CMPSC 121: Project 5

Jung-woo Sohn (jwsohn@ist.psu.edu)

June 10, 2015

1 Instructions

In this project, you are basically combining the two existing `welcome.cpp` and `navigate.cpp` files into one working program using **functions**. You will start from using the simplest type of function, with no input parameters and no return values (returns `void`).

Write a program with two source code files (`welcome.cpp` and `navigate.cpp`) from your previous projects and implement the following features. To summarize, the program will (i) display a welcome message, (ii) print out the menu for destinations and get a user input, and (iii) print out the investigation result.

1. Conversion of `main()` function in `navigate.cpp` into `navigate()` function.

   - Create a new project from your IDE. Copy `navigate.cpp` and `welcome.cpp` files from your previous project into the new project folder. Add the two files into your project.
   - In C++, recall that `main()` function is the starting point of the program execution.
   - We will keep `main()` in `welcome.cpp` and convert the other `main()` into `navigate()` function in `navigate.cpp` file.
   - In `navigate.cpp`, change `int main()` into `void navigate()`. (NOTE: `navigate()` does not have any input parameters nor return any value.)
   - Delete `return 0;` in `navigate()` function. (return type of `void` means that there will be no return values.)

2. Calling `navigate()` function from `main()` in `welcome.cpp`

   - After displaying welcome message and having detective name input, we want to let the detective (i) select the destination and (ii) make an investigation on the criminal's track.
   - At the end of `main()` function, call `navigate()` function by simply putting `navigate();` statement.
   - Specification of **function prototype** is needed when you put the details of the function in a separate file when you call a function. In this example, the details of `navigate()` is not in `welcome.cpp`, but in `navigate.cpp` file.
   - Therefore, you need to specify the function prototype for `navigate()` in `welcome.cpp` so that the compiler can know that the details of `navigate()` can be found in some other file than `welcome.cpp`.
   - Put the function prototype of `void navigate();` above `int main()`. With this, the code will compile.

3. Calling `investigate()` function from `main()` in `welcome.cpp`

- Similarly, put `investigate();` statement after `navigate();` statement so that the program will call `investigate()` function after calling `navigate()`.
- For now, the `investigate()` function will simply print out “Carmen Sandiego was here!” message, instead of providing any criminal tracking infomration. (We will cover this in later projects.) Therefore, the source code for `investigate()` function can simply be as below:

```
void investigate()
{
    cout << "Carmen Sandiego was here!" << endl;
}
```

- Add this function declartion at the end of `navigate.cpp`.
- Since the details of the function is in a different file, you need to specify its function prototype in `welcome.cpp` so that the compiler can search other source code files when it comes across the calling statement of `investigate();` in `welcome.cpp`.
- Put the function prototype of `void investigate();` above `int main()`. With this, the code will compile.

1.1 Sample output

The sample output from the working program will be as follows:

```
$ ./game
***** Welcome to WHERE IN THE WORLD IS CARMEN SANDIEGO *****

Your name, please: Foobar

Welcome, detective Foobar

You have 7 days and 12 hours to arrest a criminal.
Please select your next destination:

1. New York
2. London
3. Cairo
4. Rio de Janeiro

Please enter your selection: 0
You entered a wrong number.
Please enter your selection: 10
You entered a wrong number.
Please enter your selection: 2

Your input is : 2
Your next destination is : London
Carmen Sandiego was here!
```

1.2 Code templates

Check out the following code templates for your information.

Listing 1: `welcome.cpp`

```cpp
#include <iostream>
#include <string>

using namespace std;

/* function prototypes go here */

int main()
{
    string name = "";
    int totalHours = 0, hoursLeft = 0, daysLeft = 0;

    /* print welcome screen. Get user name input from keyboard */

    cout << "***** Welcome to WHERE IN THE WORLD IS CARMEN SANDIEGO *****" << endl;
    cout << endl;

    cout << "Your name, please: ";
    cin >> name;

    cout << endl;
    cout << "Welcome, detective " << name << endl;
    cout << endl;

    totalHours = 7 * 24 + 12;            // 7 and 1/2 days remaining
    daysLeft = totalHours / 24;
    hoursLeft = totalHours % 24;

    cout << "You have " << daysLeft << " days";
    cout << " and " << hoursLeft << " hours to arrest a criminal." << endl;

    /* select a destination */

    // call navigate() function here

    /* investigate on Carmen's destination */

    // call investigate() function here

    return 0;
}
```

Listing 2: navigate.cpp

```
#include <iostream>
using namespace std;

/* put proper function header definition for navigate() here. */
{
    int menu = 0;
    string destination = "";

    /* print destination lists and get user input */

    cout << "Please select your next destination:" << endl;
    cout << endl;

    cout << "1. New York" << endl;
    cout << "2. London" << endl;
    cout << "3. Cairo" << endl;
    cout << "4. Rio de Janeiro" << endl;
    cout << endl;

    cout << "Please enter your selection: ";
    cin >> menu;

    while (menu < 1 || menu > 4)
    {
        cout << "You entered a wrong number." << endl;
        cout << "Please enter your selection: ";
        cin >> menu;
    }

    if (menu == 1)
        destination = "New York";
    else if (menu == 2)
        destination = "London";
    else if (menu == 3)
        destination = "Cairo";
    else if (menu == 4)
        destination = "Rio de Janeiro";
    else
        cout << "ERROR! The choice has to be between 1 and 4" << endl;

    cout << endl;
    cout << "Your input is : " << menu << endl;
    cout << "Your next destination is : " << destination << endl;

}

/* investigate() function goes here. */
```