Boone Tison

Problem Set #2

1.

a. Update can read C or its attributes, but not change them.

void update()

{

print(\_a);

int num = \_a + 10;

}

b. Update cannot change C or any of its attributes.

void update()

{

\_a++;

this = new C(15);

}

c. When in a method, A, that has a constant parameter ,A1, another method, B, cannot be called with A1 unless the method B's parameter is also constant. Add a keyword for instance methods that indicates the method does not alter the object the method is attached to. Instance methods can only be called on a constant reference if the instance method's declaration contains the keyword. Any method called within an instance method with the keyword, must also have the keyword as well.

2. Output:

9

81

2

9

B is executed 4 times.

3.

i. Require that the code uses both the unit and identifier.

* This is burdensome for the user and basically removes the point of the import. This could be done for all cases, or just when the identifier has a conflict.

The more recently declared import is given priority over the previous ones.

* This requires the compiler to follow the textual order of the imports. This takes the problem out of the language and into the compiler.

During run-time, the user is prompted to indicate which unit they want to use for the identifier.

* Puts the burden of the problem on the user. Will significantly slow down execution, especially in large programs. Requires the user to have complete knowledge of how the code works and the units.

ii. The locally declared identifier takes precedence over the imported identifier.

* This works because it follows the conventionial method for scoping, where it searches locally and then globally. Puts the burden of the problem on the developer to know the scoping of their program.

Require the user the rename the identifier in the local scope when it conflicts with an imported identifier.

* You can have the IDE identify this problem before the program is compiled, but it requires the coder to handle the error instead of the language or compiler.

For the first case, I would recommend that the language requires the use of both the unit and identifier, but only in cases where a conflict is detected with the identifier.

For the second case, I would recommend that the local identifier is given precedence over the imported identifier.