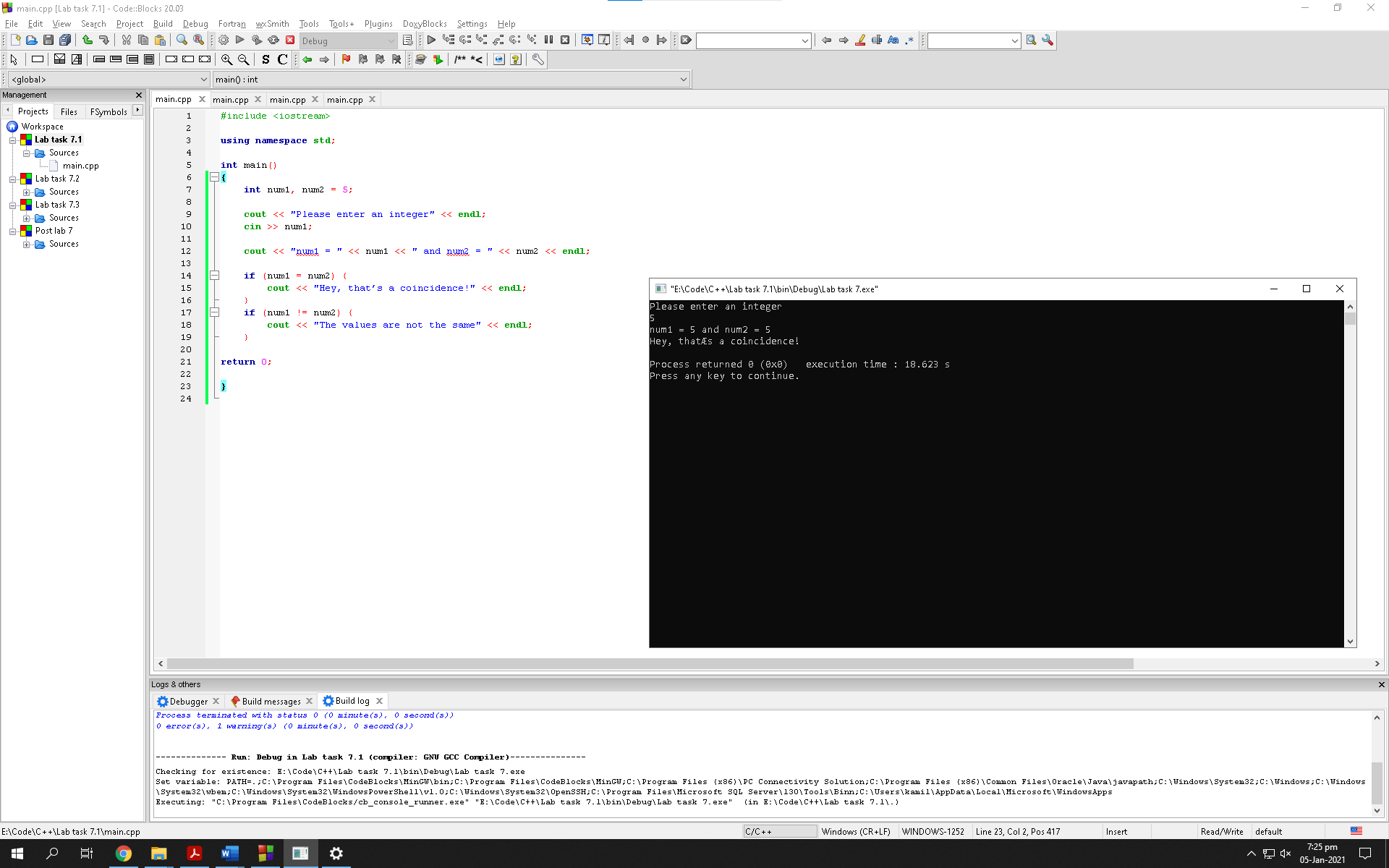
Lab task 7.1

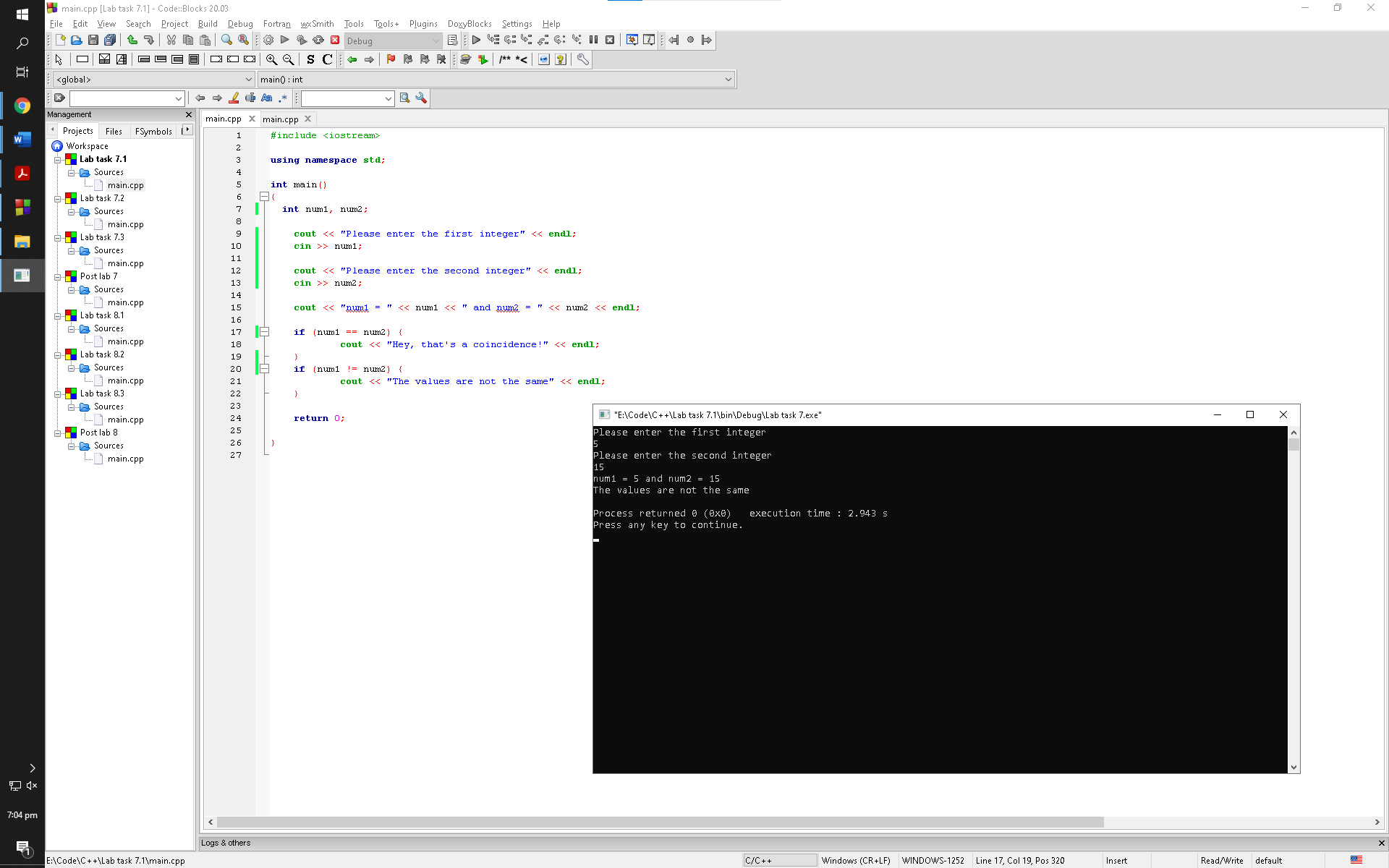
Exercise 1



Observation: The program does not go beyond the first if statement because in it the condition has an assignment operator instead of equals to operator “==” which makes the condition true for every number

