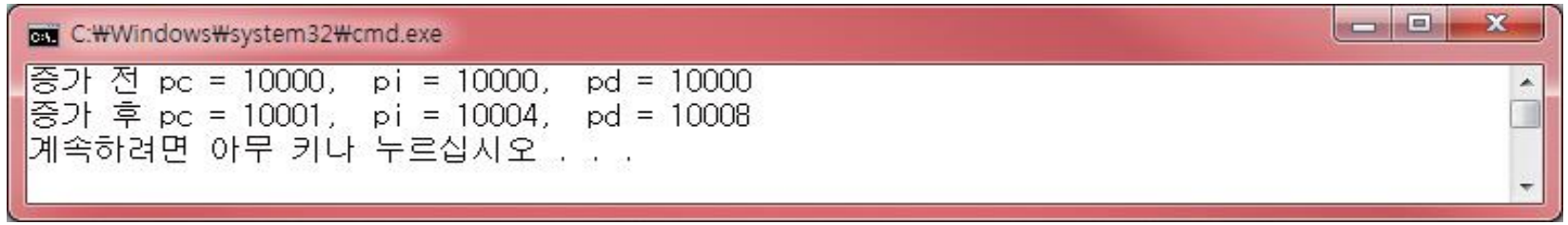
- 가능한 연산: 증가, 감소, 덧셈, 뺄셈 연산
- 증가 연산의 경우 증가되는 값은 포인터가 가리키는 객체의 크기

$$++p;$$

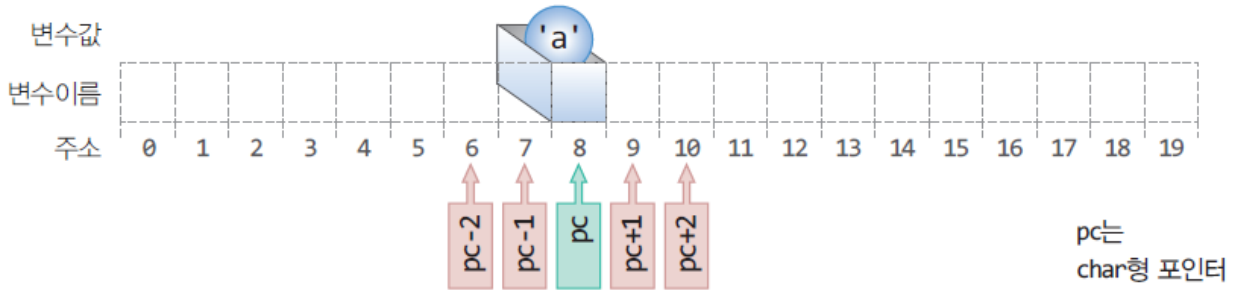
포인터 타입	++연산후 증가되는값
char	1
short	2
int	4
float	4
double	8

