

Group Activity 01

(3인 혹은 4인으로 팀을 구성하여 아래의 문제를 푼다. 팀 구성은 매 시간마다 달라져도 된다.)

팀원1: _____

팀원2: _____

팀원3: _____

팀원4: _____

1. `scanf`와 `scanf_s`의 차이점은 무엇인가? MicroSoft에서 `scanf` 대신 `scanf_s`를 권장하는 이유는?

2. 다음 프로그램의 출력은? 이유는? 만약 컴파일 오류나 런타임 오류가 난다면 그 이유는?

(1)

```
#include <stdio.h>
void fun(int x)
{
    x = 30;
}

int main()
{
    int y = 20;
    fun(y);
    printf("%d", y);
    return 0;
}
```

```

(2)
#include <stdio.h>
void fun(int *ptr)
{
    *ptr = 30;
}

int main()
{
    int y = 20;
    fun(&y);
    printf("%d", y);
    return 0;
}

```

```

(3)
#include <stdio.h>
int main()
{
    int *ptr;
    int x;

    ptr = &x;
    *ptr = 0;

    printf(" x = %d\n", x);
    printf(" *ptr = %d\n", *ptr);

    *ptr += 5;
    printf(" x = %d\n", x);
    printf(" *ptr = %d\n", *ptr);

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

    return 0;
}

```

```

(4)
#include <stdio.h>
int main()
{
    int arri[] = {1, 2 ,3};
    int *ptri = arri;

    char arrc[] = {1, 2 ,3};
    char *ptrc = arrc;

    printf("sizeof arri[] = %d ", sizeof(arri));
    printf("sizeof ptri = %d ", sizeof(ptri));

    printf("sizeof arrc[] = %d ", sizeof(arrc));
    printf("sizeof ptrc = %d ", sizeof(ptrc));

    return 0;
}

```

```

(5)
#include <stdio.h>
int main()
{
    float arr[5] = {12.5, 10.0, 13.5, 90.5, 0.5};
    float *ptr1 = &arr[0];
    float *ptr2 = ptr1 + 3;

    printf("%f ", *ptr2);
    printf("%d", ptr2 - ptr1);

    return 0;
}

```

```

(6)
#include<stdio.h>
int main()
{
    int arr[] = {10, 20, 30, 40, 50, 60};
    int *ptr1 = arr;
    int *ptr2 = arr + 5;
    printf("Number of elements between two pointer are: %d.", (ptr2 - ptr1));
    printf("Number of bytes between two pointers are: %d",
           (char*)ptr2 - (char*) ptr1);

    return 0;
}

```



```

(7)
#include<stdio.h>
int main()
{
    int a;
    char *x;
    x = (char *) &a;
    a = 512;
    x[0] = 1;
    x[1] = 2;
    printf("%d\n",a);
    return 0;
}

```



```

(8)
#include<stdio.h>
void fun(int arr[])
{
    int i;
    int arr_size = sizeof(arr)/sizeof(arr[0]);
    for (i = 0; i < arr_size; i++)
        printf("%d ", arr[i]);
}

int main()
{
    int i;
    int arr[4] = {10, 20 ,30, 40};
    fun(arr);
    return 0;
}

```

```

(9)
#include<stdio.h>
void f(int *p, int *q)
{
    p = q;
    *p = 2;
}

int i = 0, j = 1;
int main()
{
    f(&i, &j);
    printf("%d %d \n", i, j);
    return 0;
}

```