

Group Activity 01

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

팀원1: _____

팀원2: _____

팀원3: _____

팀원4: _____

다음 프로그램들의 출력은? 컴파일 오류나 실행 오류가 나는 경우에는 이유를 간략히 설명하라.

Program	Output
<pre>#include <iostream> using namespace std; int main () { int firstvalue, secondvalue; int *mypointer; mypointer = &firstvalue; *mypointer = 10; mypointer = &secondvalue; *mypointer = 20; cout << "firstvalue is " << firstvalue << '\n'; cout << "secondvalue is " << secondvalue << '\n'; return 0; }</pre>	
<pre>#include <iostream> using namespace std; int main () { int firstvalue = 5, secondvalue = 15; int *p1, *p2; p1 = &firstvalue; p2 = &secondvalue; *p1 = 10; *p2 = *p1; p1 = p2; *p1 = 20; cout << "firstvalue is " << firstvalue << '\n'; cout << "secondvalue is " << secondvalue << '\n'; return 0; }</pre>	

```

#include <iostream>
using namespace std;

int main () {
    int numbers[5];
    int * p;
    p = numbers; *p = 10;
    p++; *p = 20;
    p = &numbers[2]; *p = 30;
    p = numbers + 3; *p = 40;
    p = numbers; *(p+4) = 50;
    for (int n=0; n<5; n++)
        cout << numbers[n] << ", ";
    return 0;
}

```

```

#include <iostream>
using namespace std;

void increment_all(int* start, int* stop) {
    int *current = start;
    while(current != stop) {
        ++(*current);
        ++current;
    }
}

void print_all(const int* start, const int* stop) {
    const int *current = start;
    while (current != stop) {
        cout << *current << '\n';
        ++current;
    }
}

int main() {
    int numbers[] = {10,20,30};
    increment_all(numbers,numbers+3);
    print_all(numbers,numbers+3);
    return 0;
}

```

```

#include <iostream>
using namespace std;

void increase(void* data, int psize) {
    if (psize == sizeof(char)) {
        char* pchar;
        pchar=(char*)data;
        ++(*pchar);
    }
    else if (psize == sizeof(int)) {
        int* pint;
        pint=(int*)data;
        ++(*pint);
    }
}

int main() {
    char a = 'x';
    int b = 1602;
    increase (&a, sizeof(a));
    increase (&b, sizeof(b));
    cout << a << ", " << b << '\n';
    return 0;
}

```

```

#include <iostream>
using namespace std;

int addition(int a, int b) {
    return a+b;
}

int subtraction(int a, int b) {
    return a-b;
}

int operation(int x, int y, int (*functocall)(int,int)) {
    int g;
    g = (*functocall)(x,y);
    return (g);
}

int main() {
    int m,n;
    int (*minus)(int,int) = subtraction;

    m = operation (7, 5, addition);
    n = operation (20, m, minus);
    cout <<n;
    return 0;
}

```

<pre> #include <iostream> #include <string> int main() { std::string str("Test string"); std::cout << str.size() << "\n"; return 0; } </pre>	
<pre> #include <iostream> #include <string> int main() { std::string str ("Test string"); for (int i=0; i<str.length(); ++i) { std::cout << str[i]; } return 0; } </pre>	
<pre> #include <iostream> #include <string> int main() { std::string str ("Test string"); for (unsigned i=0; i<str.length(); ++i) std::cout << str.at(i); return 0; } </pre>	
<pre> #include <iostream> #include <string> int main() { std::string name ("John"); std::string family ("Smith"); name += " K. "; name += family; name += '\n'; std::cout << name; return 0; } </pre>	
<pre> #include <iostream> #include <string> int main() { std::string str; std::string str2="Writing "; std::string str3="print 10 and then 5 more"; str.append(str2); str.append(str3, 6, 3); str.append("dots are cool", 5); std::cout << str << '\n'; return 0; } </pre>	

<pre>#include <iostream> #include <string> int main() { std::string str="to be question"; std::string str2="the "; str.insert(6, str2); std::cout << str << '\n'; return 0; }</pre>	
<pre>#include <iostream> #include <string> int main() { std::string str ("This is an example sentence."); std::cout << str << '\n'; str.erase (10, 8); std::cout << str << '\n'; str.erase (str.begin()+9); std::cout << str << '\n'; str.erase (str.begin()+5, str.end()-9); std::cout << str << '\n'; return 0; }</pre>	
<pre>#include <iostream> #include <string> using namespace std; int main() { string str="this is a test string."; string str2="n example"; string str3="sample phrase"; str.replace(9,5,str2); cout << str << '\n'; str.replace(str.begin(),str.end()-3,str3); cout << str << '\n'; return 0; }</pre>	
<pre>#include <iostream> #include <string> int main() { std::string str ("hello world!"); str.pop_back(); std::cout << str << '\n'; return 0; }</pre>	

```

#include <iostream>
#include <cstring>
#include <string>
using namespace std;

int main() {
    string str ("Please split this sentence into tokens");

    char *cstr = new char [str.length()+1];
    strcpy (cstr, str.c_str());

    char *p = std::strtok (cstr, " ");
    while (p!=0) {
        cout << p << '\n';
        p = std::strtok(NULL, " ");
    }

    delete[] cstr;
    return 0;
}

```

```

#include <iostream>
#include <string>
using namespace std;

int main() {
    string str("There are two needles with needles.");
    string str2("needle");

    size_t found = str.find(str2);
    if (found!=string::npos)
        cout << found << '\n';

    found=str.find("needles are small",found+1,6);
    if (found!=string::npos)
        cout << found << '\n';

    str.replace(str.find(str2),str2.length(),
               "preposition");
    cout << str << '\n';

    return 0;
}

```

```
#include <iostream>
#include <string>
using namespace std;
```

```
int main() {
    string str ("Please, replace the vowels in this
sentence by asterisks.");
    size_t found = str.find_first_of("aeiou");
    while (found!=string::npos){
        str[found]='*';
        found=str.find_first_of("aeiou",found+1);
    }

    cout << str << '\n';
    return 0;
}
```

```
#include <iostream>
#include <string>
using namespace std;
```

```
int main() {
    string str="We think in generalities, but we live in
details.";
    string str2 = str.substr (3,5);
    size_t pos = str.find("live");
    string str3 = str.substr(pos);
    cout << str2 << ' ' << str3 << '\n';

    return 0;
}
```

```
#include <iostream>
#include <string>
using namespace std;
```

```
int main (){
    string str1 ("green apple");
    string str2 ("red apple");

    if (str1.compare(str2) != 0)
        cout << str1 << " is not " << str2 << '\n';

    if (str1.compare(6,5,"apple") == 0)
        cout << "still, " << str1 << " is an apple\n";

    if (str2.compare(str2.size()-5,5,"apple") == 0)
        cout << "and " << str2 << " is also an apple\n";

    if (str1.compare(6,5,str2,4,5) == 0)
        cout << "therefore, both are apples\n";

    return 0;
}
```