```cpp
// Bernard J. Gole Cruz, CS 202-2002, Assignment 2 stage1
// This program is a simulation of coffee machine
// an implementation of class, objects from class, abstraction.

#include<iostream>
#include<string>
#include<iomanip>
using namespace std;

//global constant
const int DEFAULT_CAPACITY = 10;

//class declaration
class CoffeeMachine
{
public:
    //default constructor
    CoffeeMachine()
    {
        this->curr_water = 0;
        this->curr_coffee = 0;
        this->water_capacity = DEFAULT_CAPACITY;
        this->coffee_capacity = DEFAULT_CAPACITY;
        this->coffee_spoons_per_cup = 1;
        this-> name = "UNTITLED";
        cout << "created coffee machine " << name << " with empty resources." << endl;
    }
    //second constructor with coffee machine name
    CoffeeMachine(string name)
    {
        this->curr_water = 0;
        this->curr_coffee = 0;
        this->water_capacity = DEFAULT_CAPACITY;
        this->coffee_capacity = DEFAULT_CAPACITY;
        this->coffee_spoons_per_cup = 1;
        this-> name = "DECAF";
        cout << "created coffee machine " << name << " with empty resources." << endl;
    }
    //third constructor with coffee machine, water capacity, coffee capacity parameter
    CoffeeMachine(string name, int x, int y)
    {
        this->curr_water = 0;
        this->curr_coffee = 0;
        this->coffee_spoons_per_cup = 1;
        this-> name = "BLEND";
        cout << "Starting up Coffee Machine " << name << " with empty resources and capacities:" << endl;
        this->water_capacity = x;
        this->coffee_capacity = y;
        cout << right << setw(22) << "water_capacity=" << water_capacity << endl;
        cout << right << setw(23) << "coffee_capacity=" << coffee_capacity << endl;
        cout << endl;
    }
    //destructor
    ~CoffeeMachine()
    {
        cout << "shutting down Coffee Machine " << name << " with the following resources left:" << endl;
        cout << right << setw(7) << "water:" << curr_water << endl;
        cout << right << setw(8) << "coffee:" << curr_coffee << endl;
        cout << endl;
    }

    string name;
    int makeCups(int);
    void addWater(int);
    void addCoffee(int);
    void setCoffeeSpoonsPerCup(int);
```

```
67      void displayCM();
68
69  protected:
70      int coffee_spoons_per_cup;
71
72  private:
73      int water_capacity;
74      int coffee_capacity;
75      int curr_water;
76      int curr_coffee;
77      void makeSingleCup();
78
79  };
80
81  //Functions
82  //mutator function, set water
83  void CoffeeMachine::addWater(int w)
84  {
85          int water_overflow;
86          curr_water = w;
87
88          if (curr_water <= 0)
89          //nothing happen
90              {
91              return;
92              }
93
94          //fills water to full if cups more than default capacity
95          if (curr_water > water_capacity)
96              {
97              water_overflow = curr_water - water_capacity;
98              curr_water = curr_water - water_overflow;
99              //return;
100             }
101 };
102
103 //mutator function, set coffee
104 void CoffeeMachine::addCoffee(int c)
105 {
106         int coffee_overflow;
107         curr_coffee = c;
108
109         if (curr_coffee <= 0)
110             {
111         //nothing happen
112             return;
113             }
114
115         //fills coffee to full if cups more than the default capacity
116         if (curr_coffee > coffee_capacity)
117             {
118
119             coffee_overflow = curr_coffee - coffee_capacity;
120             curr_coffee = curr_coffee - coffee_overflow;
121             //return;
122             }
123 };
124
125
126 //check if resources are enough before making a cup
127 int CoffeeMachine::makeCups(int cups)
128 {
129         //will not make coffee if resources are not enough
130         if (cups > curr_water || cups > curr_coffee)
131             {
132             cout << right << setw(3) << "ordered " << cups << " cups of coffee of strength 1" << endl;
```

```
133             cout <<"NOT ENOUGH RESOURCES!" ;
134             //cout << endl;
135             }
136 
137         //will make coffee if resources are enough
138         else
139             {
140             cout <<"ordered " << cups << " cups of coffee of strength 1" << endl;
141                 int i = 0;
142                 while (i < cups )
143                     {
144                     makeSingleCup();
145                     i++;
146                     }
147 
148         //update the status of current water/coffee level in container
149         curr_water = curr_water - cups;
150         curr_coffee = curr_coffee - cups;
151 
152         cout << endl;
153         //display current state
154         displayCM();
155             }
156 };
157 
158 //make coffee per cup
159 void CoffeeMachine::makeSingleCup()
160 {
161   cout << "...made cup of coffee " << name << "..." << endl;
162 };
163 
164 //display current state
165 void CoffeeMachine::displayCM()
166 {
167     //updates status depending on the coffee machine created
168     cout <<"Current state of CM: " << name <<endl;
169     cout << right << setw(7) << "WATER:" << right << setw(3) << curr_water << right << setw(2) << "/" <<
right << setw(3) << water_capacity << right << setw(7) << "(cups)" << endl;
170     cout << right << setw(8) << "COFFEE:" << right << setw(2) << curr_coffee << right << setw(2) << "/" <<
right << setw(3) << coffee_capacity << right << setw(9) << "(spoons)" << endl;
171     cout << right << setw(10) << "STRENGTH:" << right << setw(2) << coffee_spoons_per_cup << right << setw(
22) << "coffee spoons per cup" << endl;
172     cout << endl;
173 };
174 
175 //coffee spoon per cup
176 void CoffeeMachine::setCoffeeSpoonsPerCup(int cspc)
177 {
178      coffee_spoons_per_cup = cspc;
179 };
180 
181 
182 
183 int main()
184 {
185 
186 
187     CoffeeMachine cm1; //activate UNTITLED coffee machine
188       //cm1 objects
189     cm1.addWater(8); //add water
190     cm1.addCoffee(8);//add coffee
191     cm1.displayCM(); //display current state
192     cm1.makeCups(5); //make a cup
193     cout << endl;
194     cout << endl;
195 
```

```
196      CoffeeMachine cm2("DECAF"); //activate DECAF coffee machine
197        //cm2 objects
198      cm2.addWater(10); //add water
199      cm2.addCoffee(10);//add coffee
200      cm2.displayCM();  //display current state
201      cm2.makeCups(14); //make a cup
202      cout << endl;
203      cout << endl;
204
205      CoffeeMachine cm3("BLEND",15,20); // activate BLEND coffee machine
206       //cm3 objects
207      cm3.addWater(14); //add water
208      cm3.addCoffee(20);//add coffee
209      cm3.displayCM();  //display current state
210      cm3.makeCups(12); //make a cup
211      cm3.makeCups(5);  //make a cup
212      cout << right << setw(7) << "ABORT." << endl;
213
214
215      system("pause");
216      return 0;
217  }
```