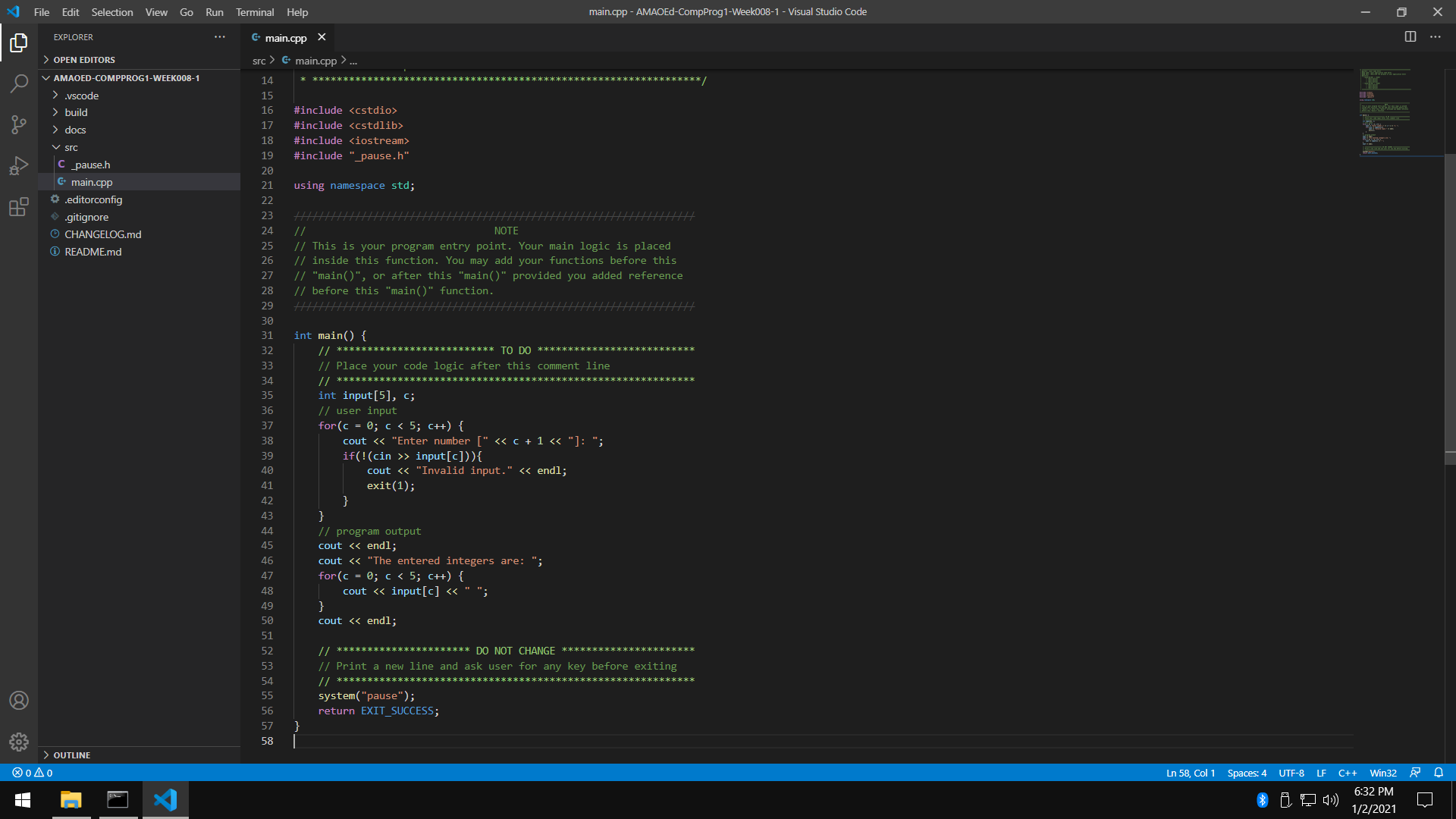
