**The text below is a the class specification (i.e. header files) of IIKH system.**

#pragma once

#include <iostream>

#include <string>

using namespace std;

// This structure contains the ingredients and amounts needed in the meal.

struct ingred\_list {

string content; // content of the ingredient

float num; // amount of ingredient

ingred\_list\* next;

};

// This structure contains the contents of the recipe and the meal.

struct arr\_list {

string content; // content of the corresponding object

arr\_list\* next;

};

// This structure is a database that contains ingredients and contents corresponding to recipes.

struct recipe\_data {

string recipe\_name; // recipe's name

ingred\_list ingredients; // list of ingredient

arr\_list recipe; // step needed to transform the ingredients into the final product

recipe\_data\* next;

};

// This structure is a database that contains the period and meals of the plan.

struct plan\_data {

string plan\_name; // plan's name

string period; // plan's period

arr\_list meals; // the corresponding plan of meal

plan\_data\* next;

};

// This class is the welcome window of IIKH. There is navigation menu which has several options like plans, recipes.

class greeter {

public :

int choice; // button

// This method browses window and displays several menu.

void browse\_window(int choice);

};

// This class is to show and browse different recipes and select them.

class meal {

public :

// This method interacts with the recipe database to select a recipe for meals by displaying information about the meal.

recipe\_data select\_recipe(recipe\_data\* recipe\_data);

// This method modifys the amount of recipe according to the number of people eating the meal provided by the parameter.

recipe\_data set\_number(recipe\_data selected\_recipe, int number);

// This method provides grocery list for a meal by scaled recipes.

ingred\_list get\_grocery\_list(recipe\_data scaled\_recipe);

};

// This class is to show and browse tasks regarded as a specific plan.

class date {

public :

// This method finds the data in the plan database in the name of the existing plan you want to change and edits it according to the period and meal you want to modify.

void edit\_meal(string plan\_name, string period, arr\_list meals);

// display grocery list.

void print\_grocery\_list(ingred\_list grocery\_list);

// This method finds the data in the database with the name of the plan you want to modify and annotates it with a new name.

void edit\_plan\_name(string dest\_name, string new\_name);

};

// This class is to show and browse create meal plans and to add them delete them or remove them.

class plans {

public :

int plan\_choice; // button

// This method displays several menu regarded as plan in windows.

void print\_plans\_menu(int plan\_choice);

// This method shows different meal plans. Various period meals can be shown here.

string select\_plan(void);

// This method adds data to the plan database according to the name, period, and meal.

void add\_plan(string name, string period, arr\_list meals);

// This method removes a specfic meal plan in the plan database. If there is no data in the plan database, an error message is printed.

void remove\_plan(string name);

};

// This class has a browse option to check different recipes, adding recipes, editing and removing them.

class recipes {

public :

int recipes\_choice; // button

// This method shows several menu regarded as recipe in windows.

void print\_recipes\_menu(int recipes\_choice);

// This method adds a new recipe data to the recipe database according to the name, ingredients, and method.

void add\_recipe(string name, ingred\_list ingredients, arr\_list method);

// This method interacts with the recipe database to provides a recipe for the name of meal.

void get\_recipe(string name);

// This method finds the data in the recipe database in the name of the existing plan you want to change and edits it according to the ingredients and recipe you want to modify.

void edit\_recipe(string dest\_name, string new\_name, ingred\_list ingredients, arr\_list recipe);

// This method removes a specfic recipe in the recipe database. If there is no data in the plan database, an error message is printed.

void remove\_recipe(string name);

};

// This class is database with all recipe informations.

class recipeDatabase {

recipe\_data\* recipe\_data;

};

// This class is database with all plan informations.

class planDatabase {

plan\_data\* plan\_data;

};