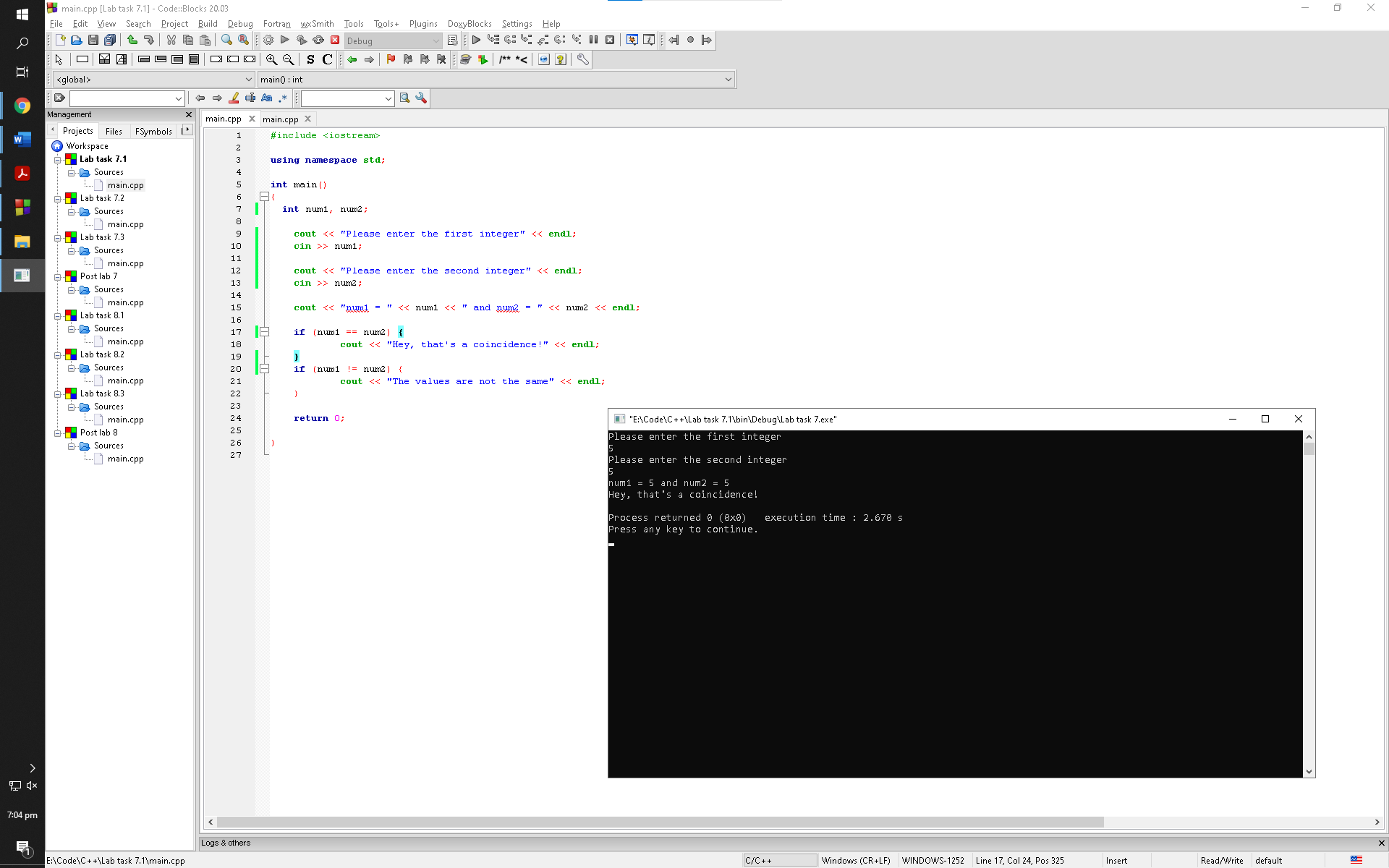
Exercise 2

Different:



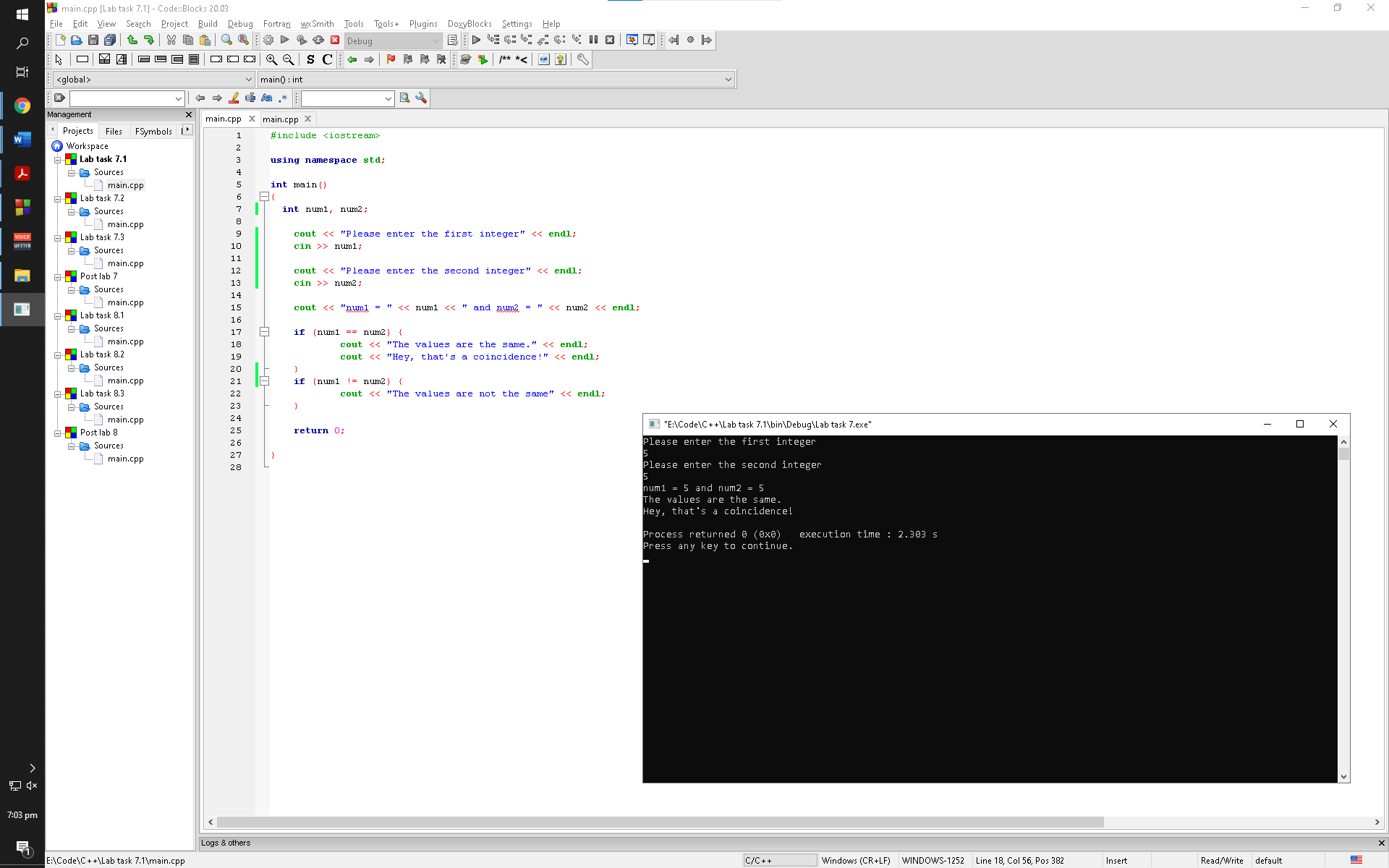
Observation: When the user inputs two different numbers, the program prints out the cout statement stated in the second if statement since the condition for the first if statement is invalid and the condition for the second is valid

Same:



