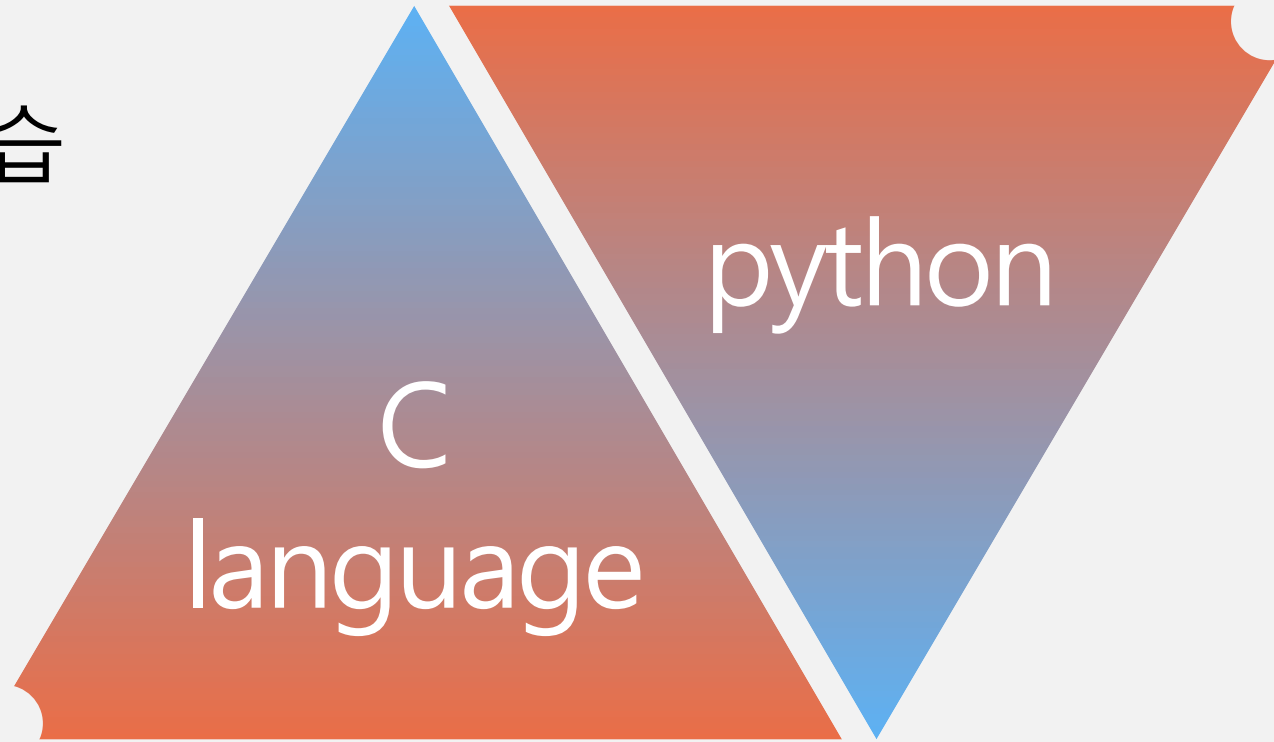




소입설
멘토링 수업 ●
(13주차)

- 1 지난 주 수업 복습
- 2 Pointer and Array
- 3 기말고사 대비

1. 지난 주 수업 복습



C08. Array.pdf

```
int arr[6]={0,0,0,0,212,0};  
int arr[6]={ [4]=212};    // after C99  
int arr[6]={0,1,[5]=10,7,[1]=9,8};
```

C08. Array.pdf

```
int oxen[6]={0,0,0,0,212,0};  
int yaks[6]={0};  
oxen=yaks;                // not allowed  
yaks[6]=oxen[6];          // invalid  
yaks[6]={1,2,3,4,5,6};    // doesn't work
```

C08. Array.pdf

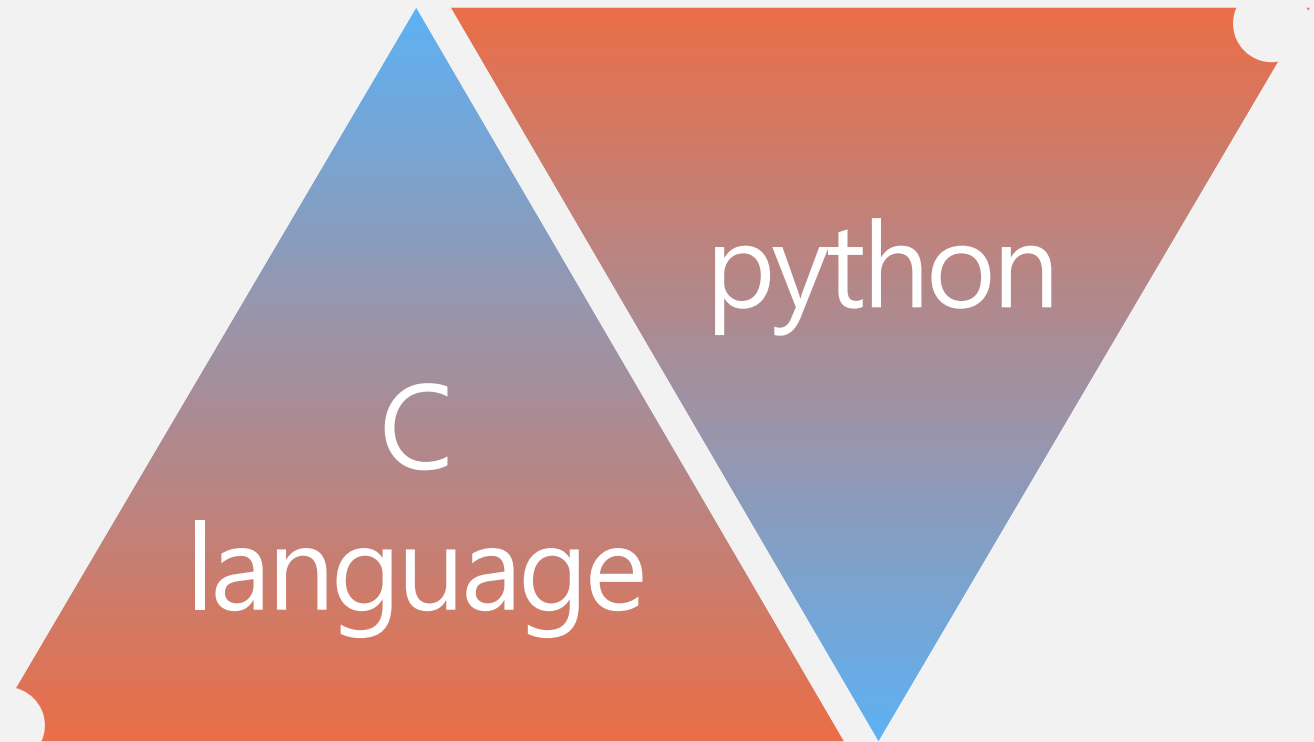
```
int n=5;
int m=8;
float a1[5]; //yes
float a2[5*2+1]; //yes
float a3[sizeof(int) +1]; //yes
float a4[-4]; //no
float a5[0]; //no
float a6[2.5]; //no
float a7[(int)2.5]; //yes
float a8[n]; //yes since C99
float a9[m]; //yes since C99
```

C08. Array.pdf

[typedef]

1. typedef unsigned int UINT32; <기본>
 UINT32 sysInt; // unsigned int sysInt;
 typedef char *STRING;
 STRING str1, str2;
 // #define STRING char* 와 비교
2. typedef int(*PF)(int,int);
 // 알려줌
3. typedef int ARY[2];
 ARY a[3]; // int a[3][2];
 ARY arr; // int arr[2];

2. Pointer and Array



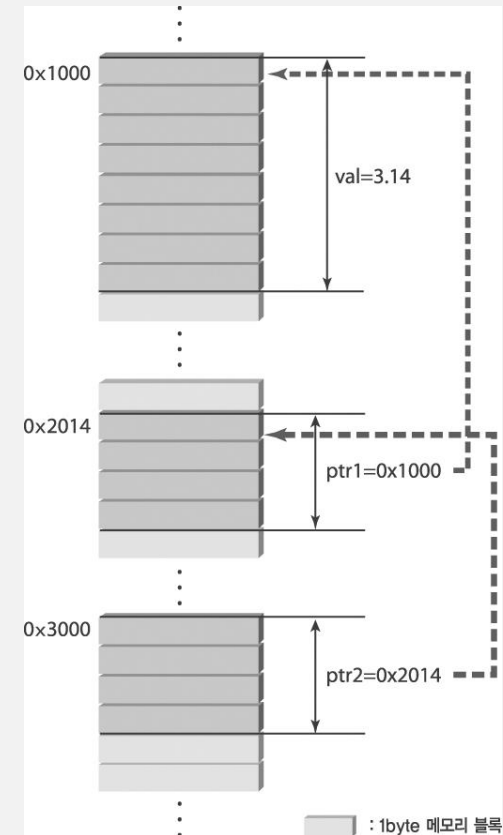
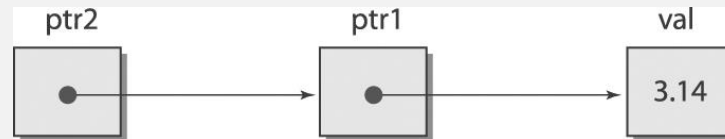
1) Pointer

포인터의 포인터

더블 포인터라고 불린다.

싱글 포인터의 주소 값을 저장하는 용도의 포인터

```
int main(void)
{
    double val=3.14;
    double *ptr1 = &val; // 싱글 포인터
    double **ptr2 = &ptr1; // 더블 포인터
    ...
}
```



구현 사례 1 : 효과 없는 swap 함수의 호출

```
#include <stdio.h>

void pswap(int *p1, int *p2);

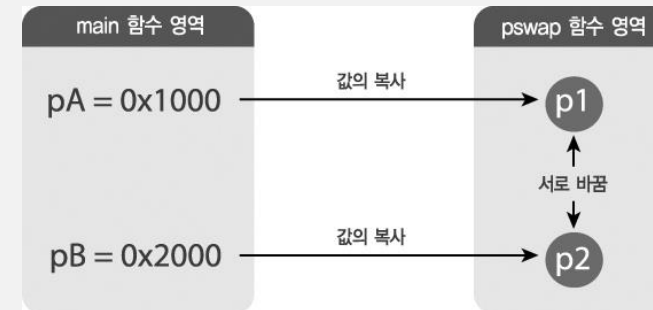
int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}
```

```
void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```



2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	CC	0xF029	CC	0xF038	CC	0xF057	CC
0xF01B	CC	0xF02A	CC	0xF039	CC	0xF058	CC
0xF01C	CC	0xF02B	CC	0xF04A	CC	0xF059	CC
0xF01D	CC	0xF02C	CC	0xF04B	CC	0xF06A	CC
0xF01E	CC	0xF02D	CC	0xF04C	CC	0xF06B	CC
0xF01F	CC	0xF02E	CC	0xF04D	CC	0xF06C	CC
0xF020	CC	0xF02F	CC	0xF04E	CC	0xF06D	CC
0xF021	CC	0xF030	CC	0xF04F	CC	0xF06E	CC
0xF022	CC	0xF031	CC	0xF050	CC	0xF06F	CC
0xF023	CC	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	CC	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	CC	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	CC	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	CC	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	CC	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	CC	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	CC	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	CC	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	CC	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	CC	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	CC	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	CC	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	CC	0xF04F	CC	0xF06E	CC
0xF022	CC	0xF031	CC	0xF050	CC	0xF06F	CC
0xF023	CC	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	CC	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	CC	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	CC	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	CC	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	CC	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	CC	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	CC	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	CC	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	CC	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	CC	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	CC	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	CC	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	CC	0xF04F	CC	0xF06E	CC
0xF022	CC	0xF031	CC	0xF050	CC	0xF06F	CC
0xF023	CC	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	CC	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	CC	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	CC	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	CC	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	CC	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	00	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	1E	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	1A	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	CC	0xF071	CC
0xF025	00	0xF034	00	0xF053	CC	0xF072	CC
0xF026	00	0xF035	00	0xF054	CC	0xF073	CC
0xF027	00	0xF036	00	0xF055	CC	0xF074	CC
0xF028	00	0xF037	00	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	00	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	1E	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	1A	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	CC	0xF071	CC
0xF025	00	0xF034	00	0xF053	CC	0xF072	CC
0xF026	00	0xF035	00	0xF054	CC	0xF073	CC
0xF027	00	0xF036	00	0xF055	CC	0xF074	CC
0xF028	00	0xF037	00	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	1E	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	1A	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	1E	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	1E	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	1A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	1E	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int *p1, int *p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(pA, pB);

    // 함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int *p1, int *p2)
{
    int *temp;
    temp=p1;
    p1=p2;
    p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int *p1, int *p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(pA, pB);
```

```
    // 함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int *p1, int *p2)
```

```
{
```

```
    int *temp;
```

```
    temp=p1;
```

```
    p1=p2;
```

```
    p2=temp;
```

```
}
```

