COMP2710 – Frequently Asked Questions
Homework 5

1. When the program starts, should there be an option for the user to not input a question?

   Answer: The user has to input a question. Your program will perform as a test driver.

2. If the caller asks for 5 questions but the list only contains 3 questions then we should print a warning. Should this warning end the program?

   Answer: No, your program should not be terminated upon this warning message. Rather, your program simply displays the warning message.

3. When asking the user the trivia questions, do you want the questions to start with the first question input by the user or the last question input by the user?

   Answer: Either way is fine.

4. Does the order the questions are asked in matter?

   Answer: You can insert a question either to the head or to the tail of the list.

5. Can you explain what the purpose of the last test case is? The one which seems to output "Warning – there are only three trivia in the list."

   Answer: Let me give you an example to explain this test case. Suppose a user would like to answer five trivia questions. Unfortunately, your trivia list contains only three questions. You program should let the user know that the trivia bank has insufficient number of questions.

6. I assumed that you just comment out #define UNIT_TESTING when you're wanting to just run your product version and you do not comment that line out when you just want to run the unit testing?

   Answer: You are right! When you comment out #define UNIT_TESTING, all the code related to unit testing will be excluded from the executable code.

7. My function is not detecting when the input is equal to the desired answer. I have implemented this in a while loop to ask all of the questions; in the while loop, an if-else statement determines whether the input answer and the desired answer are equal or not. The boolean expression that I am using is: if (answerIn == cur->answer) (where answerIn is the answer input from the getline() function, cur is the pointer to the

current question in the linked list, and answer is the desired answer to the current question). I printed both the input answer and the desired answer to make sure that they are both correct, and they are exactly the same. What can I do to fix this? Is there a different way that I should be checking string equality?

Answer: Suppose you want to compare `cur->answer` with `player_answer`, which is a string entered from the keyboard using getline(). You simply do the following:

```
string player_answer;
getline(cin, player_answer);
if (player_answer == cur->answer) {
    cout << "Put your message here!" << endl;
    score += cur->points;
}
```

8. I'm having a problem with the function that asks the user to input answers for questions. Whenever the answer is a single number or word, the function runs perfectly; however, when the answer is two or more words (ex: "Bank of Italy"), only "Bank" is saved in the string. Then, when it goes on to the next question, it automatically uses "or" as the answer.

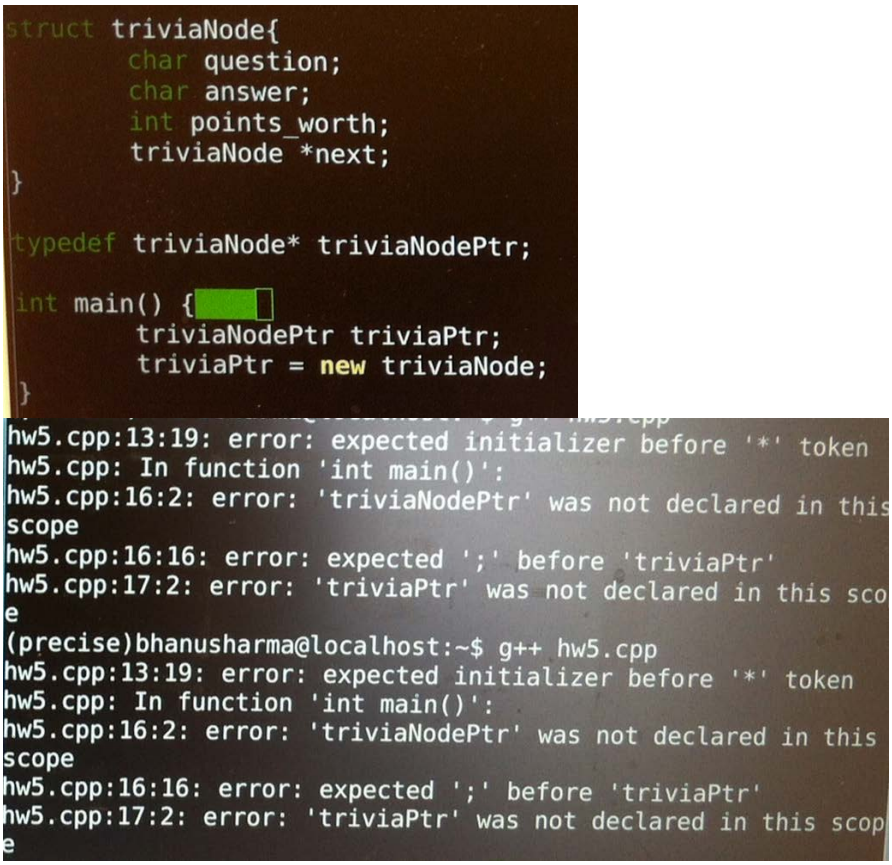
   It seems like my program is separating "Bank of Italy" into three different strings; how can I input it so that it is stored in one string?

