Programming Design, Spring 2015
Suggested Solution for Homework 4

Solution provider: Shelley Sun

Problem 1

If the function signature includes the function return type, there might be some cases that the compiler would be unable to distinguish which function should be invoked. For example, if there are two functions that differ solely in the return type:

```
int example();
void example();
```

and in the main function, you invoke:

```
int x = example();
```

the compiler would be able to recognized that you are calling the first function.

However, if you call:

```
example();
```

the compiler has no idea which one of the two functions you wanted to invoke, because it is possible to call a function returning an integer and discard the result. To avoid ambiguities like this, the function signature should not include the function return type.

Problem 2

(a)

We said that the function “couples” calculation with the output process, by the meaning that the calculation and the output process would be done together when the function is invoked.

Since we may want the two processes to separate with each other, we can modify the function to return the result value, and do the output process after invoking the function. By this way, we can “decouple” calculation with the output process.

(b)

```
bool prime(int n)
{
    bool isPrime = true;
    for(int i = 2; i * i <= n; i++)
    {
        if(n % i == 0)
        {
            isPrime = false;
            break;
        }
    }
    return isPrime;
}
```

After modifying the function, we can invoke the function in the main function to receive the Boolean value, and do the output process.

```
    bool p = prime(m);
    cout << p << "\n";
```

(c)

```
bool prime(int n)
{
    for(int i = 2; i * i <= n; i++)
    {
        if(n % i == 0)
        {
            return false;
        }
    }
    return true;
}
```

Using two return statements would be better. Because if we can return the value earlier, we can skip some redundant statements and save more time.

Problem 3

Please see the .cpp file.

Problem 4

Please see the .cpp file.