0xF01A

0A

0xF029

00

0xF038

CC

0xF057

CC

0xF01B

00

0xF02A

1E

0xF039

CC

0xF058

CC

0xF01C

00

0xF02B

F0

0xF04A

CC

0xF059

CC

0xF01D

00

0xF02C

00

0xF04B

CC

0xF06A

CC

0xF01E

14

0xF02D

00

0xF04C

CC

0xF06B

CC

*(pA)

0xF02E

00

0xF04D

CC

0xF06C

CC

0xF02F

00

0xF04E

CC

0xF06D

CC

0xF030

00

0xF04F

CC

0xF06E

CC

0xF022

1A

0xF031

00

0xF050

CC

0xF06F

CC

0xF023

F0

0xF032

CC

0xF051

CC

0xF070

CC

0xF024

00

0xF033

CC

0xF052

CC

0xF071

CC

0xF025

00

0xF034

CC

0xF053

CC

0xF072

CC

0xF026

00

0xF035

CC

0xF054

CC

0xF073

CC

0xF027

00

0xF036

CC

0xF055

CC

0xF074

CC

0xF028

00

0xF037

CC

0xF056

CC

0xF028

CC

2 Pointer and Array

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

```
void pswap(int *p1, int *p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(pA, pB);
```

```
    // 함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int *p1, int *p2)
```

```
{
```

```
    int *temp;
```

```
    temp=p1;
```

```
    p1=p2;
```

```
    p2=temp;
```

```
}
```

0xF01A

0A

0xF029

00

0xF038

CC

0xF057

CC

0xF01B

00

0xF02A

1E

0xF039

CC

0xF058

CC

0xF01C

00

0xF02B

F0

0xF04A

CC

0xF059

CC

0xF01D

00

0xF02C

00

0xF04B

CC

0xF06A

CC

0xF01E

1A

0xF02D

00

0xF04C

CC

0xF06B

CC

*(0xF01A)

0xF02E

00

0xF04D

CC

0xF06C

CC

0xF02F

00

0xF04E

CC

0xF06D

CC

0xF030

00

0xF04F

CC

0xF06E

CC

0xF022

1A

0xF031

00

0xF050

CC

0xF06F

CC

0xF023

F0

0xF032

CC

0xF051

CC

0xF070

CC

0xF024

00

0xF033

CC

0xF052

CC

0xF071

CC

0xF025

00

0xF034

CC

0xF053

CC

0xF072

CC

0xF026

00

0xF035

CC

0xF054

CC

0xF073

CC

0xF027

00

0xF036

CC

0xF055

CC

0xF074

CC

0xF028

00

0xF037

CC

0xF056

CC

0xF028

CC

2 Pointer and Array

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

```
void pswap(int *p1, int *p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(pA, pB);
```

```
    // 함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int *p1, int *p2)
```

```
{
```

```
    int *temp;
```

```
    temp=p1;
```

```
    p1=p2;
```

```
    p2=temp;
```

```
}
```

0xF01A

0A

0xF029

00

0xF038

CC

0xF057

CC

0xF01B

00

0xF02A

1E

0xF039

CC

0xF058

CC

0xF01C

00

0xF02B

F0

0xF04A

CC

0xF059

CC

0xF01D

00

0xF02C

00

0xF04B

CC

0xF06A

CC

0xF01E

1A

0xF02D

00

0xF04C

CC

0xF06B

CC

*(0xF01A)

0xF02E

00

0xF04D

CC

0xF06C

CC

0xF02F

00

0xF04E

CC

0xF06D

CC

0xF030

00

0xF04F

CC

0xF06E

CC

0xF022

1A

0xF031

00

0xF050

CC

0xF06F

CC

0xF023

F0

0xF032

CC

0xF051

CC

0xF070

CC

0xF024

00

0xF033

CC

0xF052

CC

0xF071

CC

0xF025

00

0xF034

CC

0xF053

CC

0xF072

CC

0xF026

00

0xF035

CC

0xF054

CC

0xF073

CC

0xF027

00

0xF036

CC

0xF055

CC

0xF074

CC

0xF028

00

0xF037

CC

0xF056

CC

0xF028

CC

2 Pointer and Array

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

```
void pswap(int *p1, int *p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(pA, pB);
```

```
    // 함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int *p1, int *p2)
```

```
{
```

```
    int *temp;
```

```
    temp=p1;
```

```
    p1=p2;
```

```
    p2=temp;
```

```
}
```

0xF01A

0A

0xF029

00

0xF038

CC

0xF057

CC

0xF01B

00

0xF02A

1E

0xF039

CC

0xF058

CC

0xF01C

00

0xF02B

F0

0xF04A

CC

0xF059

CC

0xF01D

00

0xF02C

00

0xF04B

CC

0xF06A

CC

0xF01E

1A

0xF02D

00

0xF04C

CC

0xF06B

CC

0xA

0xF02E

00

0xF04D

CC

0xF06C

CC

0xF02F

00

0xF04E

CC

0xF06D

CC

0xF030

00

0xF04F

CC

0xF06E

CC

0xF022

1A

0xF031

00

0xF050

CC

0xF06F

CC

0xF023

F0

0xF032

CC

0xF051

CC

0xF070

CC

0xF024

00

0xF033

CC

0xF052

CC

0xF071

CC

0xF025

00

0xF034

CC

0xF053

CC

0xF072

CC

0xF026

00

0xF035

CC

0xF054

CC

0xF073

CC

0xF027

00

0xF036

CC

0xF055

CC

0xF074

CC

0xF028

00

0xF037

CC

0xF056

CC

0xF028

CC

2 Pointer and Array

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

```
void pswap(int *p1, int *p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(pA, pB);
```

```
    // 함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int *p1, int *p2)
```

```
{
```

```
    int *temp;
```

```
    temp=p1;
```

```
    p1=p2;
```

```
    p2=temp;
```

```
}
```

0xF01A

0A

0xF029

00

0xF038

CC

0xF057

CC

0xF01B

00

0xF02A

1E

0xF039

CC

0xF058

CC

0xF01C

00

0xF02B

F0

0xF04A

CC

0xF059

CC

0xF01D

00

0xF02C

00

0xF04B

CC

0xF06A

CC

0xF01E

1A

0xF02D

00

0xF04C

CC

0xF06B

CC

10

0xF02E

00

0xF04D

CC

0xF06C

CC

0xF02F

00

0xF04E

CC

0xF06D

CC

0xF030

00

0xF04F

CC

0xF06E

CC

0xF022

1A

0xF031

00

0xF050

CC

0xF06F

CC

0xF023

F0

0xF032

CC

0xF051

CC

0xF070

CC

0xF024

00

0xF033

CC

0xF052

CC

0xF071

CC

0xF025

00

0xF034

CC

0xF053

CC

0xF072

CC

0xF026

00

0xF035

CC

0xF054

CC

0xF073

CC

0xF027

00

0xF036

CC

0xF055

CC

0xF074

CC

0xF028

00

0xF037

CC

0xF056

CC

0xF028

CC

2 Pointer and Array

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

```
void pswap(int *p1, int *p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(pA, pB);
```

```
    // 함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int *p1, int *p2)
```

```
{
```

```
    int *temp;
```

```
    temp=p1;
```

```
    p1=p2;
```

```
    p2=temp;
```

```
}
```

0xF01A

0A

0xF029

00

0xF038

CC

0xF057

CC

0xF01B

00

0xF02A

1E

0xF039

CC

0xF058

CC

0xF01C

00

0xF02B

F0

0xF04A

CC

0xF059

CC

0xF01D

00

0xF02C

00

0xF04B

CC

0xF06A

CC

0xF01E

14

0xF02D

00

0xF04C

CC

0xF06B

CC

0xF02E

00

0xF04D

CC

0xF06C

CC

0xF02F

00

0xF04E

CC

0xF06D

CC

0xF030

00

0xF04F

CC

0xF06E

CC

0xF022

1A

0xF031

00

0xF050

CC

0xF06F

CC

0xF023

F0

0xF032

CC

0xF051

CC

0xF070

CC

0xF024

00

0xF033

CC

0xF052

CC

0xF071

CC

0xF025

00

0xF034

CC

0xF053

CC

0xF072

CC

0xF026

00

0xF035

CC

0xF054

CC

0xF073

CC

0xF027

00

0xF036

CC

0xF055

CC

0xF074

CC

0xF028

00

0xF037

CC

0xF056

CC

0xF028

CC

20

구현 사례 2 : 더블 포인터 입장에서의 swap

```
#include <stdio.h>

void pswap(int **p1, int **p2);

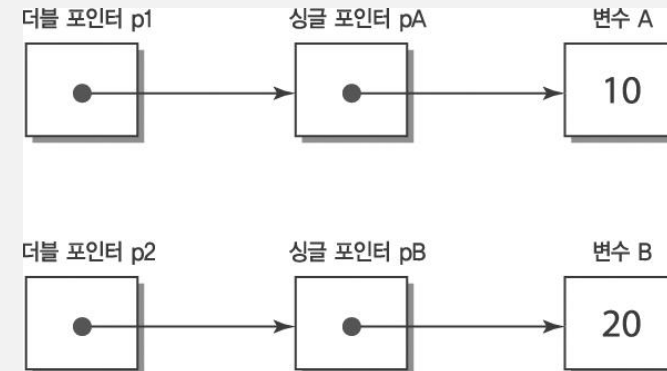
int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}
```

```
void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```



2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

0xF01A	CC	0xF029	CC	0xF038	CC	0xF057	CC
0xF01B	CC	0xF02A	CC	0xF039	CC	0xF058	CC
0xF01C	CC	0xF02B	CC	0xF04A	CC	0xF059	CC
0xF01D	CC	0xF02C	CC	0xF04B	CC	0xF06A	CC
0xF01E	CC	0xF02D	CC	0xF04C	CC	0xF06B	CC
0xF01F	CC	0xF02E	CC	0xF04D	CC	0xF06C	CC
0xF020	CC	0xF02F	CC	0xF04E	CC	0xF06D	CC
0xF021	CC	0xF030	CC	0xF04F	CC	0xF06E	CC
0xF022	CC	0xF031	CC	0xF050	CC	0xF06F	CC
0xF023	CC	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	CC	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	CC	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	CC	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	CC	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	CC	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

0xF01A	0A	0xF029	CC	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	CC	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	CC	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	CC	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	CC	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	CC	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	CC	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	CC	0xF04F	CC	0xF06E	CC
0xF022	CC	0xF031	CC	0xF050	CC	0xF06F	CC
0xF023	CC	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	CC	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	CC	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	CC	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	CC	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	CC	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

0xF01A	0A	0xF029	CC	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	CC	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	CC	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	CC	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	CC	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	CC	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	CC	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	CC	0xF04F	CC	0xF06E	CC
0xF022	CC	0xF031	CC	0xF050	CC	0xF06F	CC
0xF023	CC	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	CC	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	CC	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	CC	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	CC	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	CC	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	00	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	CC	0xF071	CC
0xF025	00	0xF034	00	0xF053	CC	0xF072	CC
0xF026	00	0xF035	00	0xF054	CC	0xF073	CC
0xF027	00	0xF036	00	0xF055	CC	0xF074	CC
0xF028	00	0xF037	00	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

0xF01A	0A	0xF029	00	0xF038	00	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	CC	0xF071	CC
0xF025	00	0xF034	00	0xF053	CC	0xF072	CC
0xF026	00	0xF035	00	0xF054	CC	0xF073	CC
0xF027	00	0xF036	00	0xF055	CC	0xF074	CC
0xF028	00	0xF037	00	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

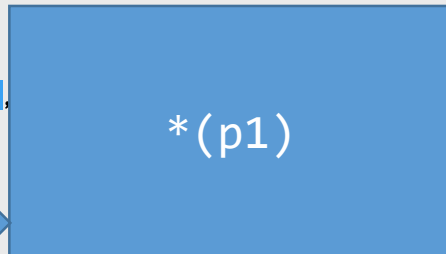
```
    int *temp;
```