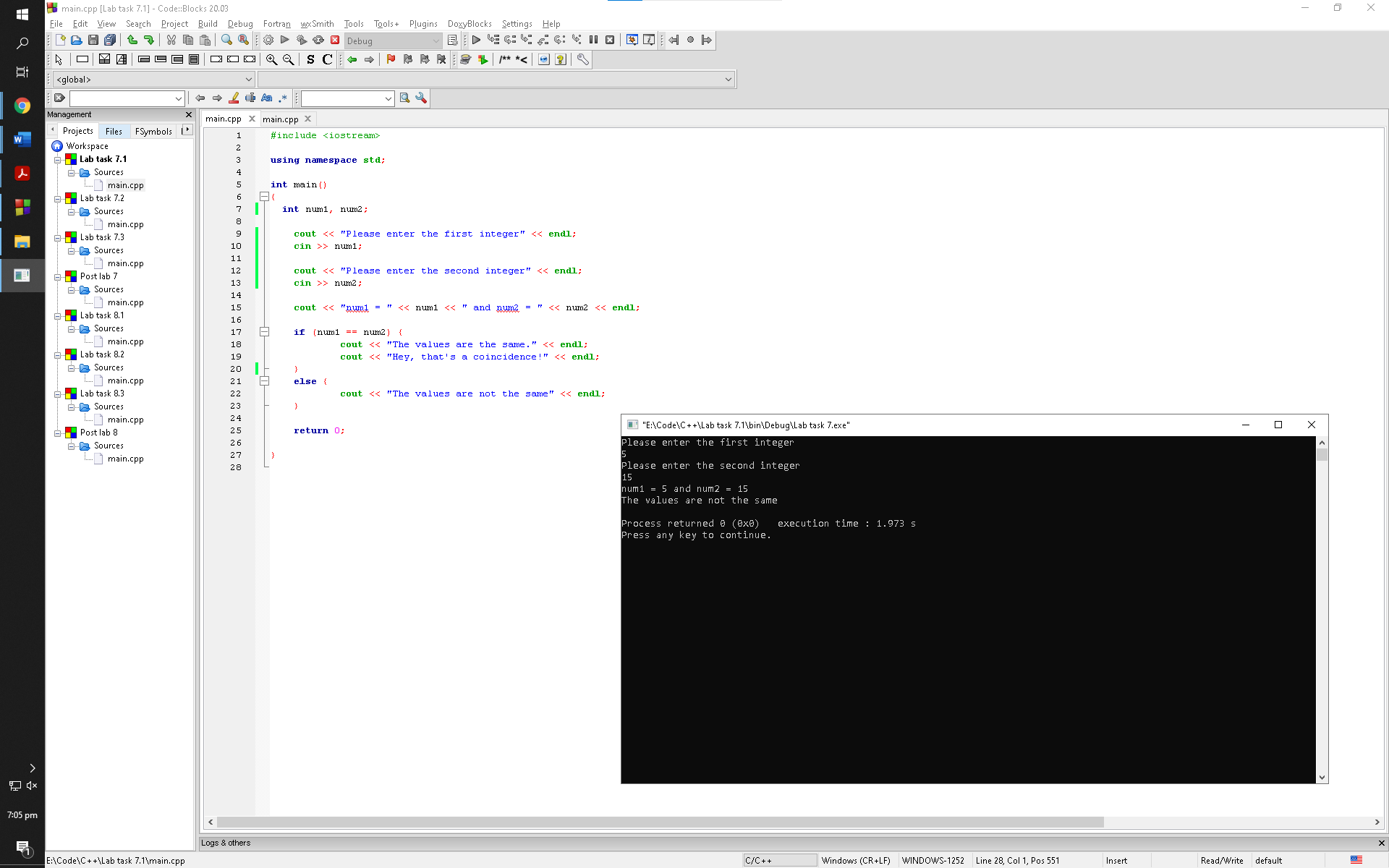
Observation: When the user inputs two same numbers, the program prints out the cout statement stated in the first if statement since the condition for the second if statement is invalid and the condition for the first is valid

Exercise 3



Observation: If the user inputs two same numbers, the program checks the first if statement for whether its valid or invalid. If its valid the program prints out two cout statements from the first if statement otherwise it checks the second if statement and prints if the condition for it is true

Exercise 4:

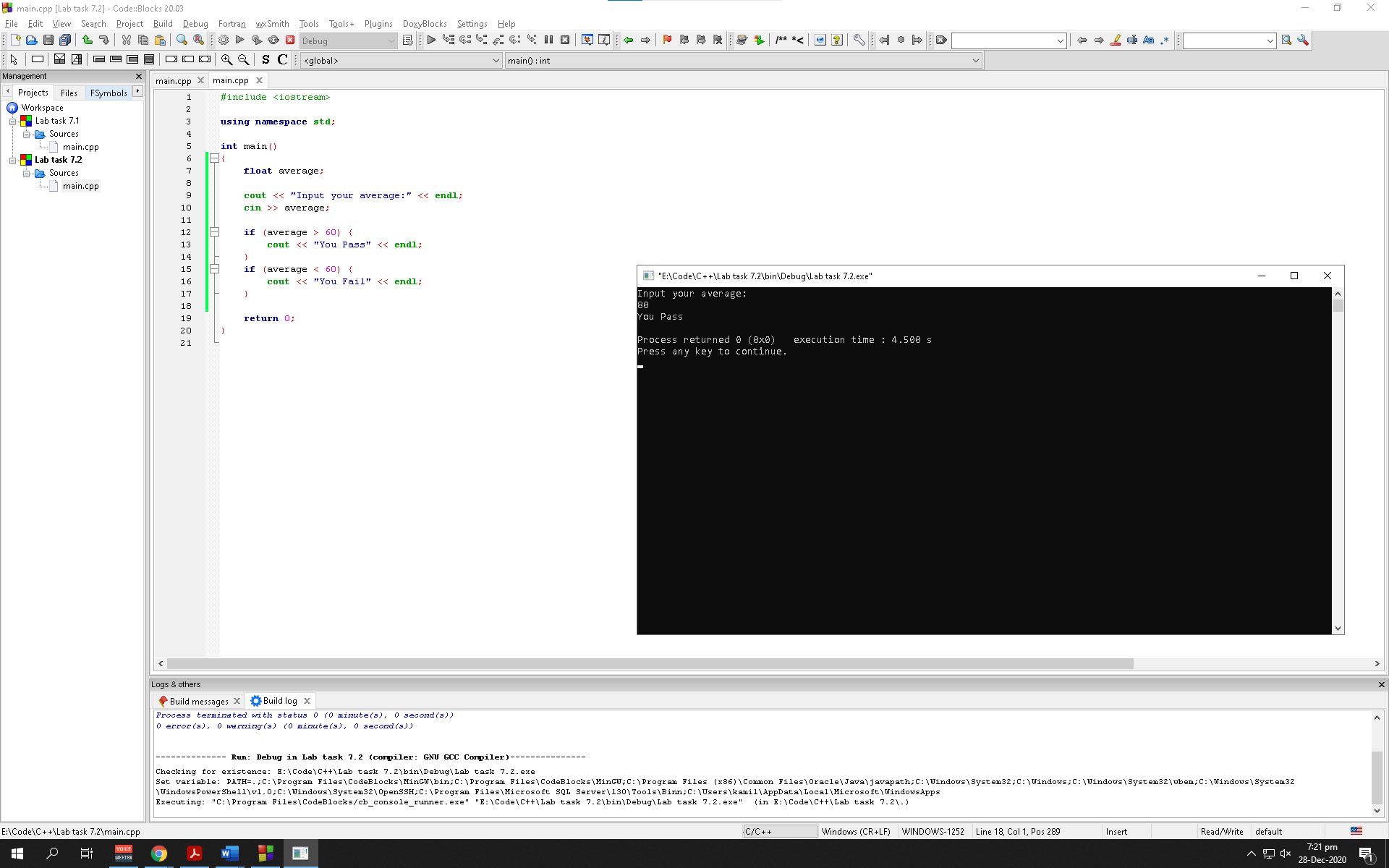


Observation: The program checks the inputted values of the user with the condition in the first if statement. If the condition is valid, the program prints the cout statement stated in the first if statement otherwise it prints out the else statement every time without checking for any condition

