

---

## Programming Assignment 6: Contacts Class Objects List

**CGS 3406 - Fall Term 2009**

**Point Value: 100 points**

**Project Due : Tuesday December 1, 2009 at 11:59 PM**

**Reminder: this assignment is due at this specified time deadline and NO late work will be accepted!!!**

---

### Learning Objectives

- To write a program which requires the creation of a class data structure
- To write constructors and other member functions for the class, including accessor and mutator (get and set) functions
- To write an application program for the class to demonstrate its use with an array of class objects
- To use the typedef construct to set up a data type for an array of class objects

### Problem Statement

Create a *Contact* class that can hold information and perform operations with contact objects. The basic idea here is that on a mobile phone, you typically have a contacts list or address book. The *Contact* class must have the following private data members:

Member	Description
name	string (use C++ string class for all strings here)
address	string
age	int
phone	string
type	enum type called <i>ContactType</i> , one of NONE, FRIEND, RELATIVE or WORK (in that order)

The class must have the following public member functions:

--	--

Member Function	Description
default constructor	sets all strings to "unavailable", age to zero and type to NONE
alternate constructor	accepts values for all private data members as arguments and sets the appropriate member values to those arguments (here the type is passed as an enum)
void setName	accepts a string argument that is copied to the name member variable
void setAddress	accepts a string argument that is copied to the address member variable
void setAge	accepts an integer argument that is copied to the age member variable
void setPhone	accepts a string argument that is copied to the phone member variable
void setType	accepts a string argument that is used to set the enum type member variable
string getName	returns the value in name
string getAddress	returns the value in address
int getAge	returns the value in age
string getPhone	returns the value in phone
ContactType getType	returns the value in type
void displayContact	prints the information for one contact (prints a string for the enum type value)
void displayType	prints out a string corresponding to the ContactType

Write this program in a *single* C++ program file.

Demonstrate the class by writing an application for it in main and other functions. Note that your application should include functions other than main, which are not part of the class, but are needed to modularize your application.

Your application program must do the following:

1. Declare an array of class objects which can store up to a maximum of 5 contacts. You must provide for sub-array processing on this array.
2. Print out values for all the data members of all 5 contact objects in the array, at this initial point, which is right after the array declaration.

3. Open the data file provided to you on the class web site, named *contacts.txt*. Then read in the first line containing the number of contacts for which data is provided, and next the actual contact data. Each item of contact data is given on one line in the file. If the file does not open when your program tries to open it, abort the run as shown in lectures, along with printing an error message for your user.
4. After you have read in the provided file data, print out the number of actual contacts provided in the file and the data for each contact. At this time you will only print contact data for the part of the array that has meaningful contact information (which may be fewer than 5).

## Input and Other Requirements

The only input is from the data file. You may assume the file data is completely correct and you do not have to do any error checking on file contents. You may assume that the number of contacts specified in the file is an integer between 1 and 5.

## Output

- As usual: introduction, echoprinted input, closing termination message, error messages as needed, and any informative messages the user may need or want to see
- Elements described in this write-up
- Follow the course style guidelines

Be creative!

## Miscellaneous

You must use the C++ string class for all string data in this program.

You must set up the class exactly as specified in this write-up, with class members exactly as described. You may not therefore add or delete data members or function members from the class. Members must behave exactly as described in this write-up.

You should have non-member functions other than main in your program, as your program needs to be modular and use functions and parameter passing appropriately. For example, a function in your main program called "PrintHeading" which printed out introductory program output, would be a non-member function.

## Regarding FSU Computer Competency Requirement

This assignment is assessed according to the FSU Computer Competency Requirement. As such students must demonstrate (1) competent use of a discipline-specific software package: Microsoft Visual Studio C++ and (2) the ability to perform simple transactions using the web/Internet: obtaining all of the assignment materials from the class web site

and downloading the required file(s), and turning in the completed work correctly to the class web site.

## **What File To Turn In and How to Turn In Your Work using Blackboard**

Turn in your C++ program source file, which must be named *contacts.cpp*

Be 100% certain you have *thoroughly* read the handout entitled "*Submitting Your Program Assignments Electronically Using Blackboard*" for complete instructions on how to submit and how to **verify your submission** after you submit it. This will help you to avoid losing large numbers of project points due to improper submission issues. This handout is available on the course web site under "Handouts."

---

Last Update: October 28, 2009 3:15 PM, A. Ford Tyson

---