

C++ PROGRAMMING (43)

Regional – 2013

TOTAL POINTS _____ (210)

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.***

Complete and submit as directed by State Advisor

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.

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.

SCENARIO:

The game of Checkers is played on an 8 x 8 grid. The checkers are placed on the black squares only. Checkers move towards the other end of the grid by moving to an unoccupied, adjacent black square. That is a checker at location (1,1) would move to location (2,2) if it were unoccupied. Checkers may “jump” over and capture an opponent’s checker if that checker is in an adjacent black square and the landing black square is unoccupied. That is a checker at location (1,1) could “jump” over an opponent’s checker at location (2,2) if location (3,3) was unoccupied. For this game your “home row” is row 1 and your opponent’s home row is row 8. If your checker lands in your opponent’s home row, the checker becomes a “king” and can move both forwards and backwards from adjacent Black Square to adjacent Black Square.

For this program we are restricting the movement of your checker so that it can only move from row 1 to row 8. Further, if there is a jump to be made from a location, then just one jump will be possible.

INPUT: In this problem, you will be given the location of one of your checkers, followed by the number of your opponent’s checkers and their locations. Locations will be given in ordered pair format (row, column). Contestant is to enter the following sample input at the console. Sample Input line #1 below indicates that your checker is at location (1,5) and your opponent has 3 checkers at locations (2,6), (4,6) and (6,6). All zero’s for input indicates that the program should stop running.

OUTPUT: For each input, it is your turn to move a checker. Print the maximum number of legal jumps that can be made by your checker. Also, if your checker finishes in the “home row” of your opponent, then print the word “king” next to your answer for the number of jumps.

SAMPLE INPUT

1. 1, 5, 3, 2, 6, 4, 6, 6, 6
2. 2, 2, 4, 3, 3, 5, 3, 7, 3, 7, 5
3. 0, 0, 0

SAMPLE OUTPUT

3
3, KING
<nothing will print>

C++ PROGRAMMING TECHNICAL TASK
REGIONAL 2013
PAGE 3 of 3

Your application will be graded on the following criteria:

Application Execution

• Application reads input that is comma separated from command line	_____ 15 pts
• Application loops until all zeros are read	_____ 15 pts
• Application correctly determines the number of moves	_____ 50 pts
• Application correctly determines KING	_____ 50 pts
• Application displays number of moves and KING when applicable	_____ 25 pts
Application exits program when all 0's entered with no output	_____ 15 pts
Code uses a data structure to represent checker board	_____ 20 pts
Code is well commented	_____ 10 pts
Code uses proper variable naming conventions	_____ 5 pts
Program runs from USB drive	_____ 5 pts
 Total Points:	 _____ 210 pts