Lab task 7.2

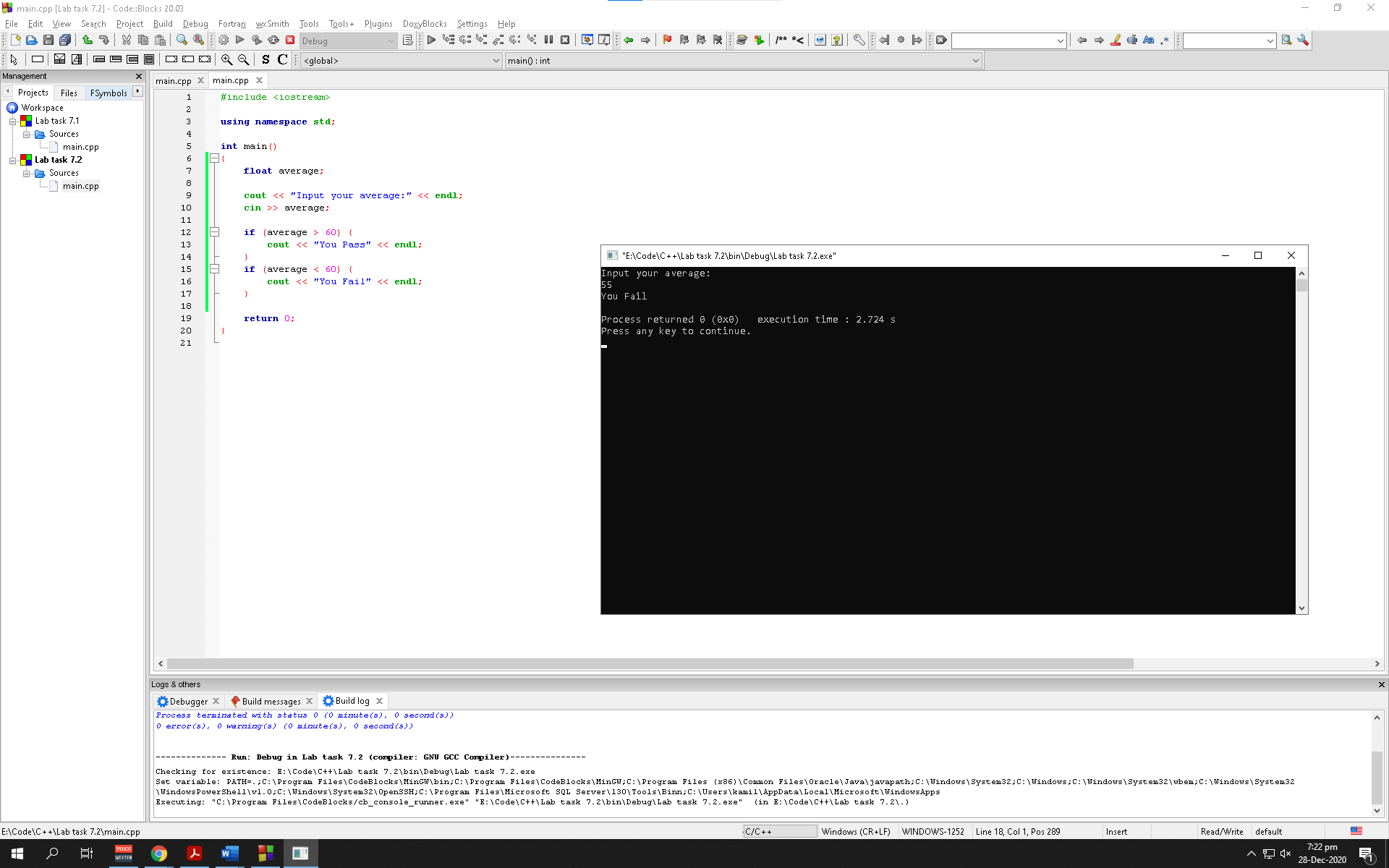
Exercise 1

Using 80:



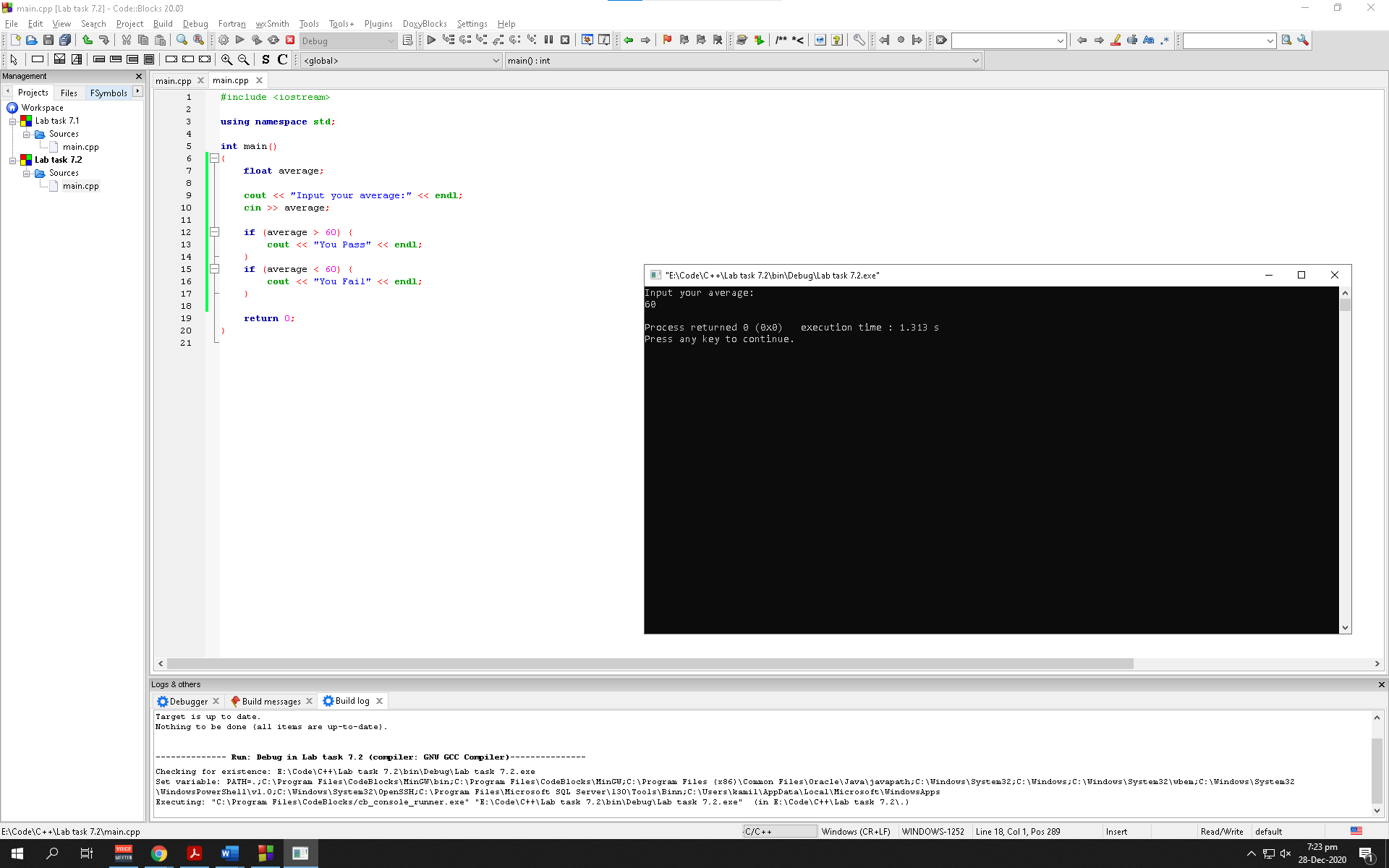
Observation: The program checks the condition of every if statement. The condition in the first if statement is valid for this scenario hence it prints out the first cout statement

Using 55:



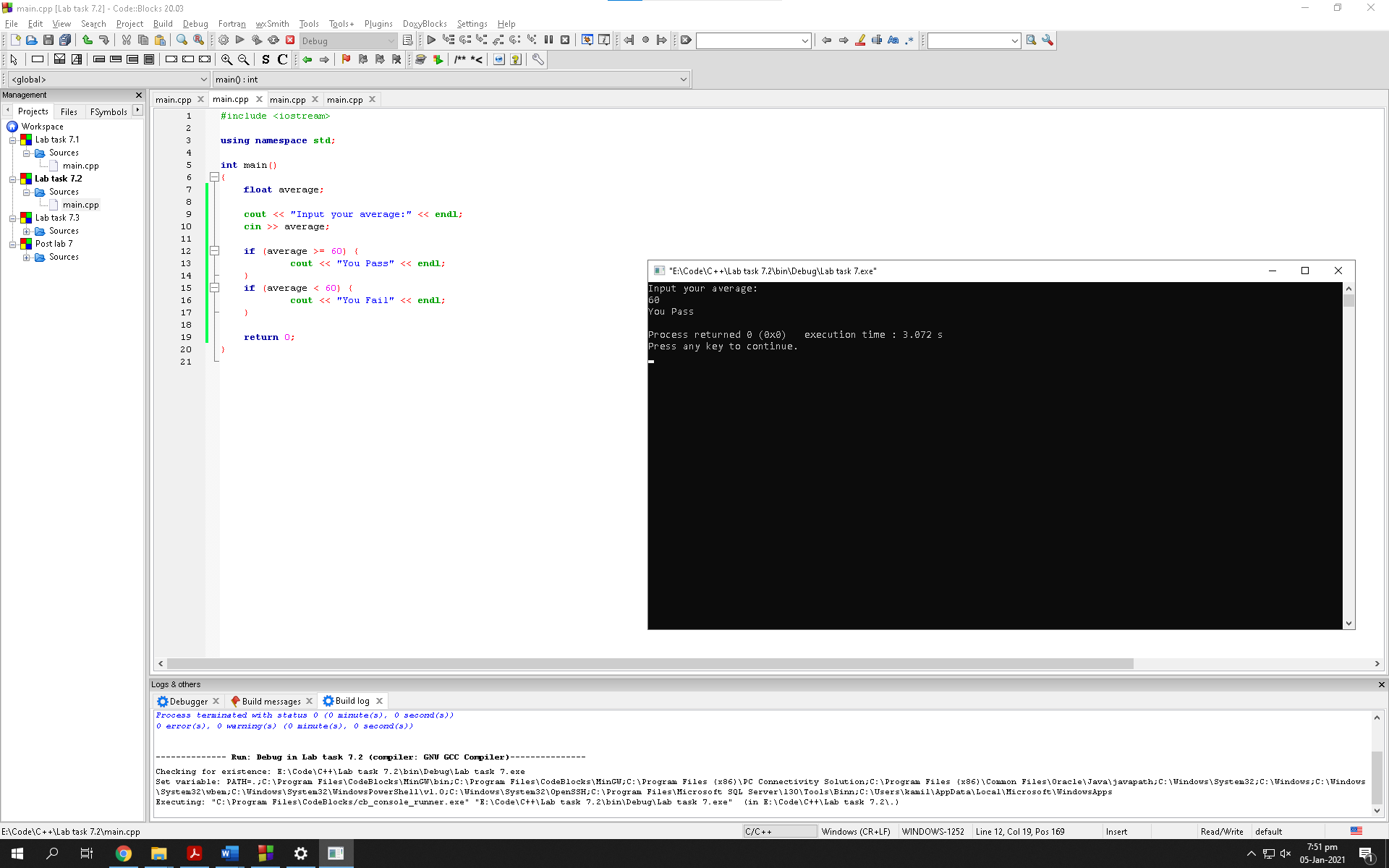
Observation: The program checks the condition of every if statement. The condition in the second if statement is valid for this scenario hence it prints out the second cout statement

