Class: CSCE1040 - Section 001

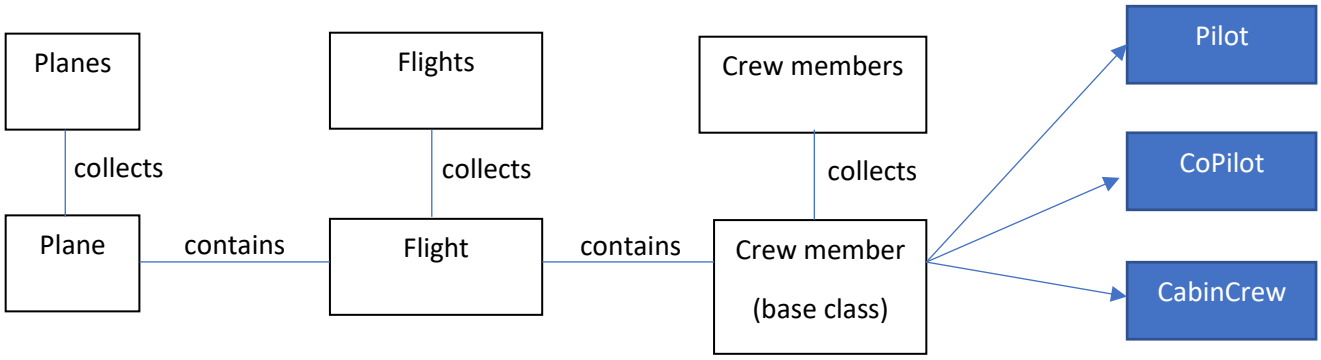
Name: Yen Pham

Instructor: David Keathly

Assignment: Homework 5

Design document for Mean Green Airlines

I. **Class Relationships**



II. **Class Contents**

Planes/ Aircrafts

Number of planes (int)

List of planes (vector)

Set/get number of planes

Add a plane

Edit a plane

Search a plane

Print a plane

Print plane list

Print an assignment schedule for a plane

Load plane list

Store plane list

Get plane vector

Add a time slot to a plane's time slot vector

Delete a time slot from its time slot vector

Sort time slot of a plane

Crew Members

Number of members (int)

List of members (vector)

Set/ get number of members

Add a member

Edit a member

Search a member

Print a member

Print member list

Print a schedule for an active crew member

Load crew member list

Store crew member list

Add a time slot to a member's time slot vector

Delete a time slot from its time slot vector

Sort time slot of a member

Flights

Number of flight (int)

List of flights (vector)

Set/get number of flights

Add a flight

Edit a flight

Update past flights' status to completed

Search a flight

Delete a flight

Print a flight

Print flight list

Print flight schedule for a plane

Print flight schedule for a crew member

Delete all cancelled/ completed flights

Load a list

Store a list

Plane/ Aircraft

Make (string)

Model (int)

Plane ID (string)

Number of Seats (int)

Range (int)

Status (string)

Time slot (vector of time slots – time slots are pairs of start time and end time)

Set/get make

Set/get model

Set/get tail number

Set/get number of seats

Set/get range

Set/get status

Add time slot to time slot vector

Delete time slot from vector

Get time slot vector

Sort time slot of a vector

Crew Member

Name (string)

ID number (int)

Type (string)

Status (string)

Time slot (vector of time slots – time slots are pairs of start time and end time)

Set/get name

Set/get id number

Set/get type

Set/get status

Add time slot to time slot vector

Delete time slot from vector

Get time slot vector

Sort time slot of a vector

Virtual PrintInfo

Flight

Plane ID (string)

Pilot ID (int)

CoPilot ID (int)

Crew IDs (int)

Start Date/Time (time_t)

End Date/Time (time_t)

Starting Airport code (char)

Ending Airport Code (char)

Number of Passengers (int)

Status (string)

Set/ get plane ID

Set/ get pilot ID

Set/ get coPilot ID

Set/ get Crew ID

Set/ get start Date/Time

Set/ get end Date/Time

Set/ get starting Airport code

Set/ get ending Airport Code

Set/ get number of passengers

Set/ get status

Pilot

Rating code (char*)

Flight hours (time_t)

Set/get rating code

Set/get flight hours

Print pilot's info

CoPilot

Rating code (char*)

Flight hours (time_t)

Set/get rating code

Set/get flight hours

Print co-pilot's info

CabinCrew

Position (string)

Set/get position

Print cabin crew's info

III. Functions/ Pseudocodes

Class Plane

Plane Constructor

Set all string value to “none” and integer value to 0

Class Planes

Add plane

Load all the data in the file “airplane”

Prompt user to enter airplane’s make

Prompt user for airplane’s model

Prompt user for air plane’s ID

Prompt user for number of seats

Prompt user for range in miles

Prompt user for airplane’s status. Map the plane status with the input time.

Create plane Object

Populate Object

Add object to plane collection

Store all the data in the file “airplane”

Edit plane

Load data in file “airplanes”

Prompt user to enter the plane’s ID

Run a loop of the plane’s collection from the beginning to end:

- If the plane’s ID does not match with the user’s input, display a message. If not, continue
- Display the menu of options that user can change for the plane. Prompt user to choose what to edit (make, model, seats number, range, status)
- Prompt user to enter the new variable they want to edit
- Call the set function to reset the variable

Store data in file “airplanes”

Search plane

Load data in file “planes”

Display a menu of options user wants to search for a plane

If user’s choice matches the option, prompt user to enter the value they want to find

Run a loop of the plane’s objects in collection from the beginning to end. If the user’s input equals to the one in the collection, display the object’s values.

Store data in file “planes”

Delete plane

Load data in file “planes”

Prompt user to enter plane’s ID

Run a loop of plane’s collection. If user’s input matches any plane’s ID in the collection, delete that object in the vector.