```
    temp=*p1
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```



0xF01A	0A	0xF029	00	0xF038	00	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	CC	0xF071	CC
0xF025	00	0xF034	00	0xF053	CC	0xF072	CC
0xF026	00	0xF035	00	0xF054	CC	0xF073	CC
0xF027	00	0xF036	00	0xF055	CC	0xF074	CC
0xF028	00	0xF037	00	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

```
    int *temp;
```

```
    temp=*p1
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

***(0xF022)**

0xF01A	0A	0xF029	00	0xF038	00	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	CC	0xF071	CC
0xF025	00	0xF034	00	0xF053	CC	0xF072	CC
0xF026	00	0xF035	00	0xF054	CC	0xF073	CC
0xF027	00	0xF036	00	0xF055	CC	0xF074	CC
0xF028	00	0xF037	00	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

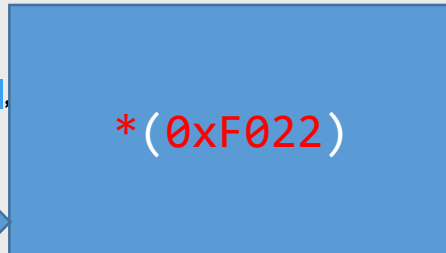
```
    int *temp;
```

```
    temp=*p1
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```



0xF01A	0A	0xF029	00	0xF038	00	0xF057	CC
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	CC	0xF071	CC
0xF025	00	0xF034	00	0xF053	CC	0xF072	CC
0xF026	00	0xF035	00	0xF054	CC	0xF073	CC
0xF027	00	0xF036	00	0xF055	CC	0xF074	CC
0xF028	00	0xF037	00	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

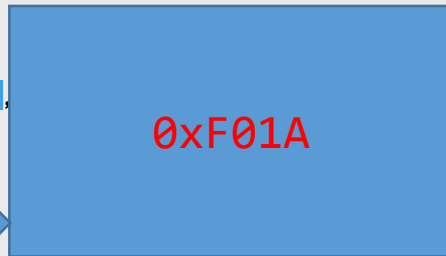
```
    int *temp;
```

```
    temp=*p1
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```



0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

*p1=*p2

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
int A=10, B=20;
```

```
int *pA, *pB;
```

```
pA=&A, pB=&B;
```

```
pswap(&pA, &pB);
```

```
//함수 호출 후
```

```
printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

```
int *temp;
```

```
temp=*p1;
```

```
*p1=*p2;
```

```
*p2=temp;
```

```
}
```

***p1=*(0xF02A)**

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

p1=(0xF02A)

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

*p1=0xF01E

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1,
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

$*(0xF022)$
=0xF01E

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1,
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```

***(0xF022)**
=0xF01E

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1A	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1,
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

$*(0xF01A)$
 $= 0xF01E$

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```



*p2=temp

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

void pswap(int **p1, int **p2);

int main(void)
{
    int A=10, B=20;
    int *pA, *pB;
    pA=&A, pB=&B;

    pswap(&pA, &pB);

    //함수 호출 후
    printf("pA가 가리키는 변수 : %d \n", *pA);
    printf("pB가 가리키는 변수 : %d \n", *pB);

    return 0;
}

void pswap(int **p1, int **p2)
{
    int *temp;
    temp=*p1;
    *p1=*p2;
    *p2=temp;
}
```



***p2=0xF01A**

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

$*(0xF02A)$
=0xF01A

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

***(0xF02A)**
=0xF01A

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1E	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

$*(0xF02A)$
 $= 0xF01A$

0xF01A	0A	0xF029	00	0xF038	00	0xF057	00
0xF01B	00	0xF02A	1A	0xF039	00	0xF058	00
0xF01C	00	0xF02B	F0	0xF04A	2A	0xF059	00
0xF01D	00	0xF02C	00	0xF04B	F0	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	00	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	00	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	00	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	00	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	00	0xF06F	CC
0xF023	F0	0xF032	22	0xF051	00	0xF070	CC
0xF024	00	0xF033	F0	0xF052	1A	0xF071	CC
0xF025	00	0xF034	00	0xF053	F0	0xF072	CC
0xF026	00	0xF035	00	0xF054	00	0xF073	CC
0xF027	00	0xF036	00	0xF055	00	0xF074	CC
0xF028	00	0xF037	00	0xF056	00	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

*pA

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1A	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

*(0xF01E)

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1A	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```


```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

***(0xF01E)**

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1A	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

0x14

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1A	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

20

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1A	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```



0xA

0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1A	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

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

```
void pswap(int **p1, int **p2);
```

```
int main(void)
```

```
{
```

```
    int A=10, B=20;
```

```
    int *pA, *pB;
```

```
    pA=&A, pB=&B;
```

```
    pswap(&pA, &pB);
```

```
    //함수 호출 후
```

```
    printf("pA가 가리키는 변수 : %d \n", *pA);
```

```
    printf("pB가 가리키는 변수 : %d \n", *pB);
```

```
    return 0;
```

```
}
```

```
void pswap(int **p1, int **p2)
```

```
{
```

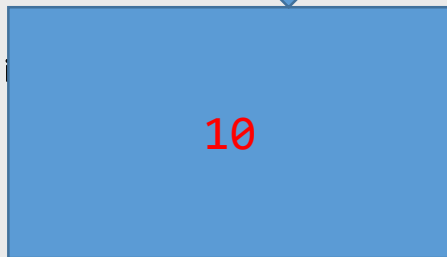
```
    int *temp;
```

```
    temp=*p1;
```

```
    *p1=*p2;
```

```
    *p2=temp;
```

```
}
```

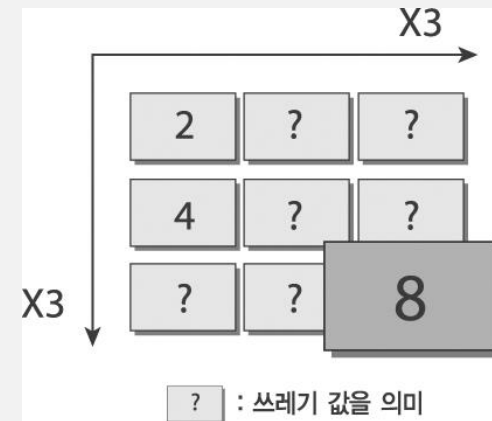
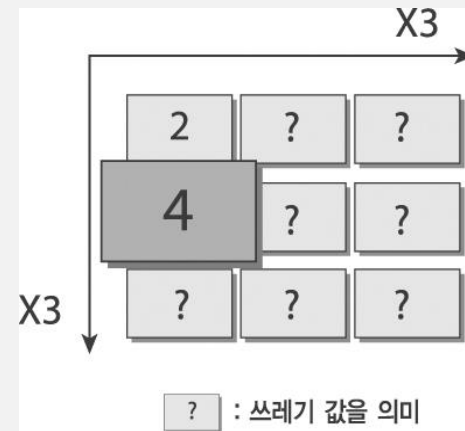
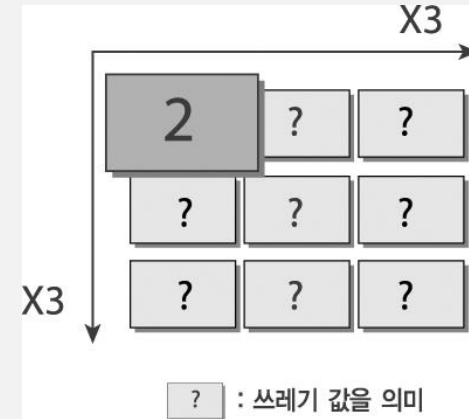


0xF01A	0A	0xF029	00	0xF038	CC	0xF057	CC
0xF01B	00	0xF02A	1A	0xF039	CC	0xF058	CC
0xF01C	00	0xF02B	F0	0xF04A	CC	0xF059	CC
0xF01D	00	0xF02C	00	0xF04B	CC	0xF06A	CC
0xF01E	14	0xF02D	00	0xF04C	CC	0xF06B	CC
0xF01F	00	0xF02E	00	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	CC
0xF021	00	0xF030	00	0xF04F	CC	0xF06E	CC
0xF022	1E	0xF031	00	0xF050	CC	0xF06F	CC
0xF023	F0	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	00	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	00	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	00	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	00	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	00	0xF037	CC	0xF056	CC	0xF028	CC

2) Array

2차원 배열 요소의 접근 방법

