

JAVA PROGRAMMING

(340)

REGIONAL – 2017

Production Portion:

Program 1: School Vending Machine _____ (400 points)

TOTAL POINTS _____ (*400 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 ten (10) minutes orientation
No more than ninety (90) minutes testing time
No more than ten (10) minutes wrap-up

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

You will have 90 minutes to complete your work.

Your name and/or school name should *NOT* appear on any work you submit for grading.

Create a folder on the flash drive provided using your contestant number as the name of the folder. Copy your entire solution/project into this folder. You must submit your entire solution/project so that the graders may open your project to review the source code. You must ensure that the files required to run your program are present and will execute on the flash drive provided. Note that the flash drive letter may *not* be the same when the program is graded as it was when you created the program. It is recommended that you use relative paths rather than absolute paths to ensure that the program will run regardless of the flash drive letter. The graders will *not* compile or alter your source code to correct for this. Submissions that do *not* contain source code will *not* be graded.

Assumptions to make when taking this assessment:

- No external files are needed for this project

Development Standards:

- Your Code must use a consistent variable naming convention.
- All subroutines, functions, and methods must be documented with comments explaining the purpose of the method, the input parameters (if any), and the output (if any). If you create a class, then you must use Javadoc comments.

Your application will be graded on the following criteria:

Solution and Project

- The project is present on the flash drive _____ 10 points
The projects main class is named SchoolVending _____ 10 points

Program Execution

- The program runs from the USB flash drive _____ 15 points

If the program does *not* execute, then the remaining items in this section receive a score of zero.

- The program runs and produces the initial menu screen _____ 20 points
The program adds the appropriate item cost total sale value based on the input option _____ 20 points
The program keeps track of the number items selected _____ 20 points
The program resets the transaction to all zeros when the “Clear Transaction” option is selected _____ 20 points
The program clears the screen every time the menu is displayed _____ 20 points
The program displays an error message and pauses for approximately two seconds if an illegal entry is made. _____ 40 points
The “Current Sales Total” is formatted to have a dollar sign and shows only two decimal places. _____ 30 points
The program displays the final output screen when appropriate _____ 30 points
Output looks like the examples given _____ 35 points

Source Code Review

- The source code is properly commented
 A comment containing the contestant number is present _____ 10 points
 Methods and code sections are commented _____ 20 points
A method exists to display the menu and validate entry _____ 40 points
A method exists to perform the sleep/pause function _____ 20 points
A method exists to perform the clear screen function _____ 20 points
Code uses a consistent variable naming convention _____ 10 points
Methods are used appropriately _____ 10 points

Total Points: _____ / 400 points

School Vending Prototype

Professional Business Associates has been asked to develop the prototype software to be used in a school vending machine. These vending machines will be used to sell school supplies. The concept of the vending machine is that it will be able to sell multiple items per transaction. The user will make choices via a menu system and the machine will tell the user how much the entire transaction will cost.

Menu:

The menu will display the input option, item description, and item price. See the example below for the list of items to be offered and their cost. Due to the hardware design of the display screen on the vending machine, the menu must be must look identical to the example found on the next page.

Input:

Since this is a prototype design, the program's input will be received from the keyboard. All input should be validated for validity. The program should be able to handle input of both upper and lower case. For example the input of "a" and the input of "A" should be considered the same. The program should continue to take input from the user until the user enters either an upper or lower case "H" or an upper or lower case "X".

Output:

The output should be as close to the examples below as possible.

Requirements:

1. You must create an application with the main class named SchoolVending.
2. Your contestant number must appear as a comment at the top of the main source code file.
3. The program should use **at least** three methods besides the main. The first method is the menu. It should clear the current screen and it should print out the current number of items selected and the cost of the transaction (see Menu Screen figure on the next page). The menu should validate that only the appropriate options have been entered. If an invalid entry is entered, the program should print a message to the user and pause for approximately 2 seconds before redisplaying the menu (see Illegal Entry figure on Page 6).
4. The second method should handle the "pause/sleep" for the invalid entry.
5. The third method should handle the "clear screen" operation.
6. If you order all six items on the menu and select the "Exit and Pay" option, your program should print the final output screen (see Final Output figure on Page 6).



