**Observations report**

**Name: Jihui. Sheng**

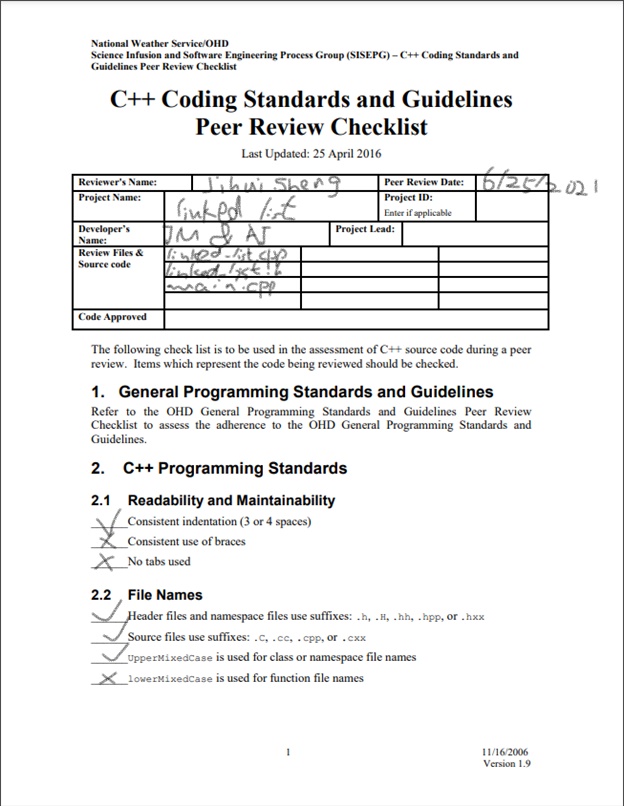
**ID: 11539324**

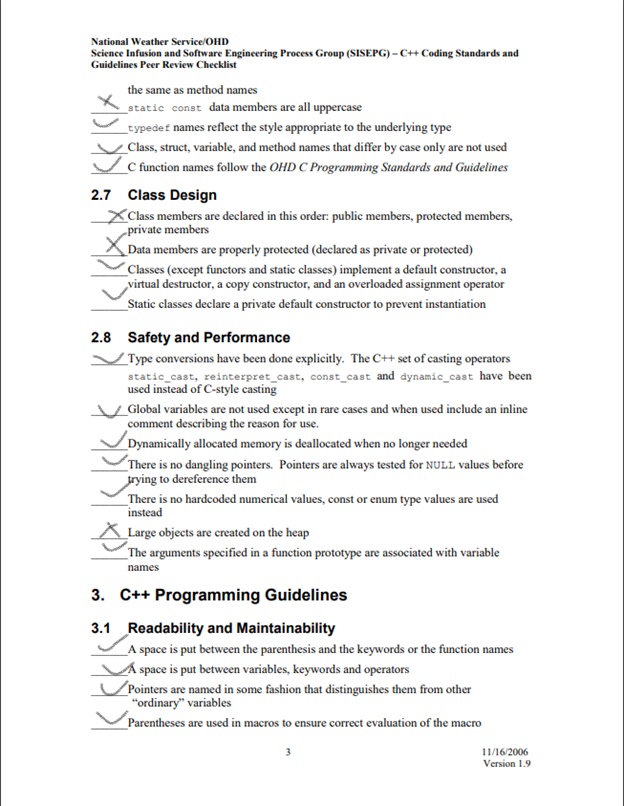
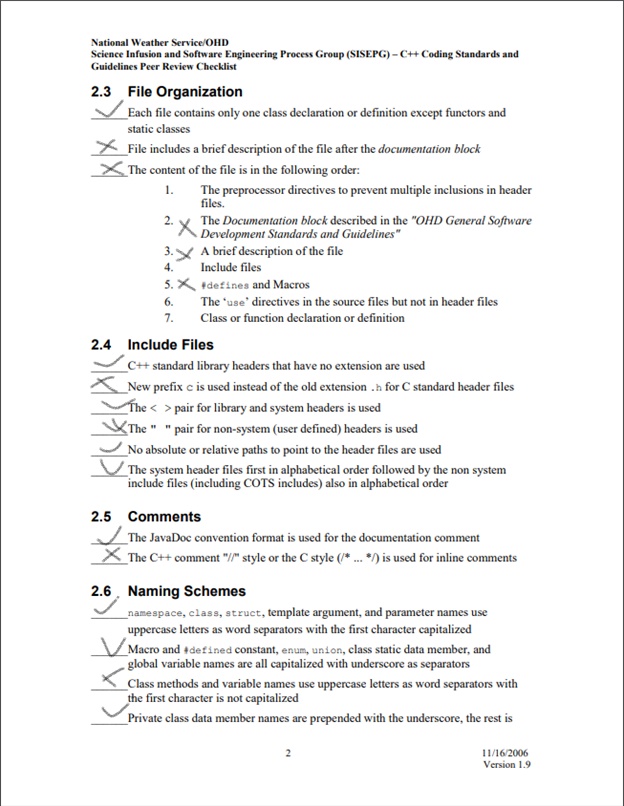
**Course: Cpt\_S 427**