```
int main(void)
{
    int arr[3][3];
    arr[0][0]=2;
    arr[1][0]=4;
    arr[2][2]=8;
    . . . . .
}
```



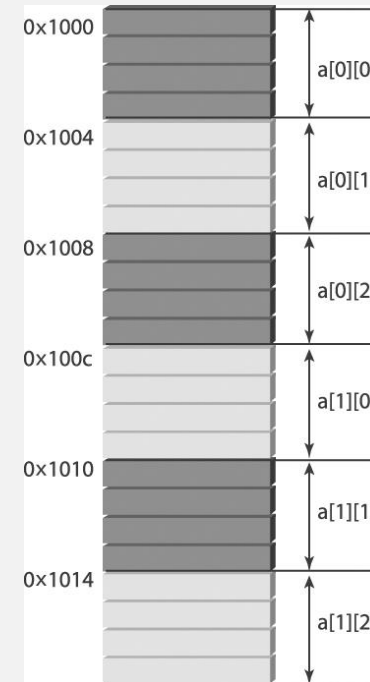
다차원 배열의 실제 메모리 구성

1차원 배열과 동일하다.

다만 접근 방법을 2차원적으로 해석할 뿐이다.

2차원적으로 이해하는 것이 좋은 습관!

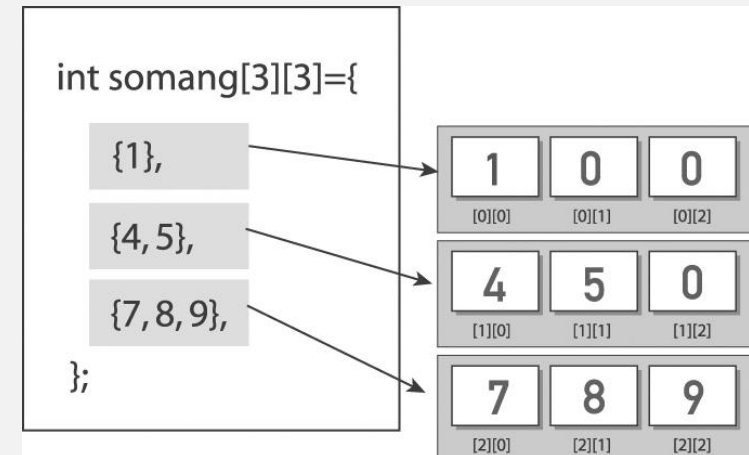
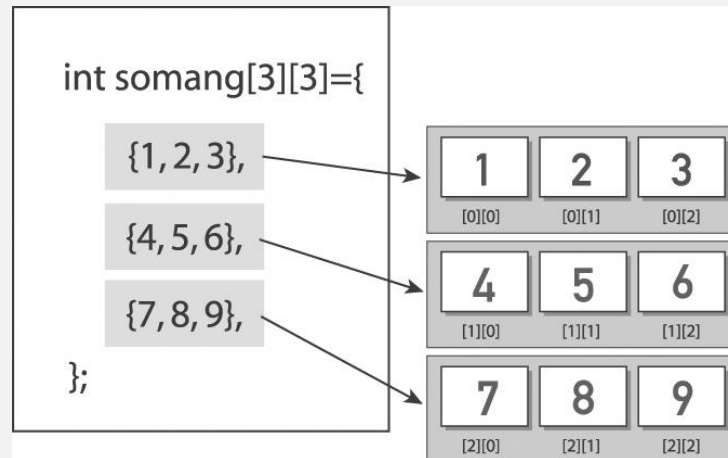
`int a[2][3]`



2차원 배열! 선언과 동시에 초기화

case 1 : 행 단위로 모든 요소들을 초기화

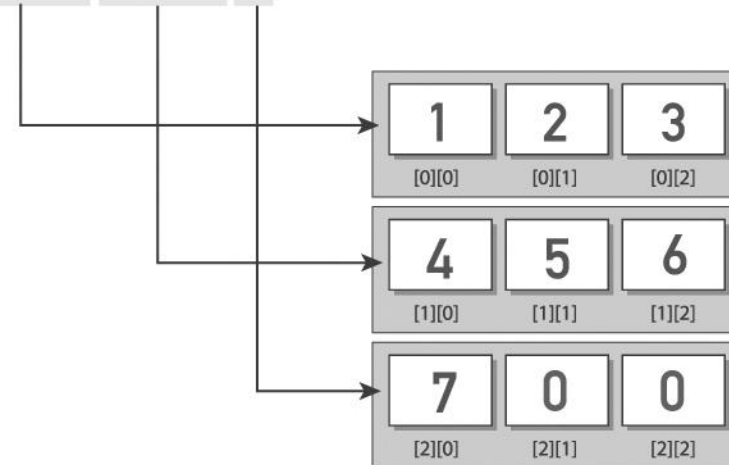
case 2 : 행 단위로 일부 요소들만 초기화



2차원 배열! 선언과 동시에 초기화

case 3 : 1차원 배열 형태의 초기화

```
int somang[3][3]={ 1,2,3, 4,5,6, 7};
```



초기화 리스트에 의한 배열 크기의 결정

1차원 배열의 예

–int arr[]={1, 2, 3, 4, 5};

–sizeof(arr): ??

2차원 배열의 예

–int arr1[4][]={1, 2, 3, 4, 5, 6, 7, 8}; //Error! 왜?

–int arr2[][4]={1, 2, 3, 4, 5, 6, 7, 8}; //OK!

–int arr3[][2]={1, 2, 3, 4, 5, 6, 7, 8}; //OK!

–sizeof(arr3): ??

3차원 배열의 선언과 의미

3차원적 메모리 구조를 의미함

개념만 이해하면 충분, 일반적으로 필요 없다.

4차원 이상의 배열은 4차원의 형태가 되므로 구조적인
이해 불가!!

```
int a[3][3][3]
```

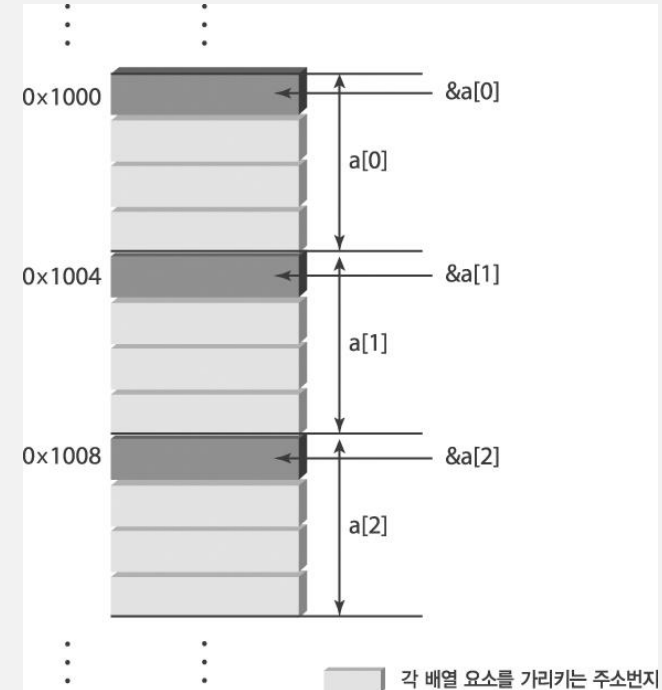


3) Pointer and Array

배열의 이름의 정체

배열 이름은 첫 번째 요소의 주소 값을 나타낸다.

```
int a[5]={0, 1, 2, 3, 4}
```



배열 이름의 활용

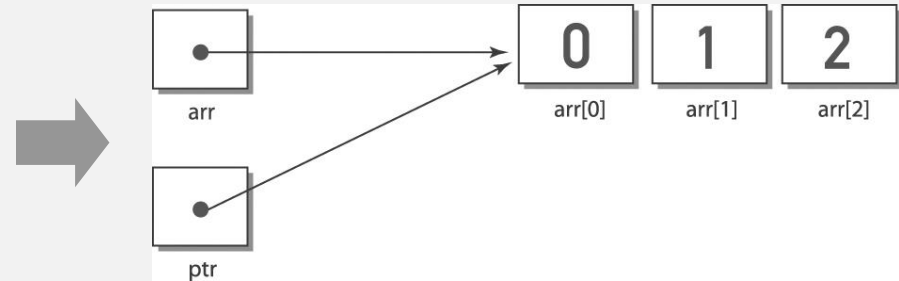
배열 이름을 포인터처럼, 포인터를 배열 이름처럼 활용하는 것이 가능!

```
#include <stdio.h>

int main(void)
{
    int arr[3]={0, 1, 2};
    int *ptr;

    ptr=arr;

    printf("%d, %d, %d\n", ptr[0], ptr[1], ptr[2]);
    return 0;
}
```



포인터 연산이란?

포인터가 지니는 값(주소)을 증가 혹은
감소시키는 연산을 의미

```
ptr1++;  
ptr1 += 3;  
--ptr1;  
ptr2=ptr1+2;
```

포인터 연산

포인터가 가리키는 대상의 자료형에 따라서 증가 및 감소되는 값이 차이를 지님

```
#include <stdio.h>

int main(void)
{
    int a; char b; double c;
    int* ptr1=&a;
    char* ptr2=&b;
    double* ptr3=&c;

    printf("%d 번지, %d 번지, %d 번지 \n", ptr1++, ptr2++, ptr3++);
    printf("%d 번지, %d 번지, %d 번지 \n", ptr1, ptr2, ptr3);

    return 0;
}
```

포인터와 배열을 통해서 얻을 수 있는 중대한 결론

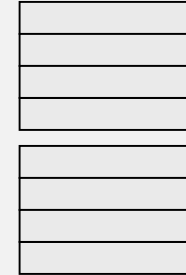
```
#include <stdio.h>

int main(void)
{
    int arr[2]={1, 2};
    int* pArr=arr;

    printf("%d, %d \n", arr[0], *(arr+1));

    printf("%d, %d \n", pArr[0], *(pArr+1));

    return 0;
}
```



$arr[i] == *(arr+i)$

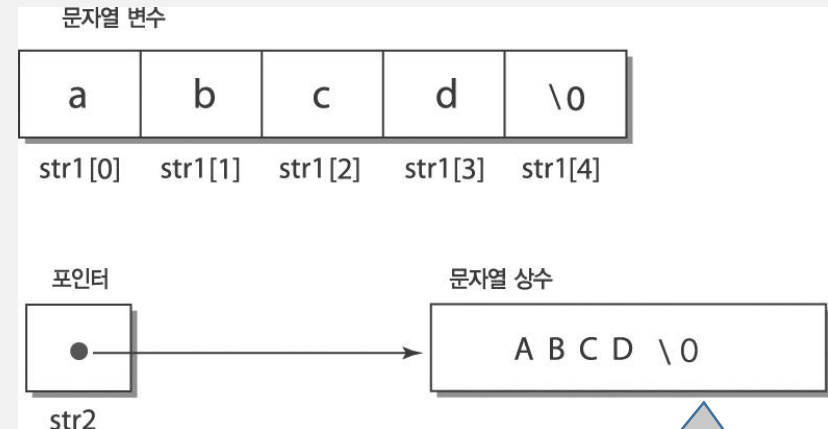
→ arr이 "포인터"이거나 "배열 이름"인 경우

문자열 표현 방식의 이해

배열 기반의 문자열 변수

포인터 기반의 문자열 상수

```
char str1[5]="abcd";  
char *str2="ABCD";
```

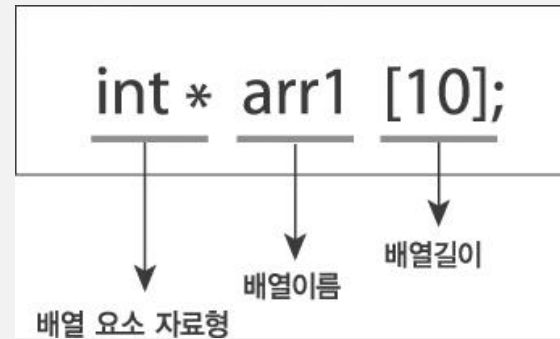


.rodata
메모리 영역

포인터 배열

배열의 요소로 포인터를 지니는 배열

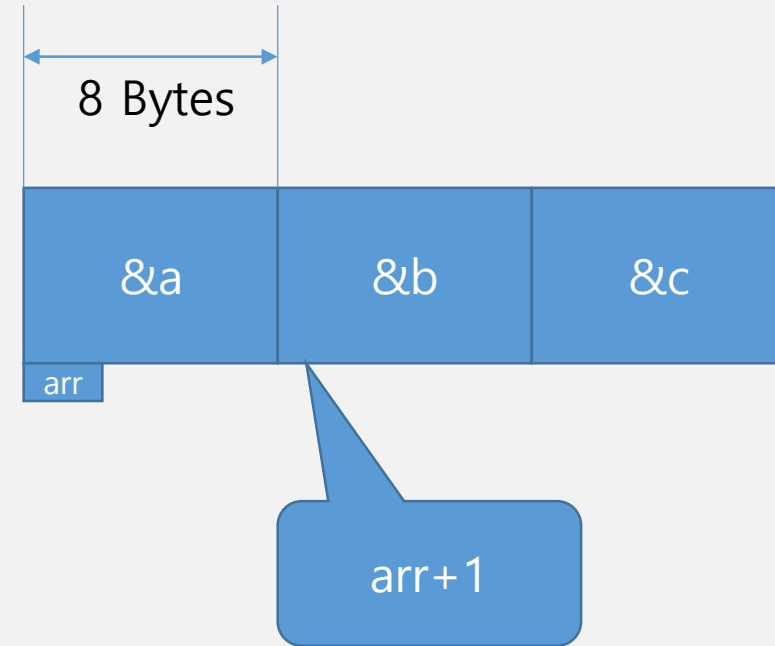
```
int* arr1[10];  
double* arr2[20];  
char* arr3[30];
```



Array of pointers vs Pointer to an Array

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

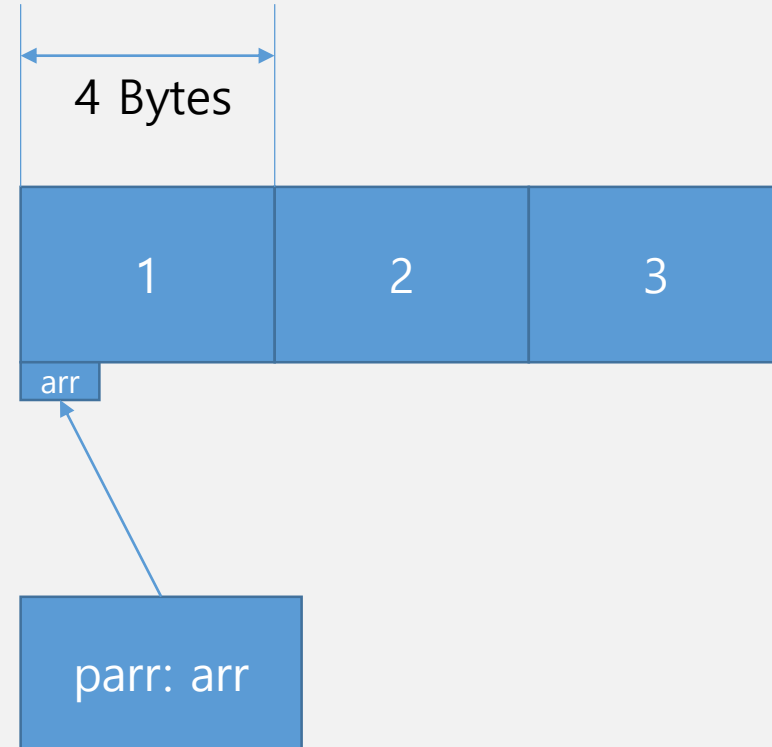
```
int main() {  
    int a=1,b=2,c=3;  
    int* arr[3]={&a,&b,&c};  
    //arr+1??  
    return 0;  
}
```



Array of pointers vs Pointer to an Array

