Exercise (Page 66~68)

1. Given the following Program 3.7, answer the given questions.

```
// Program 3.7
#include <iostream>
using namespace std;

int fun1(int n)
{
    if (n <= 0)
        return 0;
    else if (n <= 1)
        return 1;
    else
        return fun1(n - 1) + fun1(n - 2);
}

int main()
{
    cout << fun1(5) << endl;
    system("pause");
    return 0;
}
```

a. Identify the statement in function fun1() according to the following recursive principle.

   i. Base Case / Terminal case

   ii. Recursive case

b. Determine the output of the program and illustrate the recursive trace that leads to this.

2. Given the following Program 3.8, answer the given questions.

```
// Program 3.8
#include <iostream>
using namespace std;

int fun2(int, int);
int main()
{
    cout << fun2(7, 3);
    system("pause");
    return 0;
}

int fun2(int x, int n)
{
    if (n == 0)
        return 1;
    else
        return x * fun2(x, n - 1);
}
```

a. Determine the output for the above program and trace the recursive function.

Exercise (Page 66~68)

b. Describe the problem solved by the recursive function and identify the simple solution and recursive process.

3. Given two recursive functions as followed, answer the given questions.

```
// Program 3.9
void function_01(int n)
{
    cout << "Calling function_01\n";
    if (n < 5)
        function_01(n - 1);
}
```

```
// Program 3.10
void function_02(int n)
{
    if (n > 1)
        function_02(n - 2);
    cout << n << " ";
}
```

a. Write two complete C++ programs, the first program will call function_01() with parameter value equal to 3 and the second program will call function_02() with parameter value also equal 3.
b. When both programs are executed, one of them will produce an error or incorrect behavior. Identify the problem in one of the programs.
c. Show the recursive trace for the error-free program.

4. Given the following programs, identify the output of the programs and trace the recursive functions.

a. Greatest Common Divisor (GCD) Function

```
// Program 3.11
#include <iostream>
using namespace std;

int GCD(int a, int b);
int main()
{
    int first = 3, second = 8;
    cout << GCD(first, second) << endl;
    system("pause");
    return 0;
}

int GCD(int a, int b)
{
```

Exercise (Page 66~68)

```
    if (a % b == 0)   // BASE CASE
        return b;
    else               // RECURSIVE CALL
        return GCD(b, a % b);
}
```

b. Product function

```
// Program 3.12
#include <iostream>
using namespace std;

int Calc(int n)
{
    if (n < 0)
        return n;
    else
        return Calc(n - 1) * Calc(n - 2);
}

int main()
{
    cout << Calc(5) << endl;
    system("pause");
    return 0;
}
```