

```
/* arr_adder.c */
#include <stdio.h>

int ArrAdder(int* pArr, int n);

int main(void)
{
    int arr[10]={1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    int SumOfArr;

    SumOfArr=ArrAdder(arr, sizeof(arr)/sizeof(int));
    printf("배열의 총 합 : %d \n", SumOfArr);

    return 0;
}

int ArrAdder(int* pArr, int n)
{
    int sum=0;
    int i;

    for(i=0; i<n; i++)
        sum+=pArr[i];

    return sum;
}
```

```

/* fct_ptr1.c */
#include <stdio.h>

void Add(int a, int b);
void SPrint(char * str);

int main(void)
{
    char * string = "Function Pointer";
    int a=10, b=20;

    void (*fPtr1)(int, int)=Add;
    void (*fPtr2)(char*)=SPrint;

    /* 함수 포인터에 의한 호출*/
    fPtr1(a, b);
    fPtr2(string);

    return 0;
}

void Add(int a, int b)
{
    printf("덧셈 결과: %d \n", a+b);
}

void SPrint(char * str)
{
    printf("입력된 문자열: %s \n", str);
}

```

```

/* fct_ptr2.c */
#include <stdio.h>

void SelFunction(int s);
void Add(void);
void Min(void);

int main(void)
{
    int sel;

    while(1)
    {
        printf("선택 : 덧셈(1), 뺄셈(2), 종료(3) ");
        scanf("%d", &sel);
        if(sel==3)
            break;
        SelFunction(sel);
    }
    printf("프로그램이 종료 되었습니다. \n");

    return 0;
}

void SelFunction(int s)
{
    void (*fPtr)(void);
    if(s==1)
        fPtr=Add;
    else
        fPtr=Min;
    fPtr();
}

void Add(void)
{
    int a, b;

```

```
printf("덧셈을 위한 두 개의 숫자 입력 :");  
scanf("%d %d", &a, &b);  
printf("덧셈 결과 : %d \n\n", a+b);  
}  
void Min(void)  
{  
    int a, b;  
    printf("뺄셈을 위한 두 개의 숫자 입력 :");  
    scanf("%d %d", &a, &b);  
    printf("뺄셈 결과 : %d \n\n", a-b);  
}
```

```

/* ProbArray2.c */
#include <stdio.h>
#include <stdlib.h>

void function(int);
int main (void)
{
    int m=0;
    fputs("배열의 크기를 입력하세요: ", stdout);
    scanf("%d", &m);
    function(m);

    return 0;
}

void function(int i)
{
    int* array =(int*)malloc(sizeof(int)*i); //동적 메모리 할당.
    int j;
    if(array==NULL)
    {
        puts("메모리 할당에 실패");
        exit(1);
    }
    /* 동적 할당한 메모리 사용 */
    for(j=0; j<i; j++)
        array[j]=j+1;
    for(j=0; j<i; j++)
        printf("%d ", array[j]);
    printf("\n");

    free(array); // 할당된 메모리 소멸.
}

```