```
#include <stdio.h>

int main() {
    int arr[3]={1,2,3};
    int (*parr)[3]=arr;
    // arr+1 ?
    // parr++ ?
    return 0;
}
```

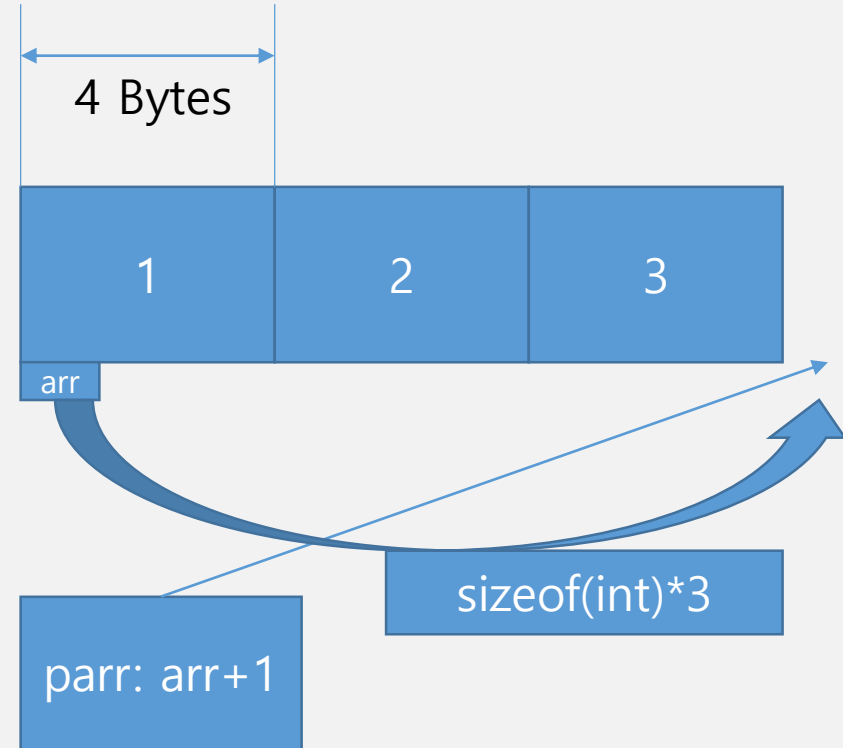


int [3]을 가리키는 포인터

Array of pointers vs Pointer to an Array

```
#include <stdio.h>

int main() {
    int arr[3]={1,2,3};
    int (*parr)[3]=arr;
    // arr+1 ?
    // parr++ ?
    return 0;
}
```



int [3]을 가리키는 포인터

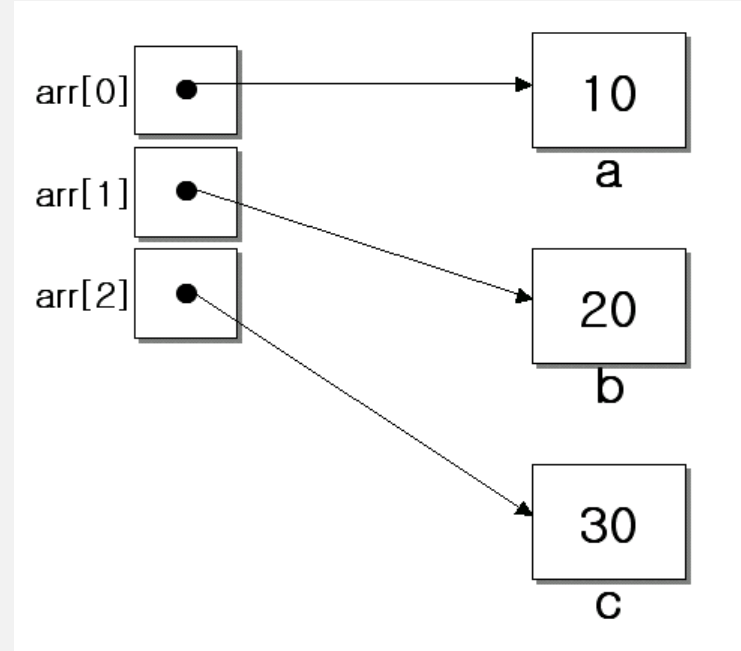
포인터 배열 예제 1

```
#include <stdio.h>

int main(void) {
    int a=10, b=20, c=30;
    int* arr[3]={&a, &b, &c};

    printf("%d \n", *arr[0]);
    printf("%d \n", *arr[1]);
    printf("%d \n", *arr[2]);

    return 0;
}
```



포인터 배열 예제 2

```
#include <stdio.h>

int main(void)
{
    char* arr[3]={
        "Fervent-lecture",
        "TCP/IP",
        "Socket Programming"
    };

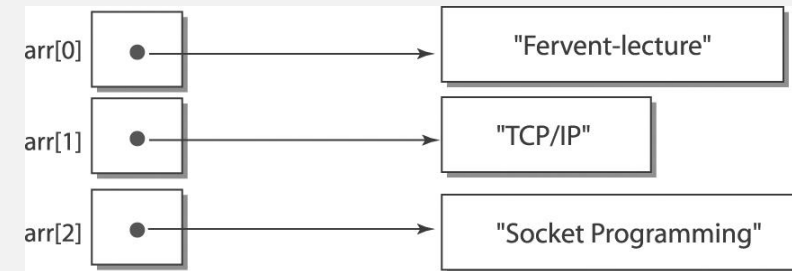
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

```
char * arr[3] = {"Fervent-lectur","TCP/IP","Socket Programming"};
```



```
char * arr[3] = {0x1000, 0x2000, 0x3000};
```



2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

0xF01A	CC	0xF029	CC	0xF038	CC	0xF057	CC
0xF01B	CC	0xF02A	CC	0xF039	CC	0xF058	CC
0xF01C	CC	0xF02B	CC	0xF04A	CC	0xF059	CC
0xF01D	CC	0xF02C	CC	0xF04B	CC	0xF06A	CC
0xF01E	CC	0xF02D	CC	0xF04C	CC	0xF06B	CC
0xF01F	CC	0xF02E	CC	0xF04D	CC	0xF06C	CC
0xF020	CC	0xF02F	CC	0xF04E	CC	0xF06D	CC
0xF021	CC	0xF030	CC	0xF04F	CC	0xF06E	CC
0xF022	CC	0xF031	CC	0xF050	CC	0xF06F	CC
0xF023	CC	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	CC	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	CC	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	CC	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	CC	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	CC	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

0xF01A	'S'	0xF029	CC	0xF038	CC	0xF057	CC
0xF01B	'o'	0xF02A	CC	0xF039	CC	0xF058	CC
0xF01C	'c'	0xF02B	CC	0xF04A	CC	0xF059	CC
0xF01D	'k'	0xF02C	CC	0xF04B	CC	0xF06A	CC
0xF01E	'e'	0xF02D	CC	0xF04C	CC	0xF06B	CC
0xF01F	't'	0xF02E	CC	0xF04D	CC	0xF06C	CC
0xF020	00	0xF02F	CC	0xF04E	CC	0xF06D	CC
0xF021	CC	0xF030	CC	0xF04F	CC	0xF06E	CC
0xF022	CC	0xF031	CC	0xF050	CC	0xF06F	CC
0xF023	CC	0xF032	CC	0xF051	CC	0xF070	CC
0xF024	CC	0xF033	CC	0xF052	CC	0xF071	CC
0xF025	CC	0xF034	CC	0xF053	CC	0xF072	CC
0xF026	CC	0xF035	CC	0xF054	CC	0xF073	CC
0xF027	CC	0xF036	CC	0xF055	CC	0xF074	CC
0xF028	CC	0xF037	CC	0xF056	CC	0xF028	CC

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

“%s”, arr[0]

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

1

“%S”, arr[0]

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

2

“%S”,
arr[0]

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

2

“%S”,
*(arr+0)

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

2

“%S”,
*(0xF021+0)

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

2

“%S”,
*(0xF021)

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

2

“%S”,
*(0xF021)

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

2

“%S”,
0xF057

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

“%s”, 0xF057

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

2 Pointer and Array

```
#include <stdio.h>

int main(void)
{
    char str[]="Socket";
    char* arr[3]={
        "lecture",
        "TCP/IP",
        str
    };
    printf("%s \n", arr[0]);
    printf("%s \n", arr[1]);
    printf("%s \n", arr[2]);

    return 0;
}
```

“lecture”

0xF01A	'S'	0xF029	6F	0xF038	00	0xF057	'l'
0xF01B	'o'	0xF02A	F0	0xF039	CC	0xF058	'e'
0xF01C	'c'	0xF02B	00	0xF04A	CC	0xF059	'c'
0xF01D	'k'	0xF02C	00	0xF04B	CC	0xF06A	't'
0xF01E	'e'	0xF02D	00	0xF04C	CC	0xF06B	'u'
0xF01F	't'	0xF02E	00	0xF04D	CC	0xF06C	'r'
0xF020	00	0xF02F	00	0xF04E	CC	0xF06D	'e'
0xF021	57	0xF030	00	0xF04F	CC	0xF06E	00
0xF022	F0	0xF031	1A	0xF050	CC	0xF06F	'T'
0xF023	00	0xF032	F0	0xF051	CC	0xF070	'C'
0xF024	00	0xF033	00	0xF052	CC	0xF071	'P'
0xF025	00	0xF034	00	0xF053	CC	0xF072	'/'
0xF026	00	0xF035	00	0xF054	CC	0xF073	'I'
0xF027	00	0xF036	00	0xF055	CC	0xF074	'P'
0xF028	00	0xF037	00	0xF056	CC	0xF028	00