Using 60:



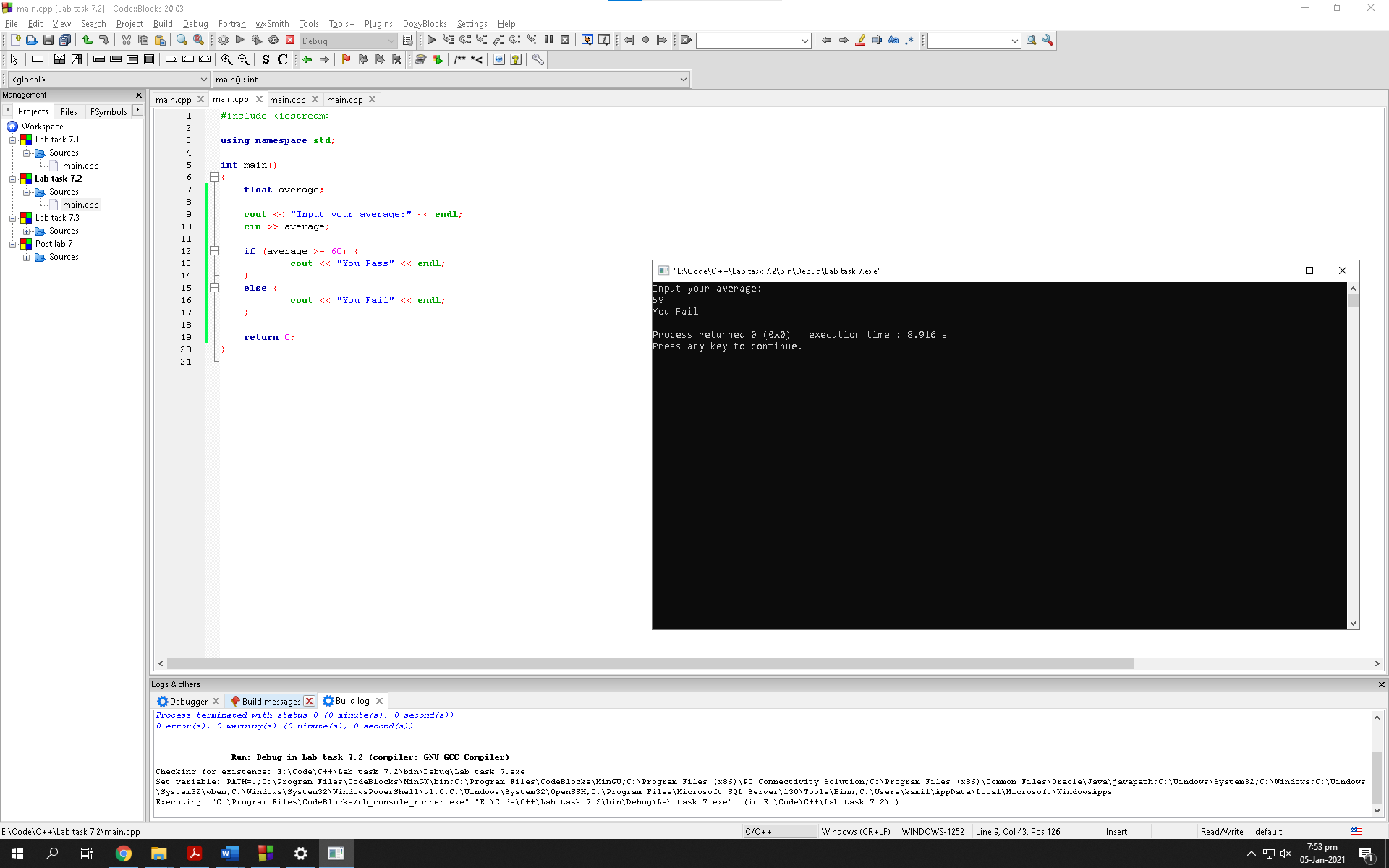
Observation: The program checks the condition of every if statement. The program is not instructed to do anything if the number 60 is inputted hence it aborts

Using 60 (After modifying):



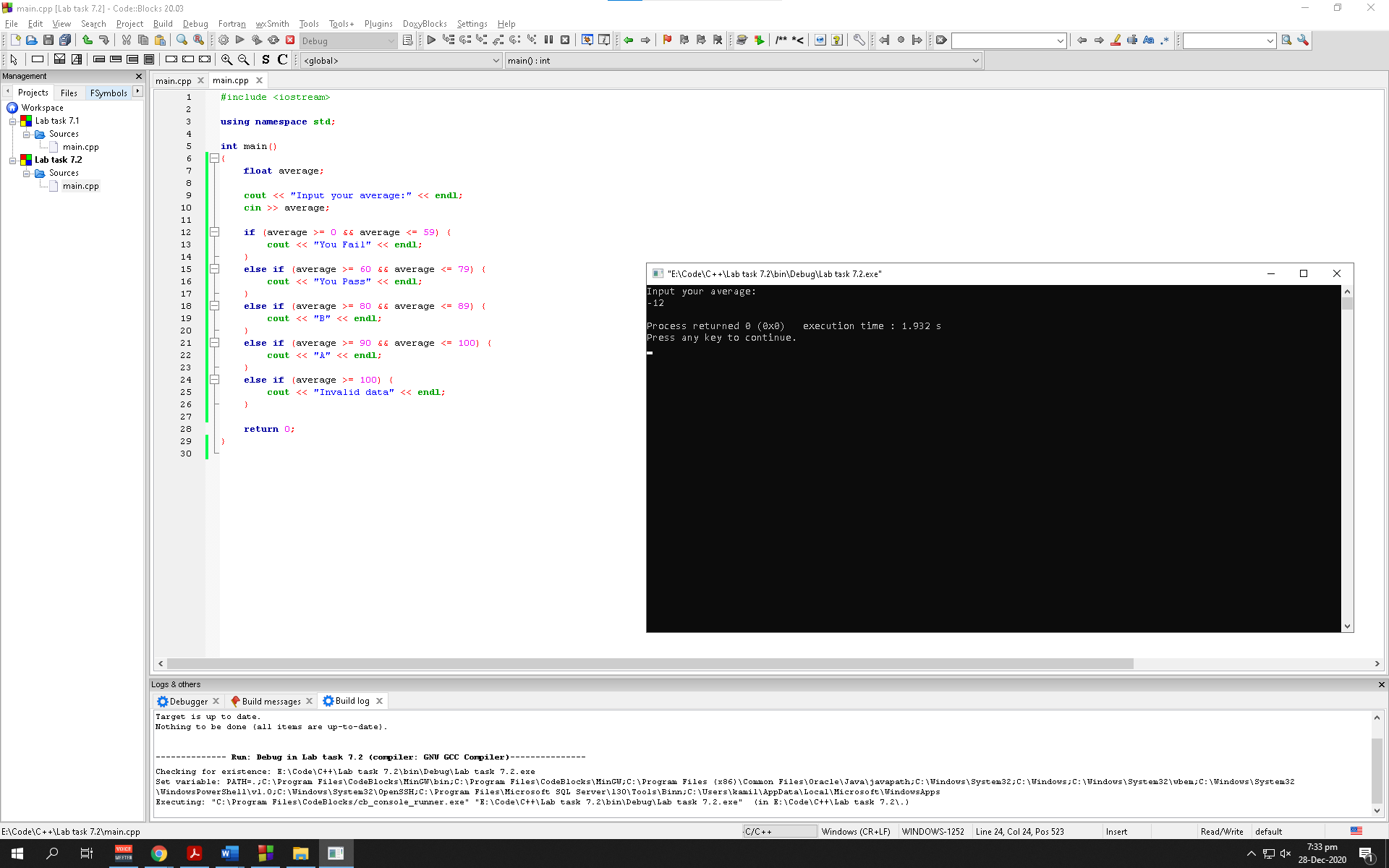
Observation: The program checks the condition of every if statement. The condition in the first if statement is valid for this scenario hence it prints out the first cout statement