   Answer: Please review the following sample code for the answer.

```
// This program shows how to get strings and non-string data from
keyboards.

#include <climits>
#include <iostream>
#include <string>

using namespace std;
//#define BUG1
//#define BUG2
#define WORKING

int main( )
{
    string question;
    string answer;
    unsigned int score;
    unsigned int i;

#ifdef BUG1
    cout << "Enter a question:";
    cin >> question;
```

```
    cout << "Enter an answer:";
    cin >> answer;

    cout << "Enter award points:";
    cin >> score;

    cout << "Question: " << question << endl;
    cout << "Answer: " << answer << endl;
    cout << "Score: " << score << endl;
#endif

#ifdef BUG2
  for (i = 0; i < 3; i++) {
    cout << "Enter a question:";
    getline(cin, question);

    cout << "Enter an answer:";
    getline(cin, answer);

    cout << "Enter award points:";
    cin >> score;
    cout << "Question: " << question << endl;
    cout << "Answer: " << answer << endl;
    cout << "Score: " << score << endl;
  }
#endif

#ifdef WORKING
  for (i = 0; i < 3; i++) {
    cout << "Enter a question:";
    getline(cin, question);

    cout << "Enter an answer:";
    getline(cin, answer);

    cout << "Enter award points:";
    cin >> score;
    cin.clear();
    cin.ignore(INT_MAX, '\n');
    cout << "Question: " << question << endl;
    cout << "Answer: " << answer << endl;
    cout << "Score: " << score << endl;
  }
#endif
    return 0;
}
```

9. I understand how to do everything but my program still does not compile because it keeps stating that triviaNodePtr does not exist, and if it does, the compiler states that triviaPtr is not a type. What am I doing wrong?



```
struct triviaNode{
        char question;
        char answer;
        int points_worth;
        triviaNode *next;
}

typedef triviaNode* triviaNodePtr;

int main() {
        triviaNodePtr triviaPtr;
        triviaPtr = new triviaNode;
}
```

```
hw5.cpp:13:19: error: expected initializer before '*' token
hw5.cpp: In function 'int main()':
hw5.cpp:16:2: error: 'triviaNodePtr' was not declared in this
scope
hw5.cpp:16:16: error: expected ';' before 'triviaPtr'
hw5.cpp:17:2: error: 'triviaPtr' was not declared in this sco
e
(precise)bhanusharma@localhost:~$ g++ hw5.cpp
hw5.cpp:13:19: error: expected initializer before '*' token
hw5.cpp: In function 'int main()':
hw5.cpp:16:2: error: 'triviaNodePtr' was not declared in this
scope
hw5.cpp:16:16: error: expected ';' before 'triviaPtr'
hw5.cpp:17:2: error: 'triviaPtr' was not declared in this scop
e
```

Answer: Please remember to place ; at the end of the structure.

```
Struct trivaNode{
      ……;
      ……;
};
```

10. On your handout you have the parameter of the function as:
    `void insert_node(node_ptr_t& root, node_ptr_t new_node_ptr);`

    Why is the `new_node_ptr` not call-by-reference?

    Answer: We don't use call by reference, because the new_node_ptr is not modified by the insert_node() function.

11. I was wondering what exactly we have to test in regards to homework 5. I see on the description the only place where it tells you test is the function that asks the questions (Reading from step 6) It also seems that the single function from step 5 is the only function tested in the sample output. So do we only need to test the function created from step 6 as you did in the description? or should we test the other functions as well?

Answer: All important functions in homework 5 must be tested. For example, if you implement the void insert_node(node_ptr_t& root, node_ptr_t new_node_ptr) function, you need to write a test driver for it.

12. I am a little confused about what we're supposed to do on Step 3. I thought the user was suppose to create their own questions.

    Answer: You should hard code three questions in addition to a few user created questions.

13. The given algorithm for comparing strings only returns true if every character is the exact same in both case and letter, should we consider using a case insensitive string comparison instead?

    Answer: No. we only consider the "exact matching" case.

14. In the test driver for askQuestions() in homework 5, tester (user) input is required. So my questions is this, should the testers input be validated? For example, if said tester fails to follow the guide lines of the test case (i.e. 1 answered correctly and the tester answers incorrectly or more than 1) the program will terminate do to an assertion error. In a situation such as this should the program be allowed to terminate (doesn't seem like the best option)? Or should the program display a message to the tester and give the option to restart or continue with the testing?

    Answer: You don't use `assert()` to implement the test driver for `askQuestions()`. In your described case, your program simply displays an error message.