

Project Description File

Purpose of the program (Revised)

The overall purpose of this program is to inform individuals with the Menstruation related topic. These individuals could be either a male or female, who is looking to gain more insight into this topic.

Furthermore, the users have access to a calendar tool, where they enter their or their relative's menstrual cycle, and the tool will generate an estimated date for the next Period. This way, they can plan their vacation or activities so they're able to share a memorable moment together. In addition to that, every user who access this program gets entered into a log file along with time, date and their names.

Finally, the program reads from a tab-space delimited text file containing most searched queries online about Menstruation. Therefore, my program is simulating a browser-like search functionality into a smaller environment; this program uses an Edit Distance algorithm that fetches the data from the text file when the user types an input on the console. In addition to this, there are also some string correction functions to give them a proper result.

Does program fit any of EC criteria? (Revised)

I believe that this program will fit EC1. The reason is, even though this program is small, both men and women may find this useful which can assist them with their quick inquiries. Furthermore, the calendar tool can be used particularly by active couples/individuals who like to plan things ahead of time. Overall, I think this program is practical enough to be used by a good portion of our population.

UML Diagram (Updated)

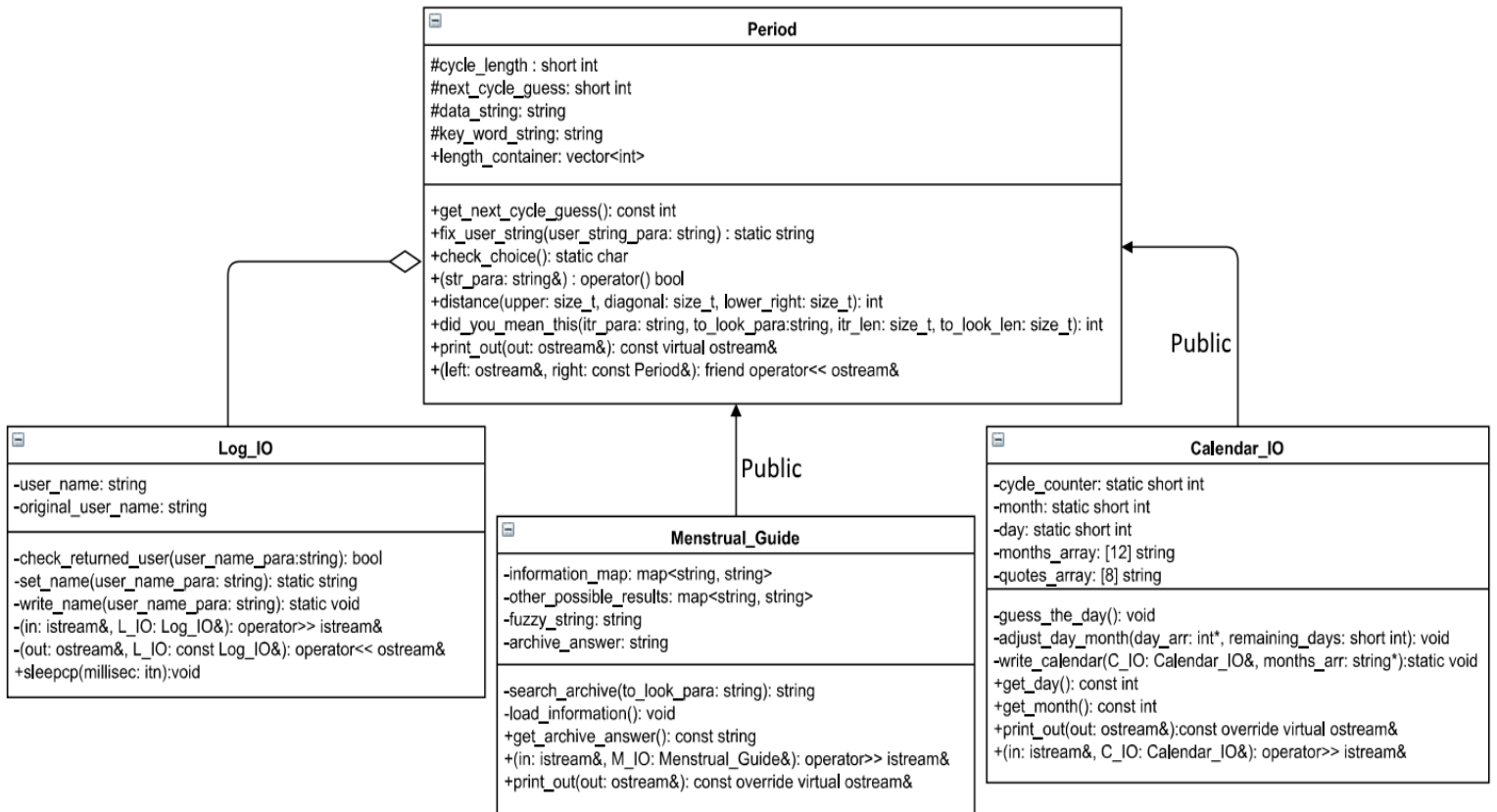
Please see page 2 of this document.