Exercise 2:



Observation: The program checks the inputted value of the user with the condition in the if statement. If the condition is valid, the program prints the cout statement stated in the if statement otherwise it prints out the else statement every time without checking for any condition

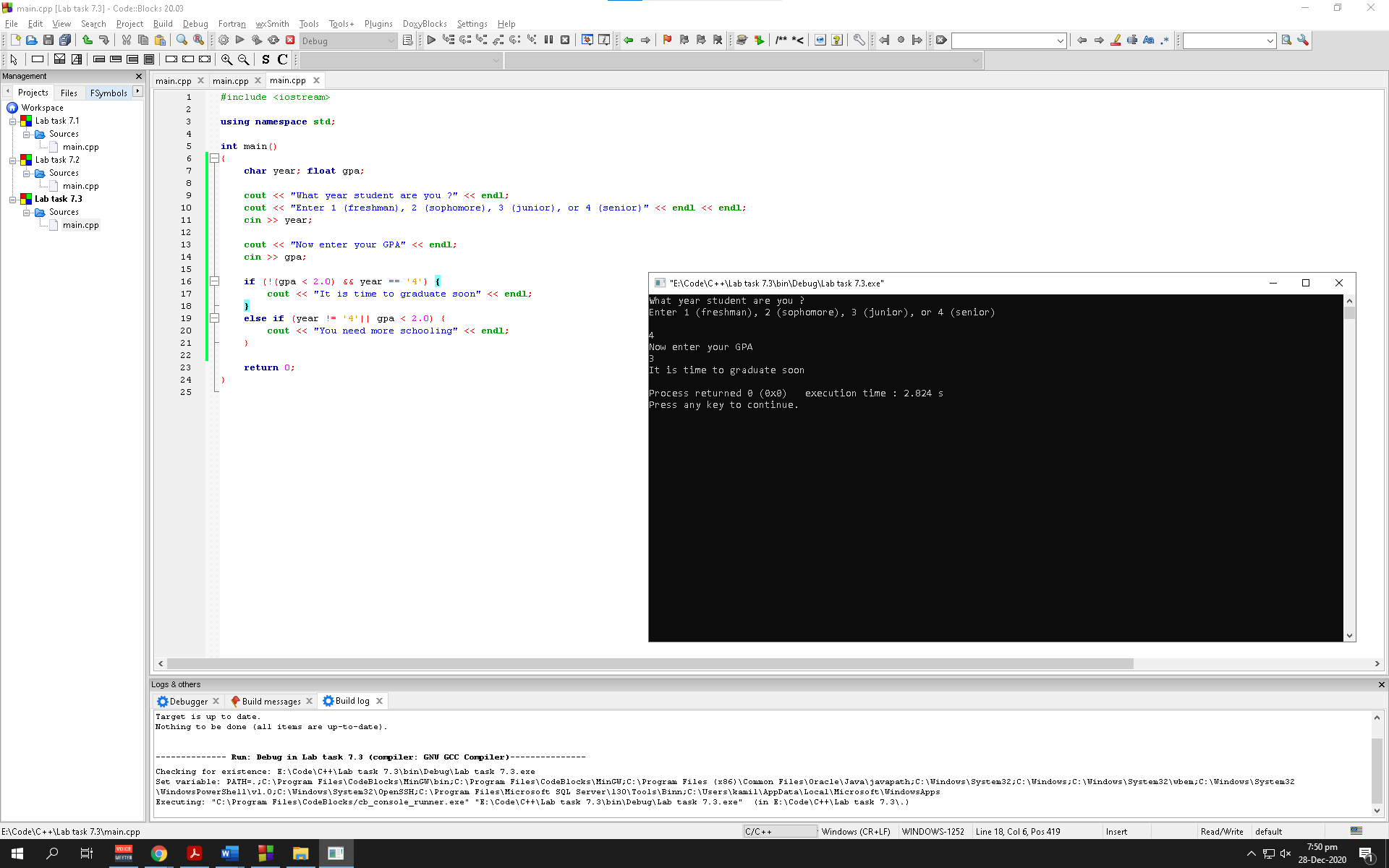
Exercise 3:



Observation: The program simply aborts since its not given any instructions to what to do when a negative number is inputted

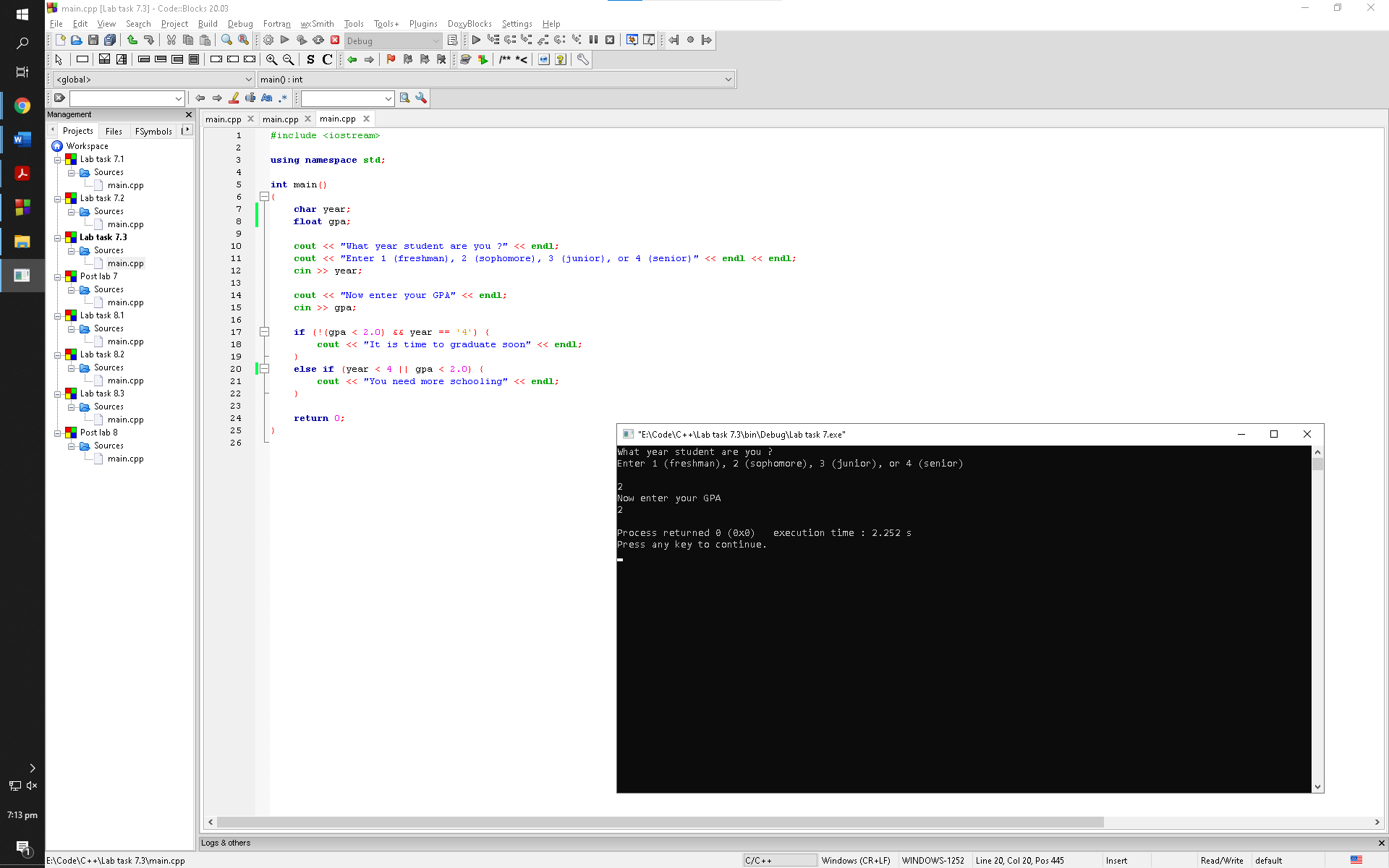
Lab task 7.3

Exercise 1:



