|  |  |
| --- | --- |
| Edith Cowan University School of Science |  |

Workshop 12

Overloading Operators

**Related Objectives:**

* Learn what a friend is
* Declare a friend function
* Examine the benefits of polymorphism and overloading
* Use a friend function to access data from two classes
* Learn the rules that apply to operator overloading
* Overload an arithmetic operator
* Overload operators to work with a class object and a primitive type

**Activity:**

In the below tasks, using separate .CPP and .H files create a program which makes use of the class declared.

1. Complete, compile and run the following incomplete source code. The input and output requirements are illustrated by the screenshot below. To complete this you are required to:
   1. Add the implementation of the constructor for SalesOffice as declared by its prototype
   2. Implement your own divide operator (operator/) overload function

class SalesOffice {

private:

string officeName;

double sales;

public:

SalesOffice(string, double);

double operator/(SalesOffice);

};

Output Example:



1. Extend your program (SalesOffice) developed above to include the following source code in your class. The output requirements are illustrated by the screenshot below.

ostream& operator<<(ostream& out, const SalesOffice &anOffice) {

out << "The " << anOffice.officeName << " Office sold $" << anOffice.sales << endl;

return out;

}

Output Example:

