#include <string>

using namespace std;

#ifndef IIKH\_H

#define IIKH\_H

class IIKH { //identifies itself as the IIKH

private:

string graphicalImageOfRecipeBox; //file name of graphical image of recipe box

bool buttonPressed; //before begining, default value is false.

int option; //index which user elected

void displayGraphicalImageOfRecipeBox(string graphicalImageOfRecipeBox); //displays a graphical image of recipe box

void displayReturnbutton(); //displays a return button

bool getButtonPressed(); //get button is pressed or not to start begining when user presses the return button

void setButtonPressed(bool buttonPressed); //set button is pressed

void displayOption();//when user presses the return button to begin

void receiveOptionPicked(); //get input from user what option index to display

void setOption(int option); //set what option did user picked

void showRecipeBook(); //if option recipe index is selected

void quit(); //if option quit is selected by user from a program menu, and the application quits.

public:

IIKH();

int getOption(); //get what option did user picked

};

class RecipeBook {

private:

Recipe recipeBook[1000]; //Array of Recipes

string keywords[1000]; //get keywords in line, and then sepearate by ETX.

Recipe searchedRecipes[1000]; //Array of Searched Recipes.

//searching

string getKeywords(); //get keywords in line from user that will be the basement of the search

void splitKeywords(); //split keywords in line by ETX.

void search(string keywords); //do searching

void setSearchedRecipes(Recipe newRecipe); //set recipes by searching

void displayResult(); //display the search result with keywords

void displayRecipe(Recipe recipe); //get input from user what recipe to display, and display the recipe which user pressed

public:

RecipeBook();

};

class Recipe {

private:

string name;

string flavoring;

string pictureOfFinishedDish; //file name of picture of finished dish

string ingredients[1000]; //the list of ingredients

string preparationSteps[1000];

int expectedPreparationTime;

void setName(string name);

void setFlavoring(string flavoring);

void setPictureOfFinishedDish(string pictureOfFinishedDish);

void setIngredients(string ingredients[1000]);

void setPreparationSteps(string preparationSteps[1000]);

void setExpectedPreparationTime(int expectedPreparationTime);

void displayRecipe();

void printRecipe(); //requests a printing of the recipe, and the output is spooled to printer.

public:

Recipe(string name, string flavoring, string pictureOfFinishedDish, string ingredients[1000], string preparationSteps[1000], int expectedPreparationTime);

string getName();

string getFlavoring();

string getPictureOfFinishedDish();

string getIngredients();

string getPreparationSteps();

int getExpectedPreparationTime();

};

#endif