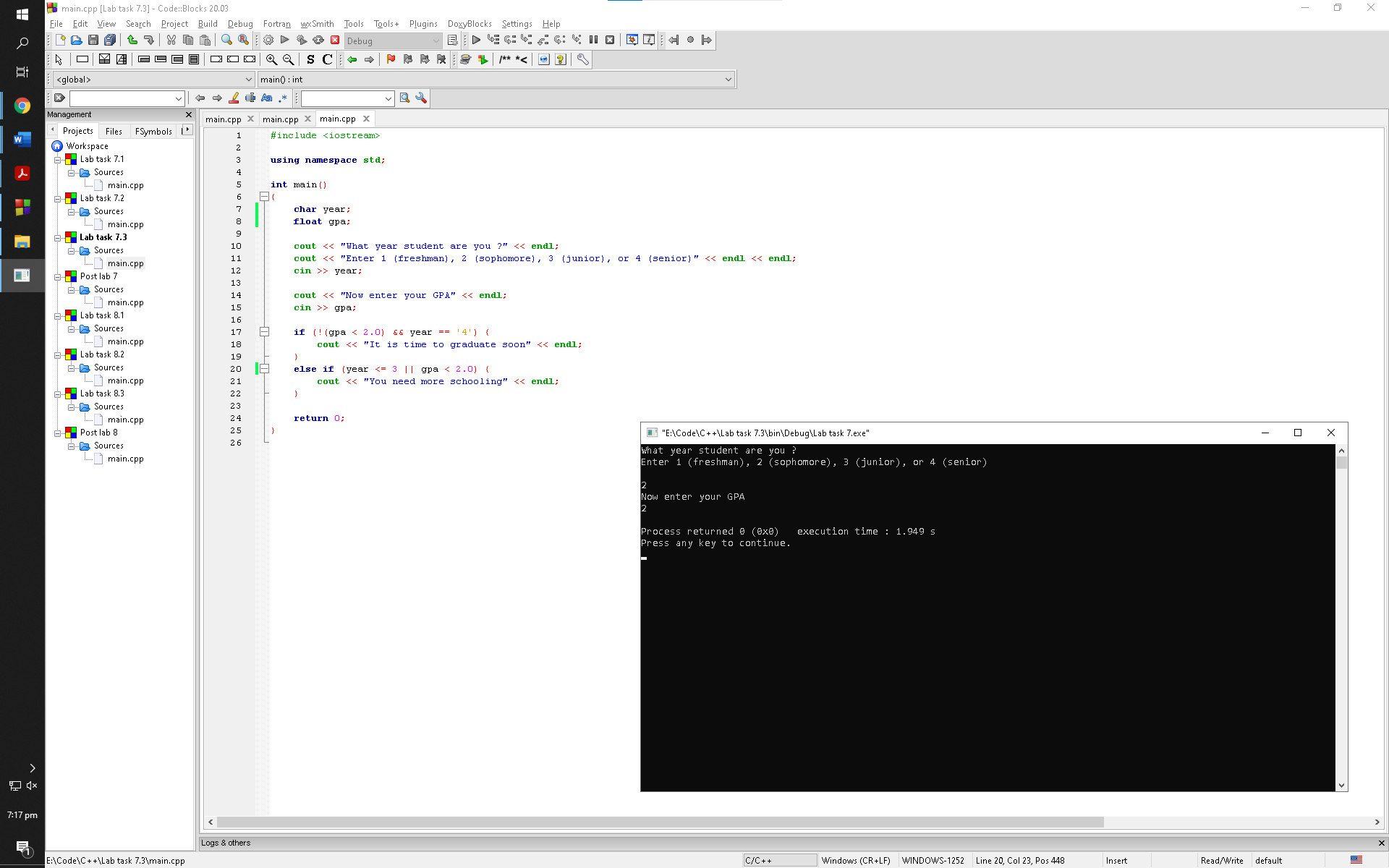
Exercise 2:

Using (year < 4):



Observation: No, we cannot. Since, variable “year” is initialized at character, the program will not execute the else if statement because the year that it needs must be an integer

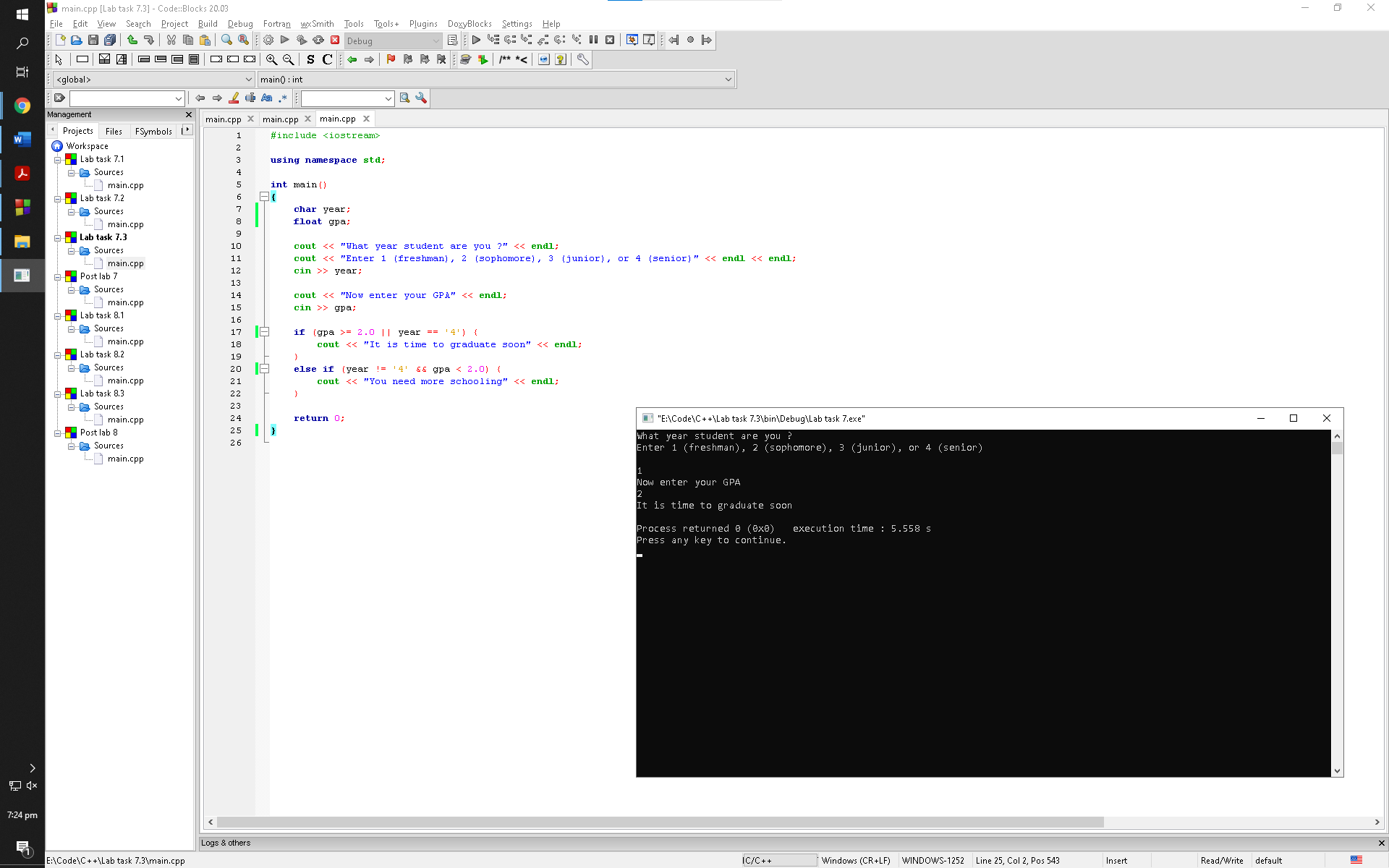
Using (year <= 3):