Store data in file “airplanes”

Print plane

Prompt user to enter plane’s ID they want to print

Run a loop of a plane’s collection. If user’s input matches with one in the plane’s collection, print that object’s values using mutators

Print plane collection

Run a loop of a plane’s collection from the beginning to end. Print each object’s values using mutators

Sort time slot of a plane

Access time slot of a plane, then run through the time slot vector using loop. If start time of one time slot is larger than the other, swap it ascendingly until all time slots are set in ascending order

Class Crew member

Crew member constructor

Set all string variable to “none” and integer variable to 0

Class Crew members

Add crew member

Load data in file “members”

Create member Object pointer

Create pilot, co-pilot, cabin crew pointers

Prompt user for member’s type

Link member pointer to the pointer of the correct member’s type

Prompt user to enter member’s name

Prompt user for member’s ID

Prompt user for member’s status

Prompt user for rating code

Populate Object

Add object to crew collection

Store data in file “members”

Edit crew member

Load data in file “members”

Prompt user to enter the crew member’s ID

Run a loop of the crew member’s collection from the beginning to end:

- If the crew member’s ID does not match with the user’s input, display a message. If not, continue
- Identify the type of the crew member. Display the menu of options that user can change for the crew member of that type. Prompt user to choose what to edit (type, status)
- Prompt user to enter the new variable they want to edit
- Call the set function to reset the variable

Store data in file “members”

Search crew member

Display a menu of options user wants to search for a crew member

If user's choice matches the option, prompt user to enter the value they want to find

Run a loop of the crew member's objects in collection from the beginning to end. If the user's input equals to the one in the collection, display the object's values.

Delete crew member

Load data in file "members"

Prompt user to enter crew member's ID

Run a loop of crew member's collection. If user's input matches any crew member's ID in the collection, delete that object in the vector.

Store data in file "members"

Print crew member

Prompt user to enter crew member's ID they want to print

Run a loop of a crew member's collection. If user's input matches with one in the crew member's collection, print that object's values using mutators

Print crew member collection

Run a loop of a crew member's collection from the beginning to end. Print each object's values using mutators

Sort time slot of a member

Access time slot of a member, then run through the time slot vector using loop. If start time of one time slot is larger than the other, swap it ascendingly until all time slots are set in ascending order

Class flight

Flight constructor

Set all string variable to "none" and integer to 0

Set start time to current time

Set end time to current time

Class flights

Add flight

Load data in file “flights”

Prompt user for crew member ID

Prompt user for pilot ID

Prompt user for coPilot ID

Prompt user for crew IDs

Prompt user for start date/time

Prompt user for end date/time

Prompt user for starting airport code

Prompt user for ending airport code

Prompt user for number of passengers

Prompt user for status

Run a loop through the crew member collection from the begin to end:

- If the user’s input of crew member ID matches any crew member ID in the crew member collection, return the status of the crew member that matches with the start time and end time. If the status of the crew member is inactive, return. Else, continue with the next step
- If the number of seats available of that crew member equals to 0, return
- Run through a loop of crew member, if user’s crew member ID input matches one in the crew collection, check if the crew member’s status at that time matches with the user’s input time. If the crew member is “available”, create flight object
- If flight is pushed back to flight vector, add time slot of the flight to plane’s time slot vector and member’s time slot vector

Populate Object

Add object to flight collection

Store data in file “flights”

Edit flight

Load data in file “flights”

Prompt user to enter the flight’s ID

Run a loop of the flight's collection from the beginning to end:

- If the flight's ID does not match with the user's input, display a message. If not, continue
- Display the menu of options that user can change for the flight. Prompt user to choose what to edit
- Prompt user to enter the new variable they want to edit
- Call the set function to reset the variable
- If user want to change start time or end time, besides reset the start/end time from the flight, we must run a loop of plane's list and member list and change the start/ end time slot of that plane/ member

Store data in file "flights"

Update flights based on current time

Set up current time

Run a loop of flight list. If flight's end time is before the current time, change the flight status to completed

Search flight

Display a menu of options user wants to search for a flight

If user's choice matches the option, prompt user to enter the value they want to find

Run a loop of the flight's objects in collection from the beginning to end. If the user's input equals to the one in the collection, display the object's values.

Delete flight

Load data in file "flights"

Prompt user to enter flight's ID

Run a loop of plane collection. If plane's ID matches with the plane ID in the flight, run a loop of time slot. If flight's start time and flight's end time matches with one time slot, delete the time slot of that plane ID

Run a loop of member collection. If member's ID matches with the member ID in the flight, run a loop of time slot. If flight's start time and flight's end time matches with one time slot, delete the time slot of that member ID

Run a loop of flight's collection. If user's input matches any flight's ID in the collection, delete that object in the vector.

Store data in file "flights"

Delete all cancelled flights

Run a loop of flight's collection. If the flight's status is cancelled or completed, delete that flight

Print flight

Prompt user to enter flight's ID they want to print

Run a loop of a flight's collection. If user's input matches with one in the flight's collection, print that object's values using mutators

Print flight collection

Run a loop of a flight's collection from the beginning to end. Print each object's values using mutators

Print an assignment schedule for an member

Prompt user to enter member's ID

Run a loop through member's collection.

If user's entered member ID matches with one in the flight list, run a loop of member list.

If that member ID matches one in the member list, sort time slots of that member ID.

Print airmember's time slot and flight details for each time slot.

Print a schedule for a crew member

Prompt user to print member ID

Run a loop of a flight list.

If user's entered member ID matches one in the flight list, run a loop of member list. If that member ID matches one in the member list, sort time slots of that member ID.

Print member's time slot and flight details for each time slot