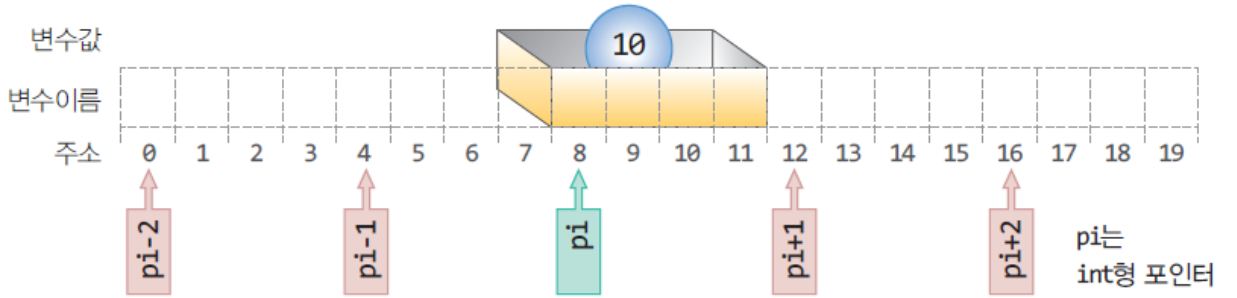
증가 연산 예제 #1

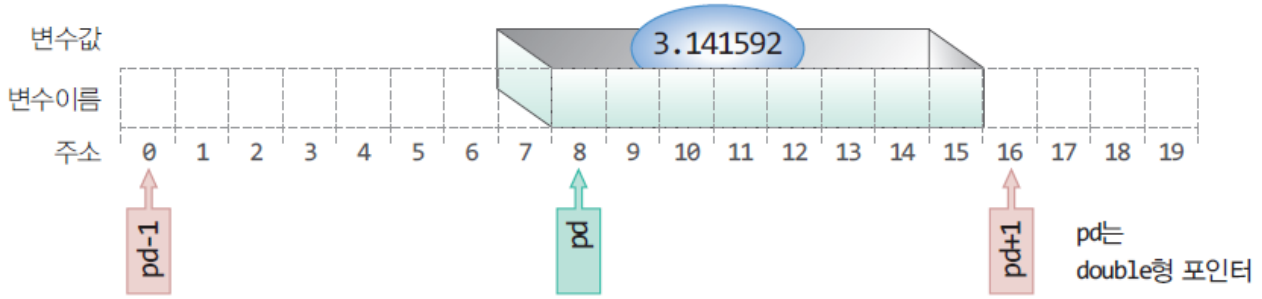
```
#include <stdio.h>
int main(void)
{
        char *pc;
        int *pi;
        double *pd;

        pc = (char *)10000;
        pi = (int *)10000;
        pd = (double *)10000;
        printf("증가 전 pc = %d,  pi = %d,  pd = %d₩n", pc, pi, pd);

        pc++;
        pi++;
        pd++;
        printf("증가 후 pc = %d,  pi = %d,  pd = %d₩n", pc, pi, pd);

        return 0;
}
```



포인터의 증감 연산


변수값
변수이름
주소
'a'
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
pc-2
pc-1
pc
pc+1
pc+2
pc는
char형 포인터


변수값
변수이름
주소
10
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
pi-2
pi-1
pi
pi+1
pi+2
pi는
int형 포인터