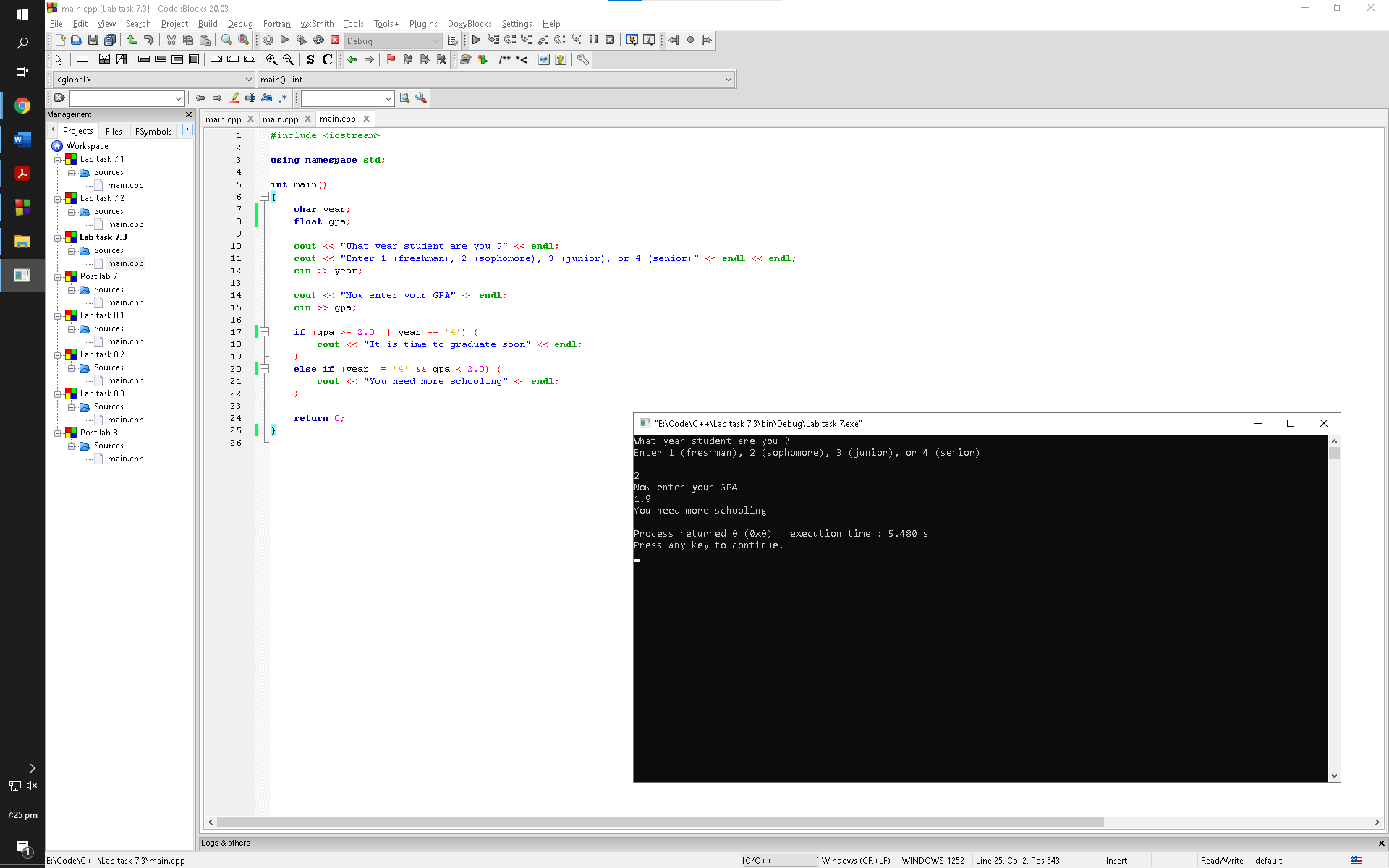


Observation: No, we cannot. Since, variable “year” is initialized at character, the program will not execute the else if statement because the year that it needs must be an integer

Post Lab:

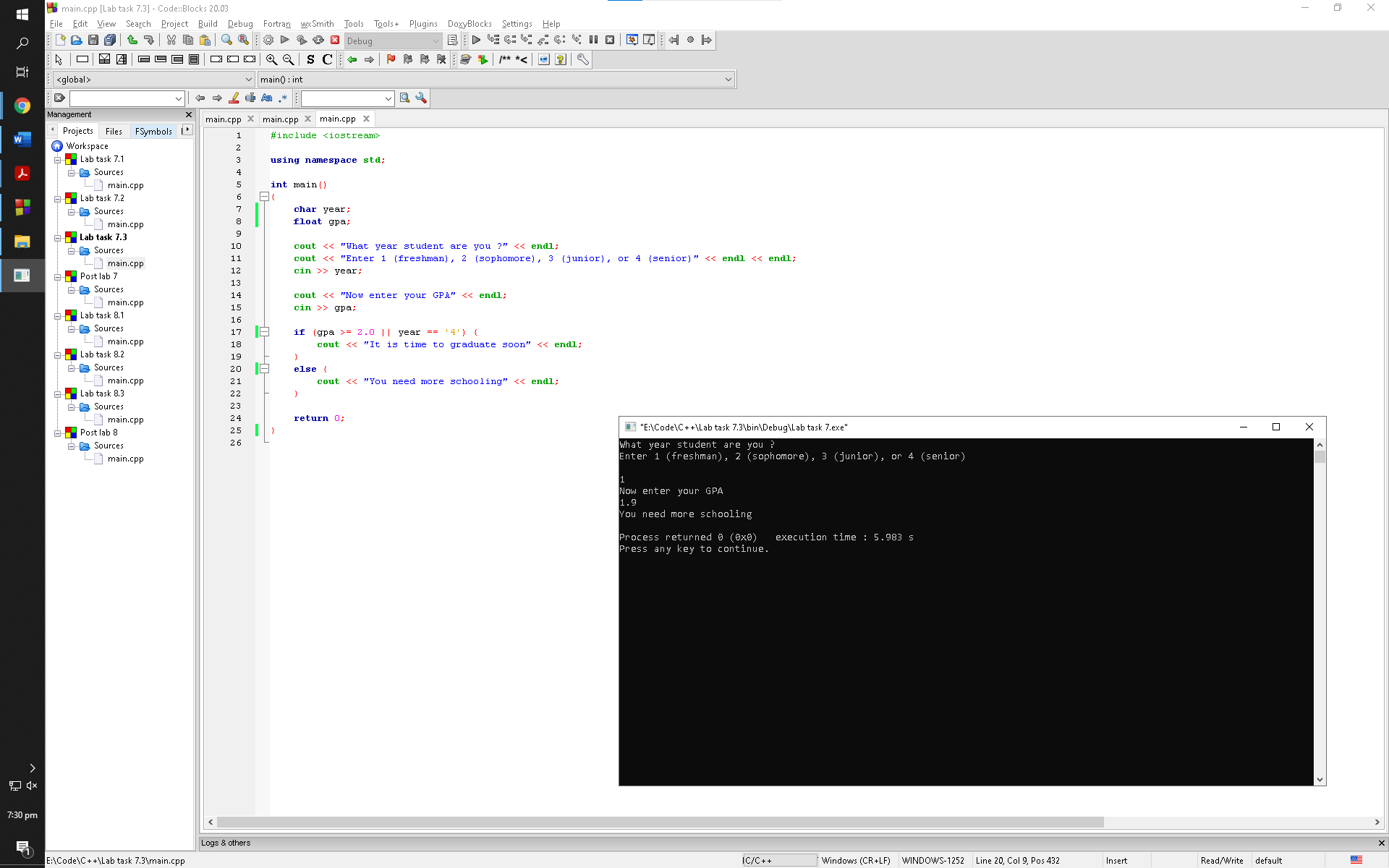
Task 1





Observation: Students of all four years having gpa of greater than equal to 2 can graduate according to this program and the students who won't graduate are the ones in year 1, 2, and 3 with a gpa of less than 2

Task 2



Observation: Students of all four years having gpa of greater than equal to 2 can graduate according to this program and the students who won't graduate are the ones in year 1, 2, and 3 with a gpa of less than 2