**Lab2ResultYourname.doc.** Use the file to save your answers to questions and/or work accomplished below each corresponding numbered task.

**Lab2 Topic: A class, its member functions, and its data members.**

**Concept/Rule 1**: All members of a class are private by default.

**Concept/Rule 2**: A variable/object of a class declared outside of the class

definition can access only public members of the class.

**Concept/Rule 3**: A class private member can only be accessed by the class

public members.

**Concept/Rule 4:** A class can have more than one constructor.

A constructor is a special member function that has no return

type and its name is the same as the class name.

A constructor should be used to initialize the data members of

the class.

**Concept/Rule 5:** The default constructor does not have any parameter.

Define the default constructor to initialize data members of a

class.

**Concept/Rule 6:** The default constructor is called when an object/variable of the

class is created in the following syntax:

ClassName objectName;

**Your Tasks:**

1. **Compile** the following program, lab2Program.cpp, and run it to see the output.

**Note:** The program is exactly the same as lab1Program.cpp but the keyword **class** is used instead of the keyword **struct**. The program should display error messages after complied.Read and copy/paste the error messages.

#include <iostream>

#include <string>

using namespace std;

**class** EmailAddressBook

{

string name; //data member

string emailAddress; // data member

EmailAddressBook( )

{ cout << "The default constructor is called " << endl;

name = "Blank ";

emailAddress = "Blank ";

}

EmailAddressBook(string n, string a )

{ cout <<"In the overloaded constructor. "<< endl;

name = n;

emailAddress = a;

}

void Display( )

{

cout << "Name = " << name << "Email address = " << emailAddress << endl;

}

};

int main( )

{

EmailAddressBook myAssociate1; // LINE 1

myAssociate1.Display(); // LINE 2

cout << " In the main ,Name1 = " << myAssociate1.name<< "Email address 1= "<< myAssociate1.emailAddress << endl; // LINE 3

EmailAddressBook myAssociate2("Jack Smith", "jsmith@msn.com"); // LINE 4

myAssociate2.Display(); // LINE 5

cout << " In the main ,Name 2 = " << myAssociate2.name<< "Email address 2= "<< myAssociate2.emailAddress << endl; // LINE 6

EmailAddressBook myAssociateArray[3]; // LINE 7

for (int i = 0 ; i < 3 ; i++) // LINE 8

{ cout << "Enter your associate name and his/her email address # " << i + 1 << endl; // LINE 9

cin >> myAssociateArray[i].name >>myAssociateArray[i].emailAddress ; // LINE 10

}// LINE 11

cin.get( ); // Wait for any character to be entered.

}

ERROR STATEMENT:

Error 1 error C2248: 'EmailAddressBook::EmailAddressBook' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 30 Lab2

Error 2 error C2248: 'EmailAddressBook::Display' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 31 Lab2

Error 3 error C2248: 'EmailAddressBook::name' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 32 Lab2

Error 4 error C2248: 'EmailAddressBook::emailAddress' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 32 Lab2

Error 5 error C2248: 'EmailAddressBook::EmailAddressBook' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 33 Lab2

Error 6 error C2248: 'EmailAddressBook::Display' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 34 Lab2

Error 7 error C2248: 'EmailAddressBook::name' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 35 Lab2

Error 8 error C2248: 'EmailAddressBook::emailAddress' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 35 Lab2

Error 9 error C2248: 'EmailAddressBook::EmailAddressBook' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 36 Lab2

Error 10 error C2248: 'EmailAddressBook::name' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 39 Lab2

Error 11 error C2248: 'EmailAddressBook::emailAddress' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 39 Lab2

1. Add the statement **public: above the statement string name;** As a result, all members become public members:

**class** EmailAddressBook

{ public:

string name; //data member

…

};

Compile it again and run the program. It should work. **What Concept/Rule has been proven?**

*That class by DEFAULT is private.*

1. Next, make all class functions to be public members and variables to be private members as follows:

**class** EmailAddressBook

{ private:

string name; //data member

string name; //data member

public:

string emailAddress; // data member

EmailAddressBook( )

{ cout << "The default constructor is called " << endl;

name = "Blank ";

emailAddress = "Blank ";

}

…

};

Compile it again and run the programto see error messages. Read each error message and double click on each message. An arrow should point to where the error occurs. Copy/paste the error messages.