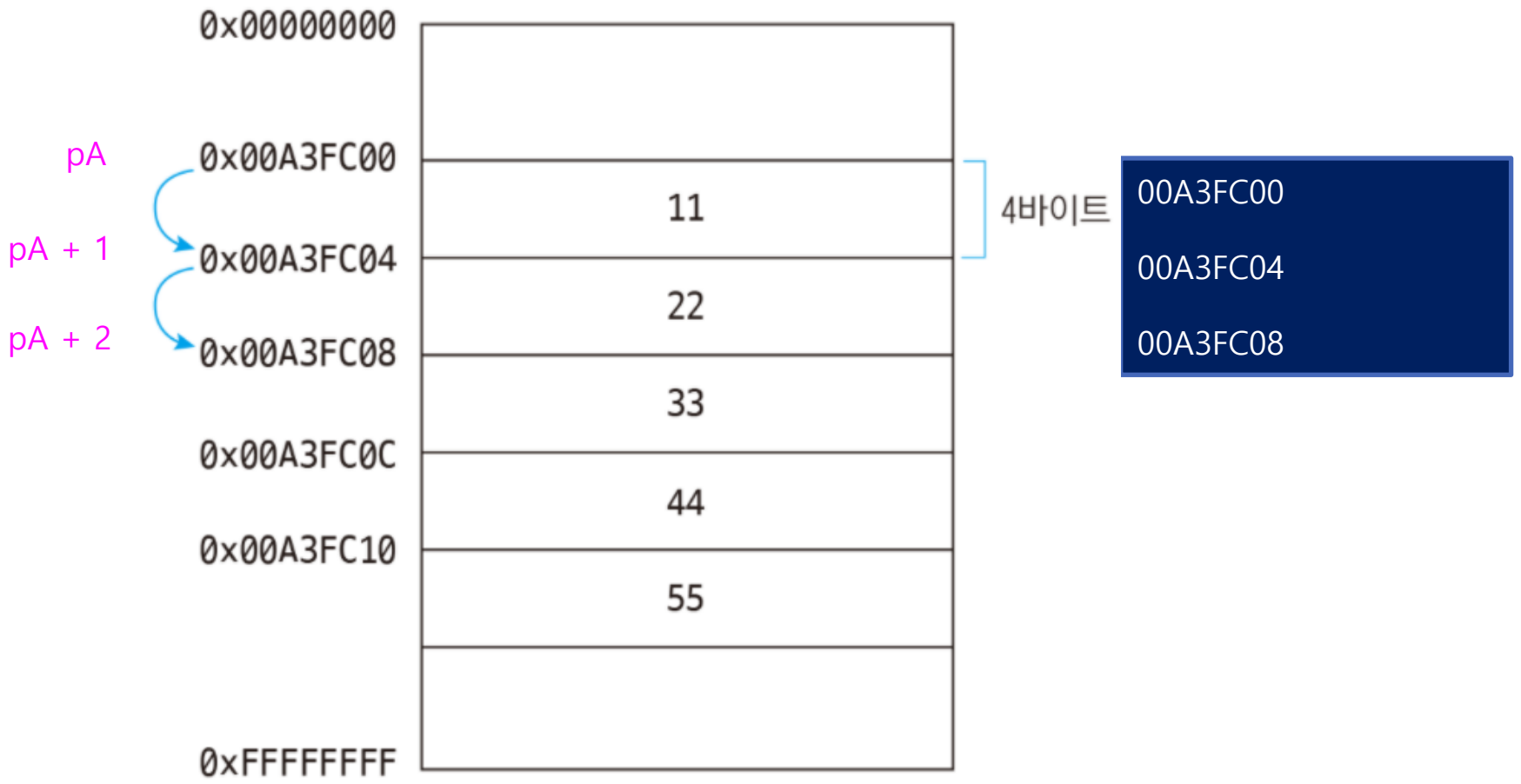

변수값
변수이름
주소
3.141592
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
pd-1
pd
pd+1
pd는
double형 포인터

간접 참조 연산자와 증감 연산자

```
#include <stdio.h>

int main()
{
	int arr[5] = { 11, 22, 33, 44, 55 };
	int *pA;
	int *pB;
	int *pC;

	pA = &arr[0];	// 배열 첫 번째 요소의 주소를 포인터에 저장

	pB = pA + 1;
	pC = pA + 2;


	printf("%p₩n", pA);	// 00A3FC00: 메모리 주소. 컴퓨터마다, 실행할 때마다 달라짐
	printf("%p₩n", pB);	// 00A3FC04: sizeof(int) * 1이므로 pA에서 4가 증가함
	printf("%p₩n", pC);	// 00A3FC08: sizeof(int) * 2이므로 pA에서 8이 증가함

	return 0;
}
}
```

```
00A3FC00
00A3FC04
00A3FC08
```

Lab : 간접 참조 연산자와 증감 연산자



[문제]

정수형 변수 3을 기억 한 후, 아래 조건을 수행하시오.

① 위 주소 값을 포인터 변수에도 대입하세요.
② 정수형 변수 a에 기억된 값, 포인터가 가리키는 값을 각각 출력하세요.
③ 정수형 변수 주소 값, 포인터 주소 값을 16진법으로 출력하세요
④ 기억된 포인터에 100이라는 값을 새롭게 저장하세요.
⑤ 정수형 변수 a에 400이라는 값을 더합니다.
⑤ 연산 된 후, 정수형 변수 a와 포인터가 가리키는 값을 각각 출력하세요.
⑥ 정수형 변수 주소를 2 byte 더한 후, 주소 값 출력 출력하세요.

연산 전 값:3, 포인터 값:3
정수형 변수 주소 : **7339228**, 포인터 변수 주소 : 006FFCDC
연산 후 정수 값 : 500, 포인터 값 출력 : 500
정수형 변수 주소를 2byte 더한 후 주소 값 출력 : **7339236**

[문제]

- 배열에 int arr[5] = { 5,3,8,9,1 }을 기억시킨 후, 기억된 배열 원소를 포인터의 주소 값과 배열 요소를 포인터 변수를 이용하여 [출력 형태]와 같이 출력하도록 프로그램을 작성하시오.

[실행 결과]

포인터 주소 값 00CFFA84 5
포인터 주소 값 00CFFA88 3
포인터 주소 값 00CFFA8C 8
포인터 주소 값 00CFFA90 9
포인터 주소 값 00CFFA94 1

[문제]

- 배열에 int arr[5] = { 5,3,8,9,1 }을 기억시킨 후, [실행 결과]와 같이 배열과 포인터의 관계를 이해하면서 동일한 주소와 데이터 값이 출력되는지 프로그램을 통해 확인해 보자.

