

CS 210 - Fundamentals of Programming I

Spring 2006 - In-class exercise for 4/04/06

10 points

Names: _____

This exercise should be completed in pairs. The purpose of this in-class exercise is to work with C++ strings. It uses the standard string class defined in `<string>`. The following is a problem statement, and an analysis and design for the program in file `inclass22.cpp`.

Problem Statement

Write a program that simulates what the phone company must do to interpret numbers entered as a phone number. It should check for the validity of the number dialed (i.e., it's 7 or 11 digits long, local numbers do not start with 0 or 1, long distance numbers must start with 1, etc.) and then prints out the number in a standard U.S. format showing the area code (if long distance), the exchange, and the local number. For example, the input 5556789 should display as 555-6789 and the input 12345556789 should display as (234)-555-6789.

Main Program

- Analysis

Objects	Type	Kind	Name
A phone number	string	variable	phoneNumber
A formatted phone number	string	variable	formattedNumber
User response to continue	char	variable	response

- Design

1. Greet the user
2. Do
 - 2.1 Read in phoneNumber
 - 2.2 If IsValid(phoneNumber) then
 - 2.2.1 Compute formattedNumber = FormatPhoneNumber(phoneNumber)
 - 2.2.2 Display formattedNumber
 - 2.3 Ask the user if they want to continue
 - 2.4 Read in response
3. While response is 'y' or 'Y'

Function: IsValid

- Analysis

Objects	Type	Kind	Movement	Name
a string to check	string	variable	received	aString
result of check	bool	variable	returned	—

- Design

1. If `aString.length()` is not 7 or 11 then
 - 1.1 Display error message, “A phone number has 7 or 11 characters”
 - 1.2 Return false
2. If `aString.length()` is 7 and `aString[0]` is ‘0’ or ‘1’ then
 - 2.1 Display error message, “A local call cannot begin with 0 or 1”
 - 2.2 Return false
3. If `aString.length()` is 11 and `aString[0]` is not ‘1’ then
 - 3.1 Display error message, “A long distance call must begin with 1”
 - 3.2 Return false
4. If `aString.find_first_not_of (“0123456789”, 0)` is not `string::npos` then
 - 4.1 Display error message, “A phone number must consist of all digits”
 - 4.2 Return false
5. Return true

Function: FormatPhoneNumber

- Analysis

Objects	Type	Kind	Movement	Name
a phone number	string	variable	received	phoneNumber
a formatted phone number	string	variable	returned	—

- Design

1. If `phoneNumber[0]` is ‘1’ then // use long distance format: (xxx)-xxx-xxxx
 - 1.1 Return “(” + `phoneNumber.substr(1,3)` + “-” + `phoneNumber.substr(4, 3)` + “.” + `phoneNumber.substr(7,4)`
2. Else // use local format: xxx-xxxx
 - 2.1 Return `phoneNumber.substr(0,3)` + “-” + `phoneNumber.substr(3, 4)`

0. Create a new project, then download file `inclass22.cpp` from the course webpage under today’s date and add it to your project.

1. (4 points) The functions `IsValid` and `FormatPhoneNumber` have not been implemented in the file `inclass22.cpp`. Using the analyses and designs given above, write the function prototypes and definitions of these two functions for this program.

2. (6 points) Build and run this program. Test it until you are satisfied that you understand how it works. Part of adequate testing is to show an input for each possible result. Give an input that will cause the program to:

- a. Display a valid long distance phone number

- b. Display a valid local phone number

- c. Display the error message “A phone number has 7 or 11 characters”

- d. Display the error message “A local call cannot begin with 0 or 1”

- e. Display the error message “A long distance call must begin with 1”

- f. Display the error message “A phone number must consist of all digits”

When you have completed this exercise and are confident the program gives the correct results, **demonstrate to the instructor that the input you have indicated above gives the results indicated.** Print out your program file and turn it in with one copy of this exercise sheet with your tests above.