

C++ PROGRAMMING

(335)

REGIONAL – 2014

TOTAL POINTS _____ (*300 points*)

Failure to adhere to any of the following rules will result in disqualification:

- 1. Contestant must hand in this test booklet and all printouts. Failure to do so will result in disqualification.**
- 2. No equipment, supplies, or materials other than those specified for this event are allowed in the testing area. No previous BPA tests and/or sample tests or facsimile (handwritten, photocopied, or keyed) are allowed in the testing area.**
- 3. Electronic devices will be monitored according to ACT standards.**

No more than 90 minutes testing time

Property of Business Professionals of America.
May be reproduced only for use in the Business Professionals of America
Workplace Skills Assessment Program competition.

Note to Contestant:

The rubric for this event grants points for each minor task completed. That being the case, the contestant should strive to complete as many of the rubric items as possible. The contestant should have received a flash drive from the proctor. **A text file, “gameinventory.txt”, must be stored on the flash when you receive it.**

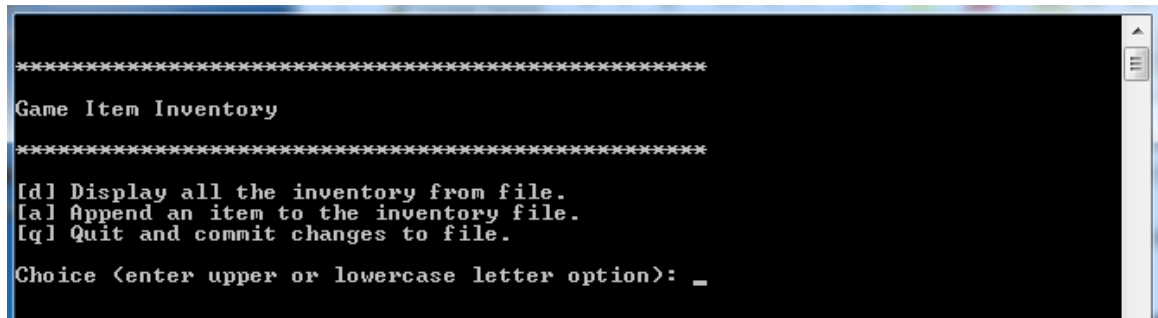
Create a folder named with your contestant ID number (ex. xx-xxxx-xxxx).

Make certain all of the files associated with your program are contained in the folder and that the program will compile/run from the folder. Failure to do so will result in the loss of points under the “Application/Execution” rubric section.

Game Item Inventory

You have been hired by BitBlit Interactive to create an application that will manage an item inventory for their text based role playing game.

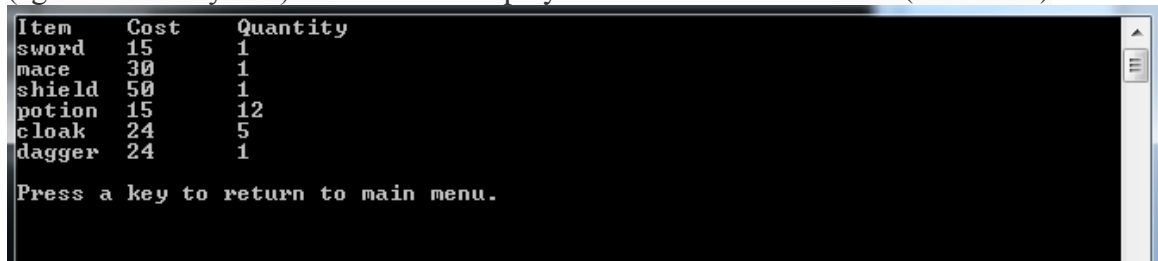
The program must provide the user with a menu (see below) that allows the user to do the following:



```
*****  
Game Item Inventory  
*****  
[d] Display all the inventory from file.  
[a] Append an item to the inventory file.  
[q] Quit and commit changes to file.  
Choice <enter upper or lowercase letter option>: _
```

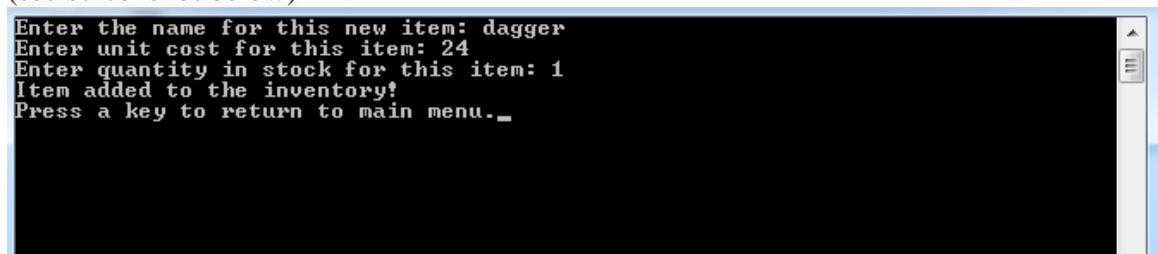
The program **must** include two methods:

- **void display_inventory()** – This method will read in the file (“gameinventory.txt”) contents and display them onscreen to the user (see below)



```
Item    Cost    Quantity  
sword   15      1  
mace    30      1  
shield  50      1  
potion  15      12  
cloak   24      5  
dagger  24      1  
Press a key to return to main menu.
```

- **void add_item()** – This method will prompt the user for an item name, item cost, and item quantity and then append the item to the file (“gameinventory.txt”) (see screenshot below)



```
Enter the name for this new item: dagger  
Enter unit cost for this item: 24  
Enter quantity in stock for this item: 1  
Item added to the inventory!  
Press a key to return to main menu._
```

Additional Program Requirements:

- Any number of choices must be handled in any one run.
- Invalid menu choices must be properly handled
- The program must prompt the user if the file cannot be opened (file exception handling)
- The program must clear the screen after each menu option selection.
 - Hint: Use `system(“cls”);`

Your application will be graded on the following criteria. You will be awarded points for each individual rubric item that you complete:

Application/Execution

Application presents user with correct options menu	_____ 10 pts
Application "Display inventory" menu item correctly displays all items from "gameinventory.txt"	_____ 30 pts
Application "Add Item" menu item correctly prompts user for new item attributes and correctly appends the item to the file	_____ 40 pts
Application "Quit" menu item successfully exits program	_____ 10 pts
Screen is cleared after each menu operation	_____ 10 pts

Application/Code Structure

File input instance variables are declared and initialized	_____ 20 pts
File output instance variables are declared and initialized	_____ 20 pts
I/O error handling is done if files cannot be opened	_____ 20 pts
Method to display inventory items is implemented (display_inventory)	_____ 40 pts
Method to add new inventory item is implemented (add_item)	_____ 40 pts
Invalid menu choice error handling implemented	_____ 20 pts
Methods are properly commented	_____ 10 pts
Variables are properly commented	_____ 10 pts
Code copied to USB drive and program runs from USB	_____ 20 pts

Total Points: _____ 300 pts