Error 1 error C2086: 'std::string EmailAddressBook::name' : redefinition f:\c++ class 162\the lab work\lab2\lab2program.cpp 10 Lab2

Error 2 error C2248: 'EmailAddressBook::name' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 36 Lab2

Error 3 error C2248: 'EmailAddressBook::name' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 39 Lab2

Error 4 error C2248: 'EmailAddressBook::name' : cannot access private member declared in class 'EmailAddressBook' f:\c++ class 162\the lab work\lab2\lab2program.cpp 43 Lab2

**What Concept/Rule has been proven?**

*It proves that items must be marked properly for proper compiling. Data being requested was marked private.*

1. Comment out LINE 3 and Line 4 in the main( ) function. Compile it again and run the programto see if the program works. **What Concept/Rule has been proven**

*That you must be careful with what you call private, or lack o f access will cause compiling error*

1. Add a for loop in the main( ) to display three sets of values in myAssociateArray. Save your program as **Lab2ProgramYourname.cpp.** Capture a screenshot of the program output.
2. **Analyze the codes and assess your understandings:** Identify line(s) of code that prove each Concept/Rule.

**Concept/Rule 1**: All members of a class are private by default.

Proof: Changing

**class** EmailAddressBook

{ private:

string name; //data member

string name; //data member

To:

**class** EmailAddressBook

{ public:

string name; //data member

…

};

This won’t compile unless it is declared public, since it is private by default

**Concept/Rule 2**: A variable/object of a class declared outside of the class

definition can access only public members of the class.

EmailAddressBook myAssociate1; // LINE 1

myAssociate1.Display(); // LINE 2

cout << " In the main ,Name1 = " << myAssociate1.name<< "Email address 1= "<< myAssociate1.emailAddress << endl; // LINE 3

This causes errors.

**Concept/Rule 3**: A class private member can only be accessed by the class

public members.

EmailAddressBook myAssociate1; // LINE 1

myAssociate1.Display(); // LINE 2

cout << " In the main ,Name1 = " << myAssociate1.name<< "Email address 1= "<< myAssociate1.emailAddress << endl; // LINE 3

This causes errors.

This does not:

**class** EmailAddressBook

{ public:

string name; //data member

…

};

**Concept/Rule 4:** A class can have more than one constructor.

A constructor is a special member function that has no return

type and its name is the same as the class name.

A constructor should be used to initialize the data members of

the class.

class EmailAddressBook

{

public:

string name;

string emailAddress;

EmailAddressBook( )

{ cout << "The default constructor is called " << endl;

name = "Blank ";

emailAddress = "Blank ";

}

EmailAddressBook(string n, string a )

{ cout <<"In the overloaded constructor. "<< endl;

name = n;

emailAddress = a;

}

**Concept/Rule 5:** The default constructor does not have any parameter.

Define the default constructor to initialize data members of a

class.

class EmailAddressBook

{

public:

string name;

string emailAddress;

EmailAddressBook( )

{ cout << "The default constructor is called " << endl;

name = "Blank ";

emailAddress = "Blank ";

}

EmailAddressBook(string n, string a )

{ cout <<"In the overloaded constructor. "<< endl;

name = n;

emailAddress = a;

}

**Concept/Rule 6:** The default constructor is called when an object/variable of the

class is created in the following syntax:

ClassName objectName;

EmailAddressBook myAssociate1; // LINE 1

**What To Turn In**

Submit:

Lab2ResultYourname.doc and Lab2ProgramYourname.cpp

Grading Rubric:

|  |  |
| --- | --- |
| Lab 2 Completion of Tasks #1-#4 | 8 points |
| Lab2ProgramYourname.cpp which contains the modified code done in Task #5. | 6 points |
| Lab2 Completion of Task #6 | 6 points |