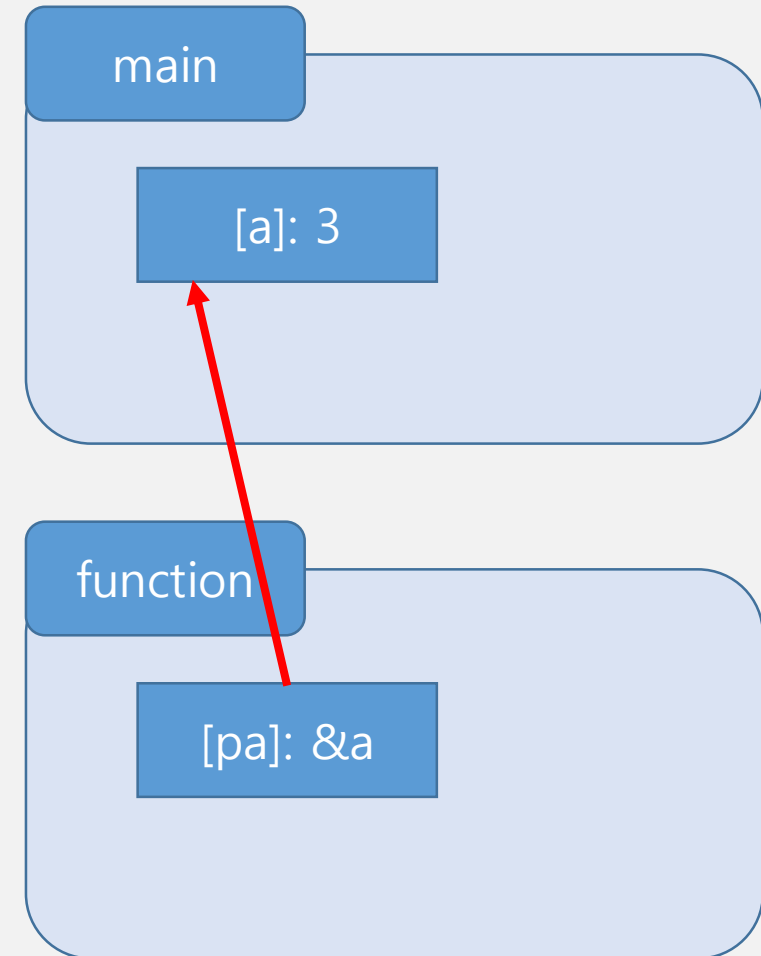
1. Pointer Parameter

```
#include <stdio.h>

void function(int*);

int main() {
    int a=3;
    function(&a);
    return 0;
}

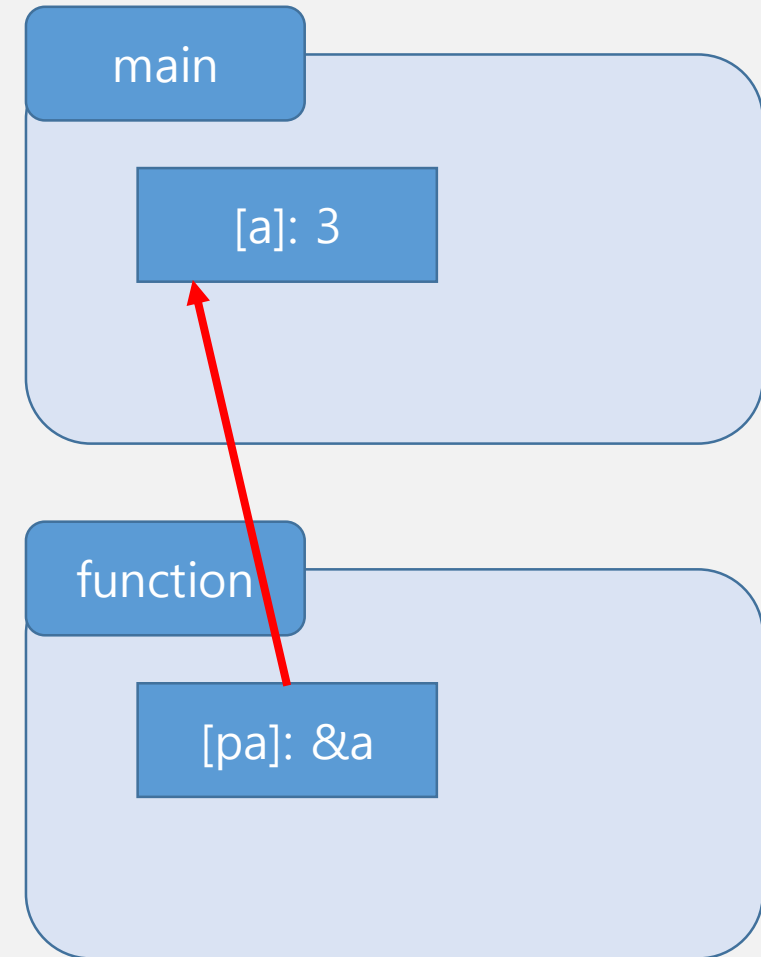
void function(int* pa) {
    printf("%d\n",*pa);
}
```



2. Array Parameter

```
#include <stdio.h>

void function(int []);
int main() {
    int a=3;
    function(&a);
    return 0;
}
void function(int pa[]) {
    printf("%d\n", pa[0]);
}
```



2. Array Parameter

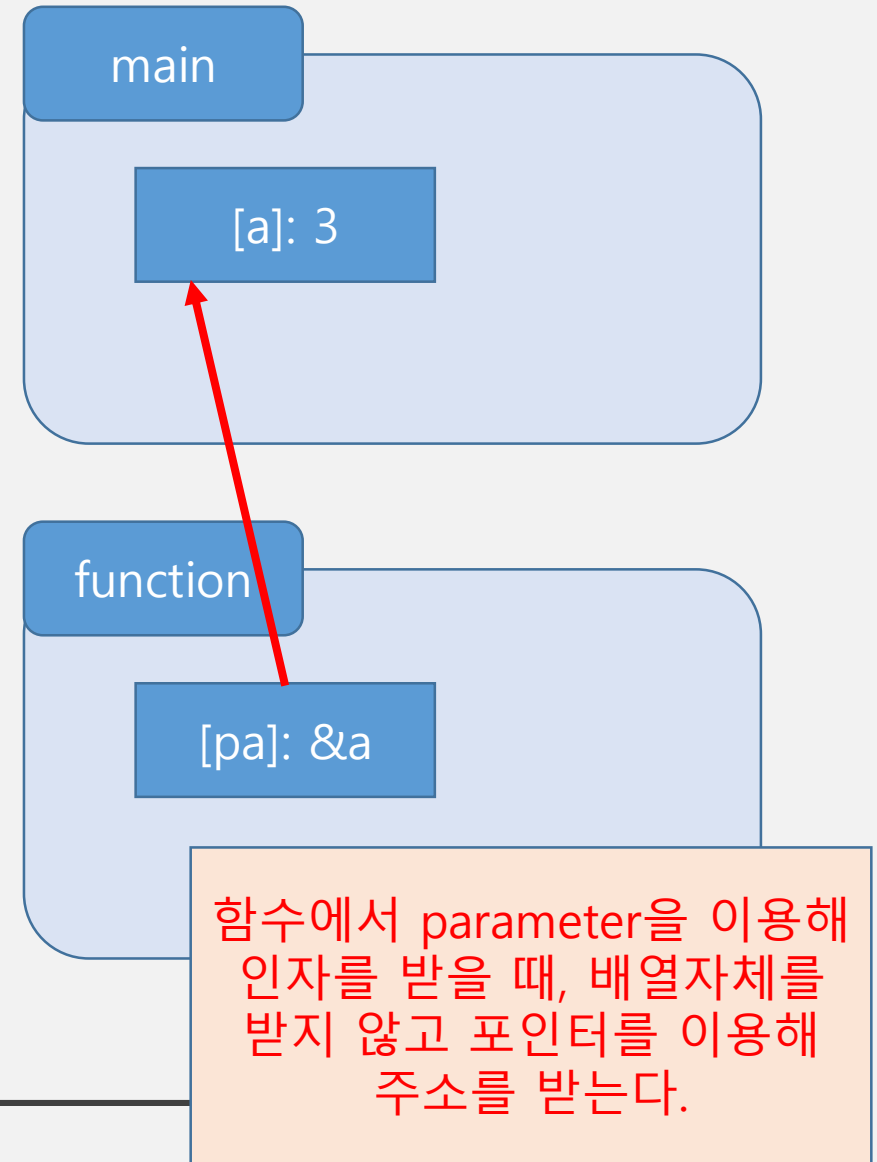
```
#include <stdio.h>

void function(int []);

int main() {
    int a=3;
    function(&a);
    return 0;
}

void function(int pa[]) {
    printf("%d\n", pa[0]);
}
```

parameter에서
포인터와 배열은 같다.
 $\text{int}^* \text{pa} == \text{int pa}[]$



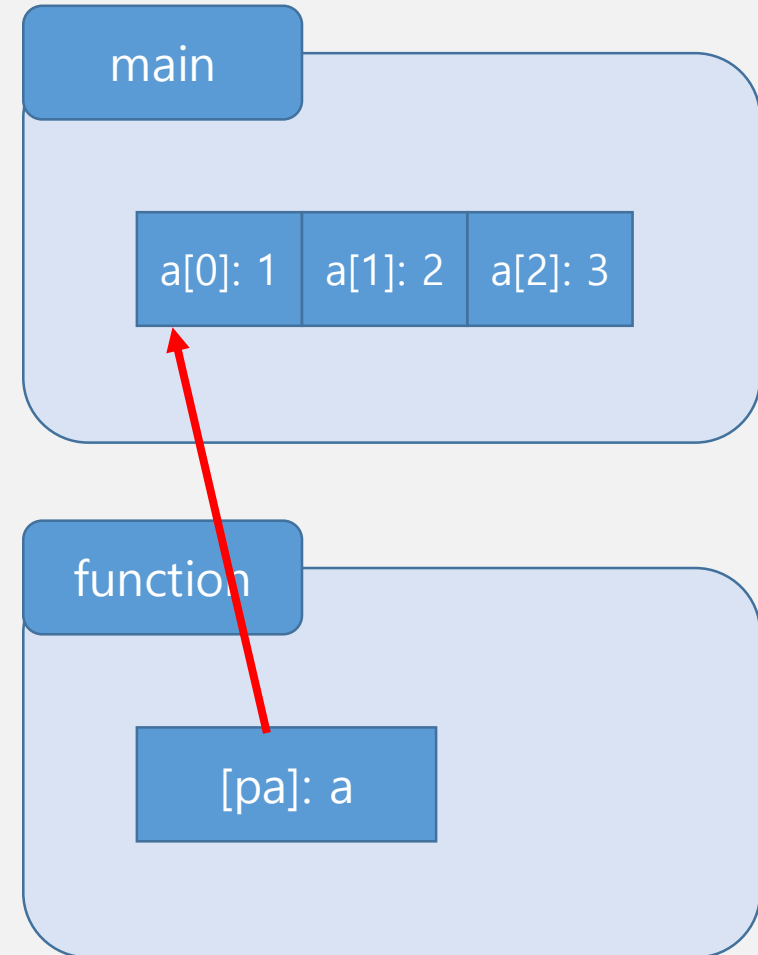
3. Pointer Parameter & Array argument

```
#include <stdio.h>

void function(int*);

int main() {
    int a[3]={1,2,3};
    function(a);
    return 0;
}

void function(int* pa) {
    printf("%d\n",*pa);
    pa++;
}
```