UML Description (Updated)

Please see page 3 & 4 & 5 & 6 of this document.

Specific OOP implementation

Please see the last page of this document.



Base-class: Period

I. Data members (No Change)

- cycle_length of type short int is used for holding user's aggregate days of past period intervals.
- next_cycle_guess of type short int is used for holding the value which gets a return after calculations are done.
- data_string of string type holds lines which would get read into a map in the information_map of made of string parameters.
- key_word_string of string type holds keywords from the text which would get read into the map in the information_map made of string parameters.
- length_container of type int in STL vector is used for holding int cycle_length at each index.

II. Member functions (Updated)

- get_next_cycle_guess() const of int type, is used in the ostream to show the final calculated result in the CalenderIO subclass.
- fix_user_string() of static string type, with the parameter that accepts a string type, removes the numbers and punctuation marks from the user's string.
- check_choice() of static char type takes the input for yes or no, and return the char. Contains a try-catch block for exception handling.
- operator() of bool type is an overloaded function object that accepts a string by reference and gives out a warning if a non-letter string is found.
- distance() of int type is a function that accepts three parameters and returns the int value which is a minimum of three. It is a recursive-helper function for did_you_mean_this().
- did_you_mean_this() is a string search function which accepts a user string, a key string of map, and their two lengths, and since it is type int, it returns the number of edits made in these strings to make them equal.
- print_out() function of ostream type is set to pure virtual because each class has a different definition of printing its output on the console. Making it pure, makes the class abstract which hides the important data members and data functions from the user.
- ostream operator << is set to be a friend of the pure virtual function above. This way every class can override that virtual function to print their output. Note, this is not built in friend ostream, since generic built-in friend operators doesn't allow pure virtual override.

NOTE: Local variables within some functions may have been used to make copies of the data members from classes.

Subclass: Menstural Guide ---> Period

I Data members (Updated)

- information_map of type map containing string parameters is an STL container which holds string key_word_string in left parameter and data_string in right parameter when gets read from text file.
- other_possible_results of type map contain strings parameters. This STL container holds the results which weren't caught by primary conditional statement.
- fuzzy_string of type string holds the user string. It is used with the search algorithm.
- archive_answer of type string is used to copy the value from the map. Then using this to output in an appropriate place.

II Member functions (Updated)

- get_archive_answer() of const string type is a getter function that return the archive_answer.
- search_archive(), is a function of type string; it also accepts a string. When the user enters input using istream of this class, that string is passed into this function. This function nests the did_you_mean_this() function and manipulates the user string to give out a proper result. It has the functionality to remove spaces from user string, and the key_string. Using iterator, we find a close match.
- load_information() function is a void function which will read the text file data and store them into the information_map in left and right parameter.
- The extraction operator>> accepts parameters istream and Menstural_Guide is a function which will take the input within this class and store into the fuzzy_string.
- The insertion operator<< is another definition of the print_out() function; It has overridden that pure virtual function. It prints out program headers and tips.

