

Holly Harmon  
OSU ID: 932456676  
Winter 2013  
CS 165  
Jess

1. lines 41-79 of main program
2. lines 41-79 of main program
3. N/A
4. Example: line 41 of main program
5. Example: cin - line 46 of main program  
    getline - line 63 of harmonh.cpp
6. Example: lines 27-35 of main program
7. Example: line 47 of main program
8. Example: line 145 of harmonh.cpp
9. Example: for - lines 91-95 of main program  
    do/while - lines 115-158 of main program  
    while - lines 47-49 of main program
10. line 97 in main program and line 457 in harmonh.cpp
11. There are many instances in the program where if I did not understand them or had errors in those categories it would be very clear. They are generally the easiest thing to check for when checking for errors.
12. Throughout.  
Examples: line 47 - cin.fail() to catch user input to an integer that is not an integer.  
Line 129 - While statement with == to make sure user is entering a valid character.  
Line 264 in harmonh.cpp - Uses toupper to change user input char to uppercase to make error checking easier.  
Line 586 in harmonh.cpp - In the function "open\_items" I checked all instances of .open for .fail and added error messages to make sure files were opening correctly. (Same for "add\_print\_items" on line 677.)  
I also had many temporary print statements throughout to make sure arrays, iostreams, etc were working correctly, but it looks like I took most of those out as I was editing.
13. Example: line 27 - "void user\_name()"
14. Demonstrated by the program. Also, most of the program was written out in sections, then pieced down to functions and put together in one main program.
15. The string variable 'first' on line 31 in harmonh.cpp is usable only by that function.
16. The function answer\_check on line 92 of harmonh.cpp uses both pass by reference (for the int variable userAns) and pass by value (the int variable answer).  
The function fill\_map\_array also passes a dynamically declared array.
17. The function answer\_check can accept either two int values or two string values and will act differently according to what parameters are passed to it.

18. Example: Line 28 of main program.
19. N/A Just couldn't think of a way to work one in.
20. The map array called 'main' declared at line 117 in the main program.
21. The map array is also a dynamically declared array.
22. Function called at line 38 (user\_name). The program will accept one or two strings entered into the command line as the user's name. Checks if one has been entered and lets the user know.
23. The struct starship can be found at line 11 of the header file.
24. You can find the class Creature at line 65 of the header file and Human and Alien below that one.
25. Lines 325-329 in harmonh.cpp declare and create a pointer to an array.
26. Couldn't quite figure this one out in time. Will definitely go back and figure it out once finals are over.
27. Lines 365-369 in harmonh.cpp create some pointers to objects.
28. The namespace 'Harmonh' can be found in the header file.
29. Header file is called harmonh.h
30. Makefile is attached with the other files.
31. Line 596 in harmonh.cpp is getting input from a text file and writing it to a vector.
32. The constructors can be found in the classes Creature, Human, and Alien.
33. N/A
34. The functions open\_items and add\_print\_items utilize and create .txt files.
35. Line 638 in harmonh.cpp uses a boost algorithm to convert all of the characters in the string buyItem to lowercase. I also included something from limits (to continue by pressing a key), but I'm not sure if that is also what this was looking for.
36. The classes Human and Alien are derived from the base class Creature.
37. N/A
38. The function user\_name at line 35 in harmonh.cpp has a throw and catch. Not really needed in this program, but this shows that I understand how to use it.
39. I made a game! I hope you have fun with it!
40. Used a stringstream to convert an integer to a string (Main program - line 104).

### **Cheat Sheet!**

Answers to the binary questions:

42, 10100011, 01100, 10100

Where are the asteroids??

I could have moved the asteroids around, but decided to keep them as static points in the array to make it easier for testing and easier for the person grading.

They are at points 4, 7, 9, and 12.