

# CSCE 221 Cover Page

## Assignment #1

Due January 31 by 5 pm to CSNet

First Name  
User Name

Last Name  
E-mail address

UIN

Please list all sources in the table below including web pages which you used to solve or implement the current homework. If you fail to cite sources you can get a lower number of points or even zero, read more at Aggie Honor System Office: <http://aggiehonor.tamu.edu/>

Type of sources				
People				
Web pages (provide URL)				
Printed material				
Other Sources				

I certify that I have listed all the sources that I used to develop the solutions/codes to the submitted work.

*On my honor as an Aggie, I have neither given nor received any unauthorized help on this academic work.*

Your Name

Date

# Assignment 1 (100 pts)

*Due January 31, 2013 by 5 pm to CSNet  
Hard copy of the report is due coming Monday/Tuesday in your lab*

The purpose of this programming assignment is to learn about organizing data using the STL classes `vector` and `list`, analyzing the best and worst scenarios for the data operations, and mastering C++ programming skills, following Dr. Stroustrup's instructions provided in "*Programming Style Guide*", see the link <http://www.stroustrup.com/Programming/PPP-style.pdf>.

## PROBLEM DESCRIPTION

In this programming challenge, you will create a simple game called "SmartyPants C++ Trivia Game". Your program should accomplish these tasks:

- Read at least 15 multiple choice questions from a text file. The questions should be related to C++. The number of possible choices should vary from two to five. When a question is displayed on the screen, a possible list of choices is also displayed. Only one answer is correct and if the player selects the correct answer, he or she gets a point.
- A player should provide answers to 10 questions displayed on the screen and get a feedback immediately about whether or not his or her answer is correct. The questions are selected randomly from the pool of available questions.
- At the end, the program displays the number of points earned and the player passes the test when the score is 70% or above, otherwise he or she fails.

## PROGRAM DESIGN

1. Write a class `Question` to hold the data for the trivia game.
  - (a) Declare the data members private.
  - (b) Use constructors to create objects of this class.
2. Outside the class definitions
  - (a) Overload the operator `==(const Question& q1, const Question& q2)` to return `true` if two questions and their lists of choices are the same.
  - (b) Overload the input and output operators.
3. Use STL containers: `vector` and linked list to store all the questions. Give a player an option to run one of the two container representations.
4. Create an input file with at least 15 good C++ questions with a list of choices. Each question and its list of choices is separated by an empty line from another question and its list of choices. The input file is a part of the assignment and it will be graded for correctness and technical creativity. Provide hard and software copies of your input file for grading.
5. Use exceptions to inform the player about an invalid selection for an answer.
6. You need to work with this code to eliminate run-time errors and/or logical errors with the help of a debugger. You need to fix all the bugs, and describe them in your report.
7. Provide results of your tests for both the implementations (`vector` and linked list).
8. **Important:** Test your program on a CS UNIX machine before returning your assignment for grading. Include all the testing cases in your report.

## **REPORT AND PROGRAM DOCUMENTATION**

- Follow the general steps to prepare a hard and soft copy of the report, see report instructions
- Write which C++11 features you have used in your implementation of the game.
- Apply programming principles and styles, see the link <http://www.stroustrup.com/Programming/PPP-style.pdf>

## **SUBMISSION**

- Electronic submission of your code, report and input file to CSNet
- Hard copy of your report and input file to a TA on the first lab after the deadline.
- The Question class should be declared in the "Question.h" file.
- Write a Makefile for running your programs, and include that in your submission. Refer to the end of the document for a Makefile example: [https://wiki.cse.tamu.edu/index.php/Makefiles\\_C\\_C%2B%2B](https://wiki.cse.tamu.edu/index.php/Makefiles_C_C%2B%2B)
- Tar all your files and name it by userID-Lastname-Firstname-A1.tar before submission Refer to for how to tar: <http://courses.cse.tamu.edu/teresa/csce221/html/tar-file.html>