#include <iostream>

class B;

class C;

class A {

private:

int privateA;

protected:

int protectedA;

friend void friendFunction(A objA, B objB, C objC);

};

class B {

private:

int privateB;

protected:

int protectedB;

friend void friendFunction(A objA, B objB, C objC);

};

class C {

private:

int privateC;

protected:

int protectedC;

friend void friendFunction(A objA, B objB, C objC);

};

void friendFunction(A objA, B objB, C objC) {

std::cout << "Accessing data from class A:" << std::endl;

std::cout << "Private A: " << objA.privateA << std::endl;

std::cout << "Protected A: " << objA.protectedA << std::endl;

std::cout << "\nAccessing data from class B:" << std::endl;

std::cout << "Private B: " << objB.privateB << std::endl;

std::cout << "Protected B: " << objB.protectedB << std::endl;

std::cout << "\nAccessing data from class C:" << std::endl;

std::cout << "Private C: " << objC.privateC << std::endl;

std::cout << "Protected C: " << objC.protectedC << std::endl;

}

int main() {

A objA;

B objB;

C objC;

friendFunction(objA, objB, objC);

return 0;

}