

Due: Electronic copy of .cpp file on Blackboard for lab no later than Tuesday March/29/2016 at 11:59 p.m.
Late submissions will not be accepted under any circumstances.

Important:

All projects are to be completed individually. Do NOT collaborate on this project.

Do NOT discuss the project with anyone other than the instructor and Teaching Assistants.

All projects are to be completed individually; duplicate programs will not receive credit.

Your code will be executed on the systems in Dragas. Make sure that your code compiles and runs on the systems in Dragas before you submit your .cpp file. Review the grading algorithm before you submit your project!

Wow! Imaginary sales of our imaginary game have been skyrocketing! We need to put out the next version, and we have a deadline to meet. You have been tasked with updating and improving our game program that helps beginner programmers study and understand C++.

The program operates as before – see the output of the gameInfo function (option 3), and see the comments in that function for a general description of the game. You can call the function gameInfo(), in the provided project 3 code template.

You have to expand on the algorithm provided in the template, to solve this project. You are required to demonstrate the use of an algorithm in completing this assignment – meaning that you must include the number and short description of each step in the program documentation (comments)... showing your stepwise approach to solving this problem.

The program uses the same input files (questions1.txt ect) that were used in the previous two projects. Changes to this version include a user menu, and the option to play a two-player game. There is also a game history file created and updated every time a player finishes a game. An example of this file is provided for you to examine.

Run the provided executable to see sample output of the program.

```
//=====
///BY DEFAULT - THE GAME PRESENTS ONLY THREE QUESTIONS TO EACH PLAYER
/// IN A TWO PLAYER GAME.,
```

Notes:

1. Remember to include programmer documentation in the source code and in the output.
2. Graders will follow the directions in your prompts. You should test your program with different data sets. Create your own question input files to test your solution.
3. Use a separate prompt for each user input value.
4. Format floating point values in the output with **3** decimal places.
5. Remember to save a copy of the file you hand in. Leave the file unchanged after your submission until you receive a grade on the project assignment. Note that if you submit the wrong file as your solution, this will not earn you credit. You must submit your source code solution in the form of a file with the .cpp extension. Once the solution for a project has been released (usually coinciding with the due date), then no credit can be given or late submissions accepted. It is your responsibility to start early, and submit your solution in plenty of time to account for errors or questions for the lab TA on your submission.
6. Name your source code file using your last name and first initial as follows: lastname_firstInitial_Prj3.cpp ;for example, Meg Griffin would save her source code in a file name **Griffin_M_Prj3.cpp**
7. Remember that you can work in the Problem Solving Lab in Dragas 1103G or the lab located in room 3104 in the E & C S building. Hours for the labs are posted on the CS home page. You can also use the Remote Desk-Top Connection.
8. **Do not discuss the project with anyone other than the instructor and Teaching Assistants. Do not collaborate on this project. Projects are to be completed individually;** duplicate programs will not receive credit, and will be reported as a violation of the ODU Honor Code.

Submission details: Your TA will review the process of submitting your project in class. You will hand in an electronic copy of your source code. (.cpp file) We will run your program so it is important that you hand in a copy of your source file (**Griffin_M_Prj3.cpp**). **The file you hand in must have a .cpp extension.** You need to start this project early – do NOT wait until the last day to complete and submit your solution. Consult with your TA BEFORE submitting, if you have any questions. **Your code will be executed on the systems in Dragas. *Your code must compile and run on the systems in Dragas.***

Helpful Tips

- No late submissions – under any circumstances.
- Start Now. Do not procrastinate.
- Examine the sample code from our textbook, and run the programs in your IDE (codeblocks).
- Go to the recitations to get help from the TAs.
- Go see the TAs during their office hours.
- The algorithm steps you write will prove useful in creating and implementing your own solution.