[실행 결과]

배열의 주소 값 00B3FE20 5
배열의 주소 값 00B3FE24 3
배열의 주소 값 00B3FE28 8
배열의 주소 값 00B3FE2C 9
배열의 주소 값 00B3FE30 1

포인터의 주소 값 00B3FE20 5
포인터의 주소 값 00B3FE24 3
포인터의 주소 값 00B3FE28 8
포인터의 주소 값 00B3FE2C 9
포인터의 주소 값 00B3FE30 1

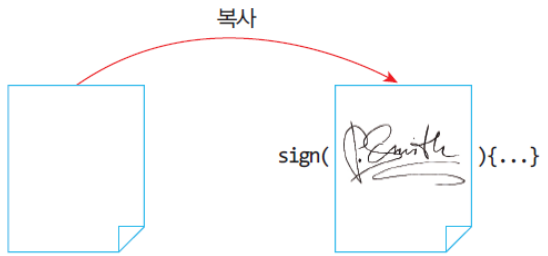
2

함수와 포인터

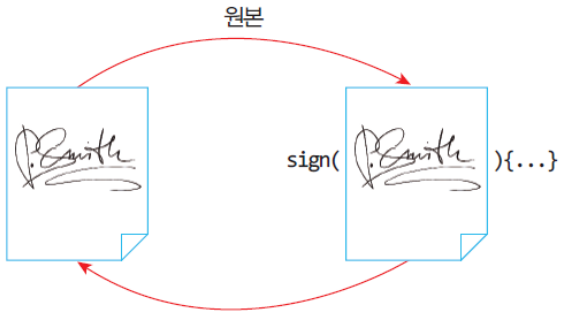
- 다른 사람에게 넘겨주어야 하는 정보가 상당히 방대하다고 하자. 이런 경우에는 전체를 복사해서 주는 것보다는 페이지 수만 알려주는 편이 간결할 수 있다.

함수 호출시 인수 전달 방법

- **값에 의한 호출(call-by-value)**
  - C의 기본적인 방법
  - 인수의 값이 매개 변수로 복사된다.



- **참조에 의한 호출(call-by-reference)**
  - C에서는 포인터를 이용하여 흉내낼 수 있다.
  - 인수의 주소가 매개 변수로 복사된다.



값에 의한 호출

```
#include <stdio.h>

void modify(int value)
{
	value = 99;
}

int main(void)
{
	int number = 1;

	modify(number);
	printf("number = %d₩n", number);

	return 0;
}
```

참조에 의한 호출

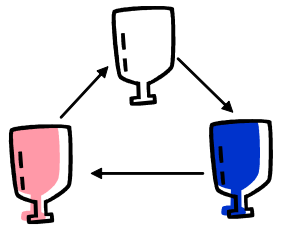
```
#include <stdio.h>

void modify(int *ptr)
{
        *ptr = 99;                      // 매개 변수를 통하여 원본을 변경한다.
}

int main(void)
{
        int number = 1;

        modify(&number);          // 주소를 계산해서 보낸다.
        printf("number = %d₩n", number);


        return 0;
}
```

swap() 함수 #1

- 변수 2개의 값을 바꾸는 작업을 함수로 작성

```
int main(void)
{
    int a = 10, b = 20;
       printf("swap() 호출 전 a=%d, b=%d₩n", a,
    b);

    swap(a,      b);

    printf("swap() 호출후 a=%d b=%d₩n",a, b);

    return 0;
}
```

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

    tmp = x;
    x = y;
    y = tmp;
}
```



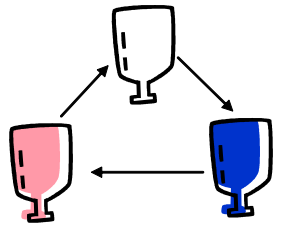
```
C:₩Windows₩system32₩cmd.exe
swap() 호출 전 a=10, b=20
swap() 호출 후 a=10, b=20
```

swap() 함수 #2

- **포인터를 이용**

```
int main(void)
{
    int a = 100, b = 200;
    printf("swap() 호출 전 a=%d b=%d₩n",a, b);

    swap(&a, &b);

    printf("swap() 호출 후 a=%d b=%d₩n",a, b);

    return 0;
}
```

```
void swap(int *px, int *py)
{
    int   tmp;

    tmp = *px;
    *px = *py;
    *py = tmp;