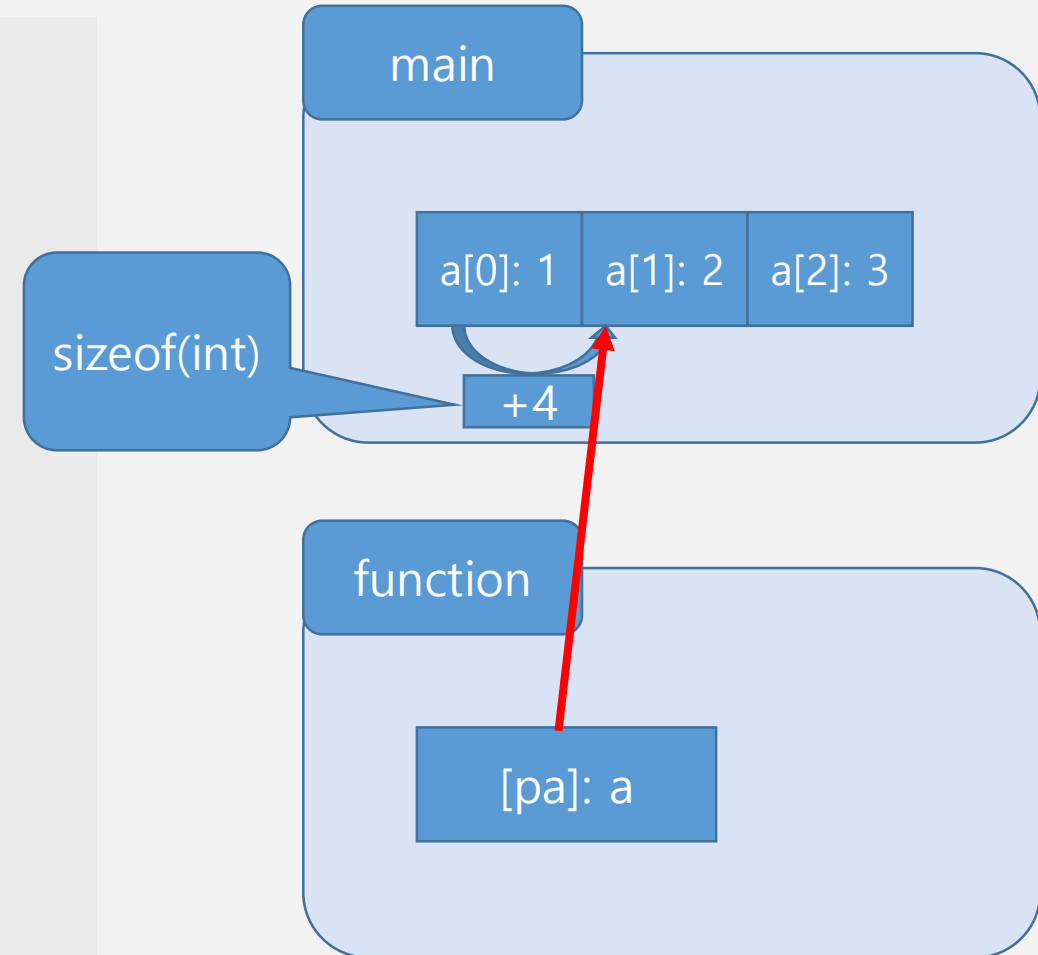


3. Pointer Parameter & Array argument

```
#include <stdio.h>

void function(int*);
int main() {
    int a[3]={1,2,3};
    function(a);
    return 0;
}

void function(int* pa) {
    printf("%d\n",*pa);
    pa++;
}
```



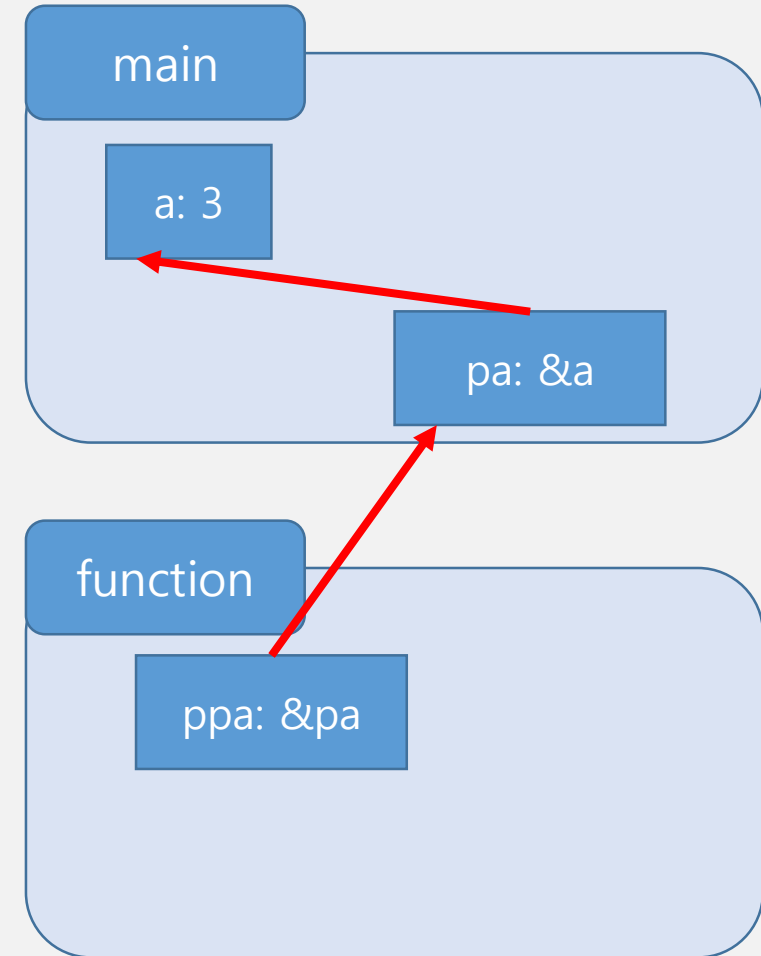
4. Pointer to Pointer Parameter

```
#include <stdio.h>

void function(int*);

int main() {
    int a=3;
    int* pa=&a;
    function(&pa);
    return 0;
}

void function(int** ppa) {
    printf("%d\n",**ppa);
    ppa++;
}
```



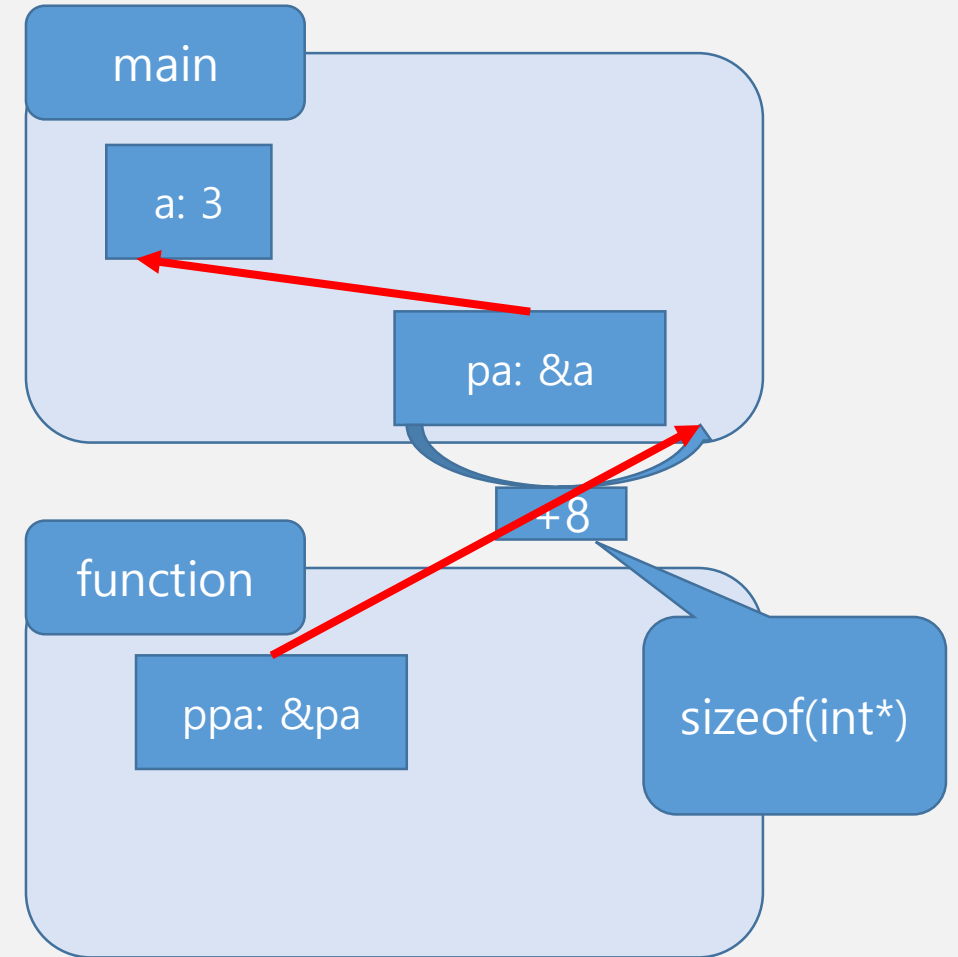
4. Pointer to Pointer Parameter

```
#include <stdio.h>

void function(int*);

int main() {
    int a=3;
    int* pa=&a;
    function(&pa);
    return 0;
}

void function(int** ppa) {
    printf("%d\n",**ppa);
    ppa++;
}
```



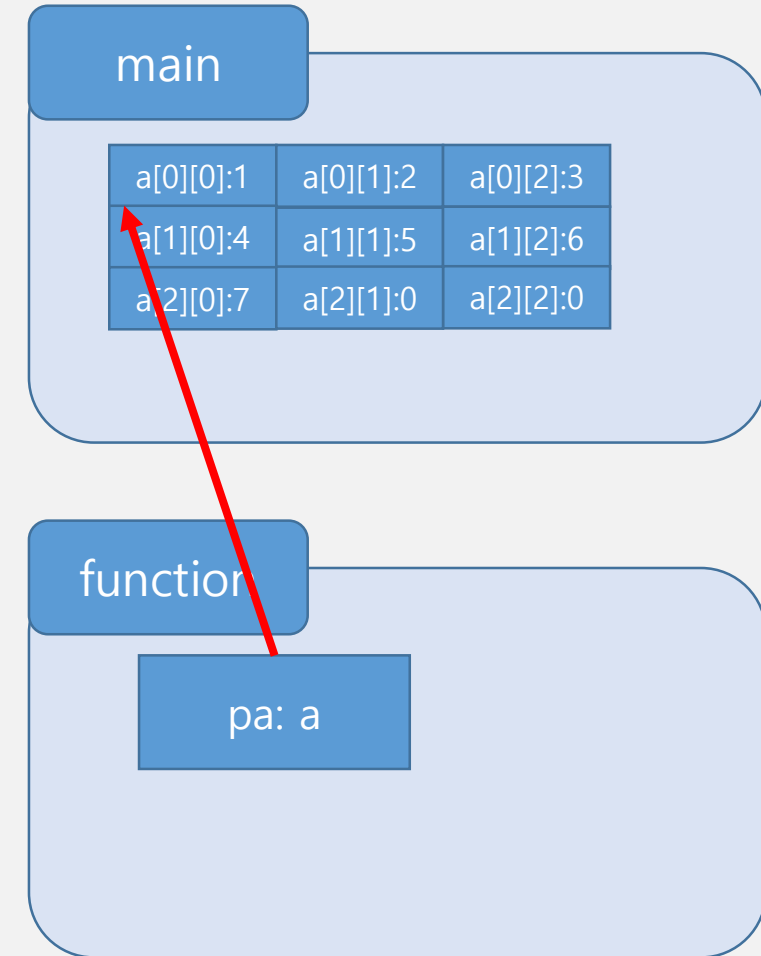
5. 2 Dimensional Array Parameter

```
#include <stdio.h>

void function(int[][3]);

int main() {
    int a[3][3]
        = {1,2,3,4,5,6,7};
    function(a);
    return 0;
}

void function(int pa[][3]) {
    printf("%d\n", *pa);
    pa++;
}
```



5. 2 Dimensional Array Parameter

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

```
void function(int[][3]);
```

```
int main() {
```

```
    int a[3][3]
```

```
        = {1,2,3,4,5,6,7};
```

```
    function(a);
```

```
    return 0;
```

```
}
```

```
void function(int pa[][3]) {
```

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

```
    pa++;
```

```
}
```

int pa[][3]
== int (*pa)[3]

main

+12

sizeof(int)
* 3

a[0][0]:1	a[0][1]:2	a[0][2]:3
a[1][0]:4	a[1][1]:5	a[1][2]:6
a[2][0]:7	a[2][1]:0	a[2][2]:0

function

pa: a

sizeof(a) ??
sizeof(pa) ??