NOTE: Local variables within some functions may have been used to make copies of the data members from classes.

Subclass: Calender IO ---> Period

I Data members (Updated)

- cycle_counter of type static short int will get incremented every time cycle_length is entered in the length_container vector.
- month of type short int will take the month value from the insertion operator.
- day of type short int will take the date value from the insertion operator.
- months_array of size 12, of string type, contains 12 months of the year.
- quotes_array of size 8, of string type, contains quotes which will be used randomly and will be printed to the calendar.

II Member functions (Updated)

- get_day() of type int is a getter that returns the value of the day.
- get_month() of type int is a getter function that returns the value of the month.
- guess_the_day() function of void type calculates the result and stores that result in the next_day_guess variable in Base class.
- adjust_day_month() is type void and accepts a monts_array and a short in value, which is an offset of days coming in from istream of this class. This function uses that offset to predict the next Period. It shifts the days, and month accordingly to reflect that.
- write_calender() function is a static void type that accepts the CalenderIO object and an array containing quotes. It makes a CSV file in the folder, shows a reminder marker symbol "X" in there to show when the next Period is expected.
- The extraction operator>> here will take the data for CalenderIO data members and store them in their fitting variables. The istream operator of this function controls the flow of this program.
- Insertion operator ostream<< gets another definition by overriding the base class pure virtual function. This accepts parameters of ostream and CalendarIO, both by reference, to show an appropriate message on the console. Here it displays the next date and average interval for the menstrual cycle.

NOTE: Local variables within some functions may have been used to make copies of the data members from classes.

Composition-class: Log IO ---◇Period

I Data members (New)

- user_name is of type string, stores the corrected version of user name.
- original_string_name is of string type, stores the original user_name. It is used to show "Hello..." "Bye..."

II Member functions (New)

- check_returned_user(), returns bool type, accepts string. It reads the file, and checks if string passed matches the names in file.
- set_name(), returns string, accepts string parameter, and returns the fixed version of the user_name. This function removes the spaces, converts the string into lower case.
- sleepcp() is type void, accepts a parameter of type int. It gives the proper Sleep() function for the console version. Sleep() is amount of lag in console. It has definition of #ifdef and #else, this way overhead will be managed automatically.
- The extraction operator>> here takes the input for user_name. It also has a colored loading bar that uses the sleepcp() function.
- Insertion operator ostream<< shows the farewell message.

NOTE: Local variables within some functions may have been used to make copies of the data members from classes

OOP Implementations (Please use the Peek Definition option in VS to see the implementation in .cpp files)

<i>OOP Concept</i>	<i>Class</i>	<i>Line Numbers</i>
Encapsulation	<i>Period.h</i>	60,65
	<i>Menstrual_Guide.h</i>	21,30
	<i>Calendar_IO.h</i>	17,40
	<i>Log_IO.h</i>	21,31
Inheritance	<i>Period.h</i>	N/A (Parent Class)
	<i>Menstrual_Guide.h</i>	19
	<i>Calendar_IO.h</i>	17
	<i>Log_IO.h</i>	N/A (Composition used)
Polymorphism	<i>Period.h</i>	78
	<i>Menstrual_Guide.h</i>	36
	<i>Calendar_IO.h</i>	47
	<i>Log_IO.h</i>	N/A
Static data members and functions	<i>Period.h</i>	71,72
	<i>Menstrual_Guide.h</i>	N/A
	<i>Calendar_IO.h</i>	20,39
	<i>Log_IO.h</i>	26,27
Friend functions	<i>Period.h</i>	79-81(defined)
	<i>Menstrual_Guide.h</i>	35(pre-defined)
	<i>Calendar_IO.h</i>	48 (pre-defined)
	<i>Log_IO.h</i>	29(pre-defined),30(pre-defined)
Overload operators	<i>Period.h</i>	73(defined)
	<i>Menstrual_Guide.h</i>	35(pre-defined)
	<i>Calendar_IO.h</i>	48(pre-defined)
	<i>Log_IO.h</i>	29(pre-defined), 30(pre-defined)