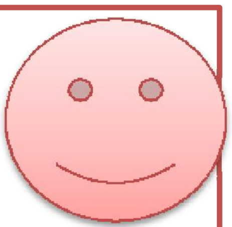
Friendship in C++



```
class  A {
private:
   secrete  myLove;
  public:
   A ():myLove(❤) {}
   void tell()
    {
      std::cout<<"I dont have BF ";
    }
   void tellBsSecrete(const B& b)
    {
      std::cout<<"B is in Love
with"<<b.myLove;
    }

};
```

```
class  B {
  friend class A;
private:
   secrete  myLove;
  public:
  B ():myLove(💙) {}
   void tell()
    {
       std::cout<<"I dont have GF ";
    }
   void tellAsSecrete(const A& a)
    {
     std::cout<<I don't  know\n";
     std::cout<<"I don't have access to her
secrets";//a.myLove is error
    }
};
```

```
int main () {
 A  a;
 a.tell () ;//Tells "I don't have BF"
 B b;
 b.tell () ;//Tells "I don't have GF"

 //Okay lets find out
 //Ask A about B's secrete
 a.tellBsSecrete(b) ; ///Here is secrete of B

 //Ask B about A's secrete
 b.tellAsSecrete(a) ;
 //But poor B is doesn't know any secret's about A.

 return 0;
}
```

In this example,
class A is a friend of class B

Here **B is** allowing A to access
private and protected secrets
about him.
More concretely,
A knows,with whom B is in Love with.

Friendships are never corresponded unless specified:
A is considered a friend class by B,
but B is not considered a friend by A.
Therefore, the member functions of A can access the protected and private members of B but not the other way around.

Friendship is not transitive:
The friend of a friend is not considered a friend unless explicitly specified.

Friendship is not inherited :
your friend's children are not your friends

reference :
http://www.cplusplus.com/doc/tutorial/inheritance/
https://en.cppreference.com/w/cpp/language/friend

(C) R-BNK
rohitbenake@gmail.com