// Author: Kenry Yu

// Demo:

#include “Can.h”

#include <iostream>

Using namespace std;

Int main() {

Can c1(“Coke”, 12);

Can c2(“beans”, 20);

Can c3 = c2;

{

Can c4(“juice”, 32);

Can c5(“juice”, 12);

C4 = c5;

Cout << “\n--------Can #4->”;

C4.display();

Cout << “\n--------Can #5->”;

C5.display();

}

C3 = max(c1, c2);

Cout << “\n--------Max size->”;

C3.display();

// the following code does not compile – why?

// Because c4 and c5 were initialized in a block (scope?)

// but now we are outside the block

// so they have been destructed

// cout << “--------Can #4->”; c4.display(); cout << “--------Can #5->”;

// c5.display();

}

#include <iostream>

#include <string>

class Can {

private:

std::string content;

float ounces;

public:

Can() : content("Empty"), ounces(0){};

Can(const Can &);

Can(std::string con, float num) : content(con), ounces(num){};

~Can();

void operator=(Can);

void display();

friend Can max(Can, Can);

};

#include "Can.h"

Can::Can(const Can &right) {

this->content = right.content;

this->ounces = right.ounces;

}

Can::~Can(){}

void Can::operator=(Can right) {

this->content = right.content;

this->ounces = right.ounces;

}

void Can::display() {

std::cout << this->ounces << " ounce(s) of " << this->content << std::endl;

}

Can max(Can left, Can right) {

return (left.ounces >= right.ounces) ? left : right;

}

Text

Description automatically generated