6. 2 Dimensional Array Argument & Pointer to Pointer Parameter

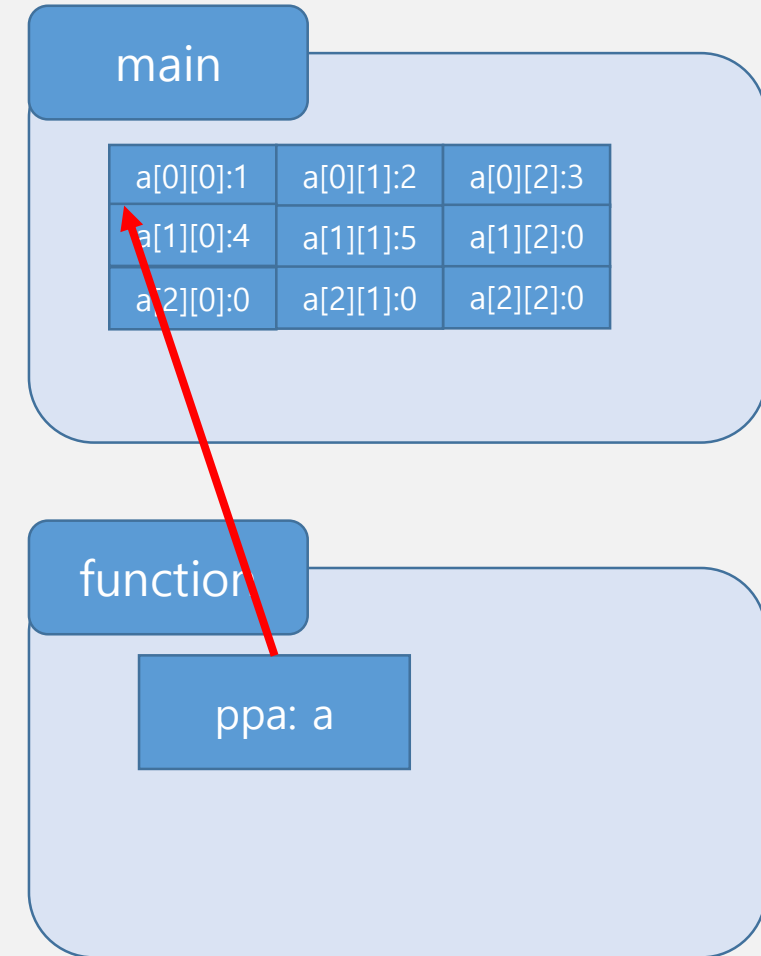
```
#include <stdio.h>

void function(int*);

int main() {
    int a[3][3]={1,2,3,4,5};
    function(a);
    return 0;
}

void function(int** ppa) {
    printf("%d\n",**ppa);
    ppa++;
    printf("%d\n",**ppa);
    ppa++;
    printf("%d\n",**ppa);
    ppa++;
}
```

pointer
to int*



6. 2 Dimensional Array Argument & Pointer to Pointer Parameter

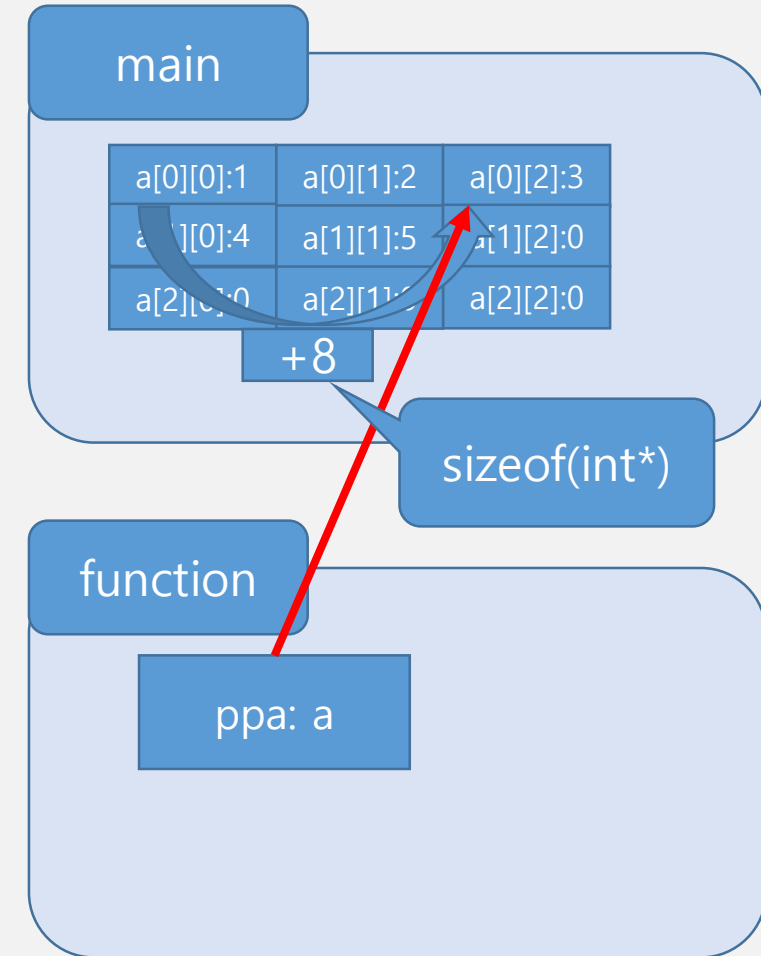
```
#include <stdio.h>

void function(int*);

int main() {
    int a[3][3]={1,2,3,4,5};
    function(a);
    return 0;
}

void function(int** ppa) {
    printf("%d\n",**ppa);
    ppa++;
    printf("%d\n",**ppa);
    ppa++;
    printf("%d\n",**ppa);
    ppa++;
}
```

pointer
to int*



~~6. 2 Dimensional Array Argument & Pointer to Pointer Parameter~~

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

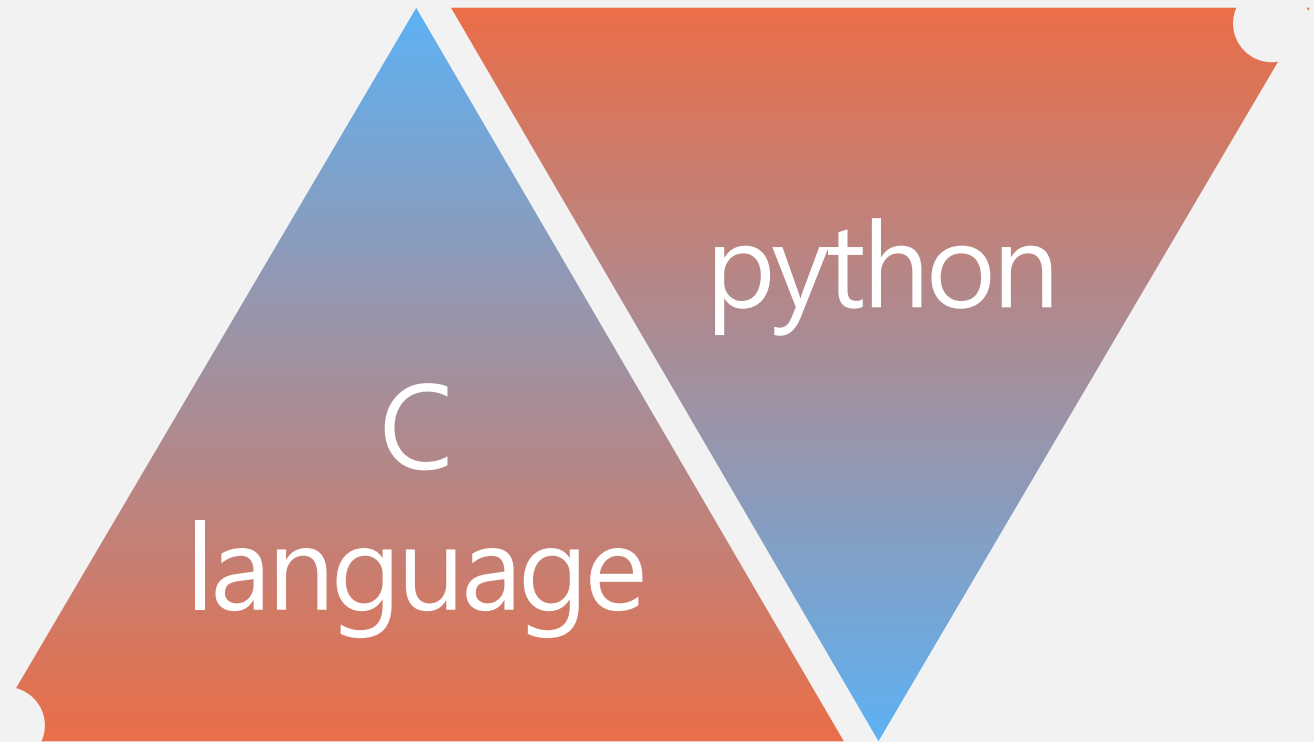
main

<COMPILE ERROR>

int [][3]과 int**는 다르다!

64-bit 환경에서
int [][2]와 int**도 다르다!

3. 기말고사 대비



Quiz

```
show_array(const double ar[], int n);
mult_array(double arr[], int n, double mult);
-----

double rates[5] = {88.99 , 100.12};
const double locked[4] = {0.08, 0.075, 0.725, 0.07};
const double *pc=rates;
*pc=29.89;
pc[2]=241;
pc++;
pc = locked;
pc = &rates[3];
show_array(rates,5);
show_array(locked,4);
mult_array(rates,5,1.2);
mult_array(locked,4,1.2);
```

Quiz

```
show_array(const double ar[], int n);
mult_array(double arr[], int n, double mult);
-----

double rates[5] = {88.99 , 100.12};
const double locked[4] = {0.08, 0.075, 0.725, 0.07};
double * const pc=rates;
*pc=29.89;
pc[2]=241;
pc++;
pc = locked;
pc = &rates[3];
show_array(rates,5);
show_array(locked,4);
mult_array(rates,5,1.2);
mult_array(locked,4,1.2);
```

Quiz

```
show_array(const double ar[], int n);
mult_array(double arr[], int n, double mult);
-----

double rates[5] = {88.99 , 100.12};
const double locked[4] = {0.08, 0.075, 0.725, 0.07};
const double * const pc=rates;
*pc=29.89;
pc[2]=241;
pc++;
pc = locked;
pc = &rates[3];
show_array(rates,5);
show_array(locked,4);
mult_array(rates,5,1.2);
mult_array(locked,4,1.2);
```

Quiz

```
int * pt;  
int (*pa)[3];  
int ar1[2][3];  
int ar2[3][2];  
int **p2;
```

```
-----  
pt = &ar1[0][0];  
pt = &ar1[0];  
pt = ar1;  
pa = ar1;  
pa = ar2;  
p2 = &pt;  
*p2 = ar2[0];  
p2 = ar2;
```

Quiz

```
int * p1;  
const int* p2;  
const int** pp2;  
const int n= 13;  
-----  
  
p1 = p2;  
p2 = p1;  
pp2 = &p1;  
*pp2 = &n;  
*p1=10;
```

1. 포인터 개념 확인
2. 2012년도 기말고사
3. 2014년도 기말고사