}
```

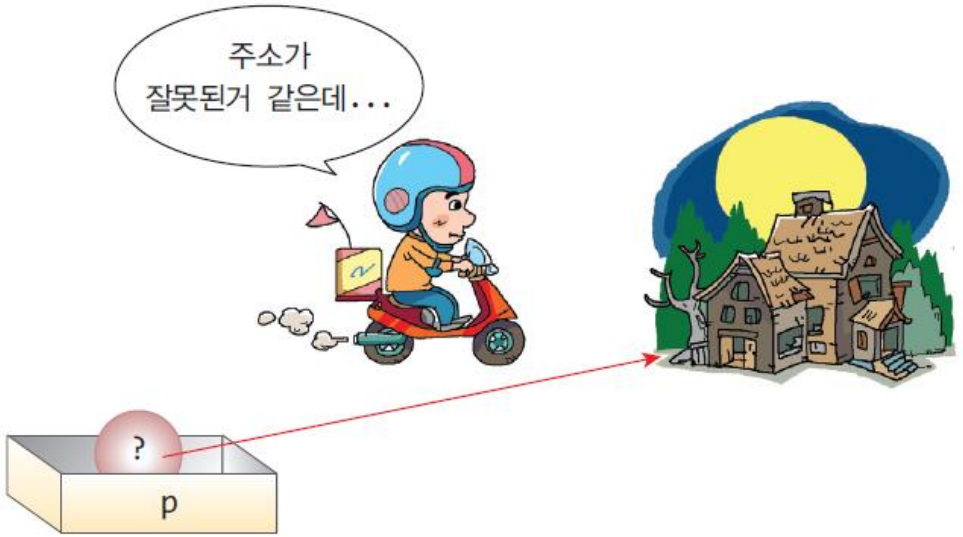


```
C:₩Windows₩system32₩cmd.exe
swap() 호출전 a=100 b=200
swap() 호출후 a=200 b=100
계속하려면 아무 키나 누르십시오 . . .
```

포인터 사용시 주의점

- **초기화가 안된 포인터를 사용하면 안된다.**

```
int main(void)
{
    int *p;             // 포인터 p는 초기화가 안되어 있음
    *p = 100;           // 위험한 코드
    return 0;
}
```



포인터 사용시 주의점

- **포인터의 타입과 변수의 타입은 일치하여야 한다.**

```
#include <stdio.h>

int main(void)
{
    int i;
    double *pd;

    pd = &i;          // 오류! double형 포인터에 int형 변수의 주소를 대입
    *pd = 36.5;

    return 0;
}
```

[문제]

사용자로 부터 어떤 수 A, 어떤 수B를 각각 입력 받아 함수를 호출하여 A와 B에 입력된 두 수를 참조에 의한 호출로 서로 바꾸어 출력하도록 프로그램을 작성하시오.

[실행 결과]

어떤 수A:35

어떤 수B:100

호출 전 : 어떤 수 A=35, 어떤 수B=100

호출 후 : 어떤 수 A=100, 어떤 수B=35

[문제]

다음과 같이 배열에 숫자 정수 원소가 기억되어 있다.

int arr[5] = {25,15,23,2,144 };

함수를 호출하여 함수 프로그램에서 기억된 배열 포인터 명령을 이용하여 [실행 결과]와 같이 출력하시오.

[실행 결과]

25
15
23
2
144

[문제]

다음과 같이 배열에 숫자 정수 원소가 기억되어 있다.
int arr[5] = {25,15,23,2,144 };
함수를 호출하여 함수 프로그램에서 기억된 배열 원소 및 주소 값이 배열과 포인터를 활용하여 동일하게 출력되도록 [실행 결과]와 같이 출력하시오.

[실행 결과]

```
배열 값  25   주소 값 5240968
배열 값  15   주소 값 5240972
배열 값  23   주소 값 5240976
배열 값   2   주소 값 5240980
배열 값 144   주소 값 5240984
포인터 값  25   주소 값 5240968
포인터 값  15   주소 값 5240972
포인터 값  23   주소 값 5240976
포인터 값   2   주소 값 5240980
포인터 값 144   주소 값 5240984
```

감사합니다.