CSCI 3110 C++ class friendship

- **Function as a friend to a class**

  Example: overloaded << operator, overloaded >> operator

```
class A {
private:
          int data;
public:
               …
          friend ostream & operator <<(ostream & os, const A& rhs);
               …
};

ostream & operator <<(ostream & os, const A& rhs) {
      os << rhs.data;

      return os;
}
```

- **Class friendship**
  - class B is a friend of class A,
    - class B's methods have access to all data in class A
    - does not automatically make class A a friend of class B

```
class A {
public:
          …
          friend class B;
          void g (B* p);
private:
          int   data1;
};

class B {
public:
          void f (A* p);
          …
private :
          int   data2;
}

void B::f(A* p)   { // can access A's private data
          cout << p→data1 << endl;   // or *p.data1
          // does not support :   cout << A::data1 << endl;
}

// A is not a friend of B
void A::g(B* p) {
          cout << p→data2;   // error
}
```

One more example:

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

class CSquare; // forward declaration

class CRectangle {
private:
    int width, height;
  public:
    int area ()       {return (width * height);}
    void convert (CSquare a);
};

class CSquare {
  private:
    int side;
  public:
    void setSide (int a)   {side=a;}
    friend class CRectangle;   // class CRectangle is a friend of class CSquare
};

void CRectangle::convert (CSquare a) {
  width = a.side;// can access a's private data
  height = a.side;
}

int main () {

  CSquare sqr;
  CRectangle rect;

  sqr.setSide(4);

  rect.convert(sqr);
  cout << rect.area();

  return 0;
}
```

friendship is neither inherited nor transitive:

```
class A  {
        friend class B;

        int a;
};

class B  {
        friend  class C;

        string s;
};

// C is a friend of B, B is a friend of A, does
not make C a friend of A
class C {
        void f(A* p)      {
                p→a++;  // error
        }
};

// friendship cannot be inherited
class D: public B
{
        void f(A* p)      {
                p→a++; // error
        }
};
```