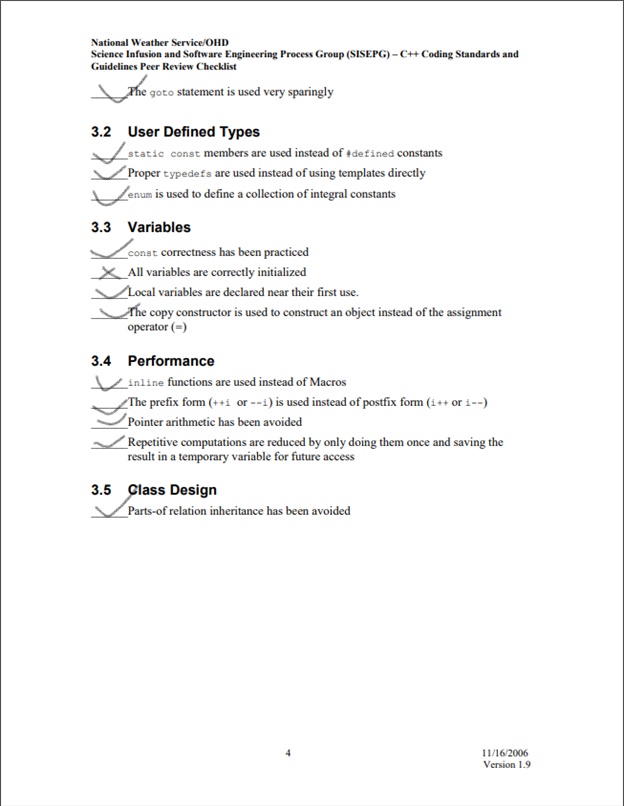
Deliverable:

An 'observations report'. The report should include the following content:

* Screenshots proving you did perform the tutorial activities (code review exercise checklist)
* Report any bugs, typos, broken links etc
* A brief discussion on the skills you've learned from the tutorial (7 lines maximum)

**Screenshots:** 

****

****

**Report any bugs, typos, broken links etc:**

**main.cpp**

1.

class UberNode : public int\_list

**Problem:** not a class or struct name.

UberNode \*pRoot = new UberNode;

**Problem:** "UberNode::UberNode()" is inaccessible.

pNew->add\_value("COUNT", count);

**Problem:** function "UberNode::add\_value" is inaccessible.

delete pRoot;

**Problem:** function "UberNode::~UberNode()" is inaccessible.

int count = pRoot->get\_count();

**Problem:** class "UberNode" has no member "get\_count".

pLast->append(pNew);

**Problem:** class "UberNode" has no member "append".

**Modify:**

class UberNode : public int\_list\_t

{

    public :

        UberNode() {}

        ~UberNode() {}

        void add\_value(std::string key, int value)

        {

        m\_map[key] = value;

        }

        int value(std::string key) { return m\_map[key]; }

    private:

        std::map<std::string, int> m\_map;

};

2.

double\_list\_t pDoubleRoot = new double\_list\_t;

double\_list\_t \*pDoubleLast = pDoubleRoot;

**Problem:** no suitable constructor exists to convert from "double\_list\_t \*" to "LinkedListNode<double, 75000>".

int count = pDoubleRoot->get\_count();

**Problem:** operator -> or ->\* applied to "double\_list\_t" instead of to a pointer type.

delete pDoubleRoot;

**Problem:** expression must be a pointer to a complete object type.

**Modify:**

double\_list\_t \* pDoubleRoot = new double\_list\_t();

    double\_list\_t \*pDoubleLast = pDoubleRoot;

3.

pDoubleNew->add\_value("COUNT", count);

**Problem:** class "LinkedListNode<double, 75000>" has no member "add\_value"

**Modify:**

Add add\_value function into LinkedListNode Class as UberNode Class did.

**Linked\_list.h**

**1.**

#endif // LINKED\_LIST\_H

**Problem:** expected a ';'

**Modify:**

};

#endif // LINKED\_LIST\_H

**Learned from the tutorial:**

1. Not coding is better than substandard coding.
2. "Code revies" can help programmers review the code. Reviews can be classified into three categories:
3. Code Walkthrough
4. Technical Review
5. Code Inspection
6. People are imperfect. Engineers are also human, so engineers are not perfect and make mistakes. Attackers will use errors in the code to turn them into vulnerabilities.
7. Dynamic testing is expensive, so before dynamic testing is executed, using static testing (which uses mechanical methods to statically verify the software) can reduce costs, and at the same time can detect and repair defects more effectively.
8. Ignoring false positives leads to a maintenance nightmare.
9. The better development practice steps are:
10. Careful needs analysis.
11. Reasonable design practice.
12. Effective dynamic testing.
13. Static analysis.
14. Code review
15. There are four static analysis models:
    1. Syntax and construct analysis.
    2. Class structure and inheritance analysis.
    3. State machine model analysis.
    4. Control and data flow graph analysis.