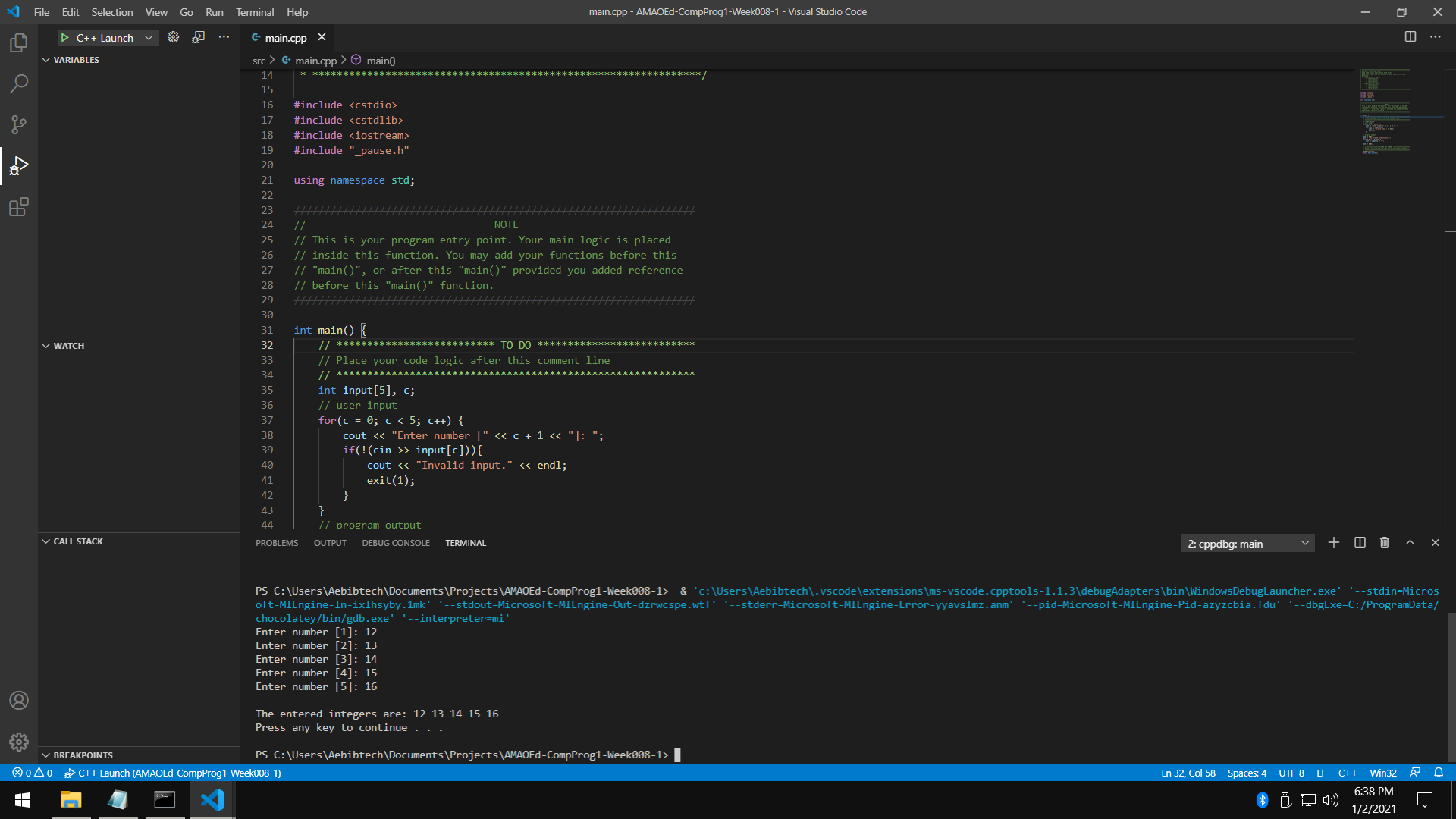
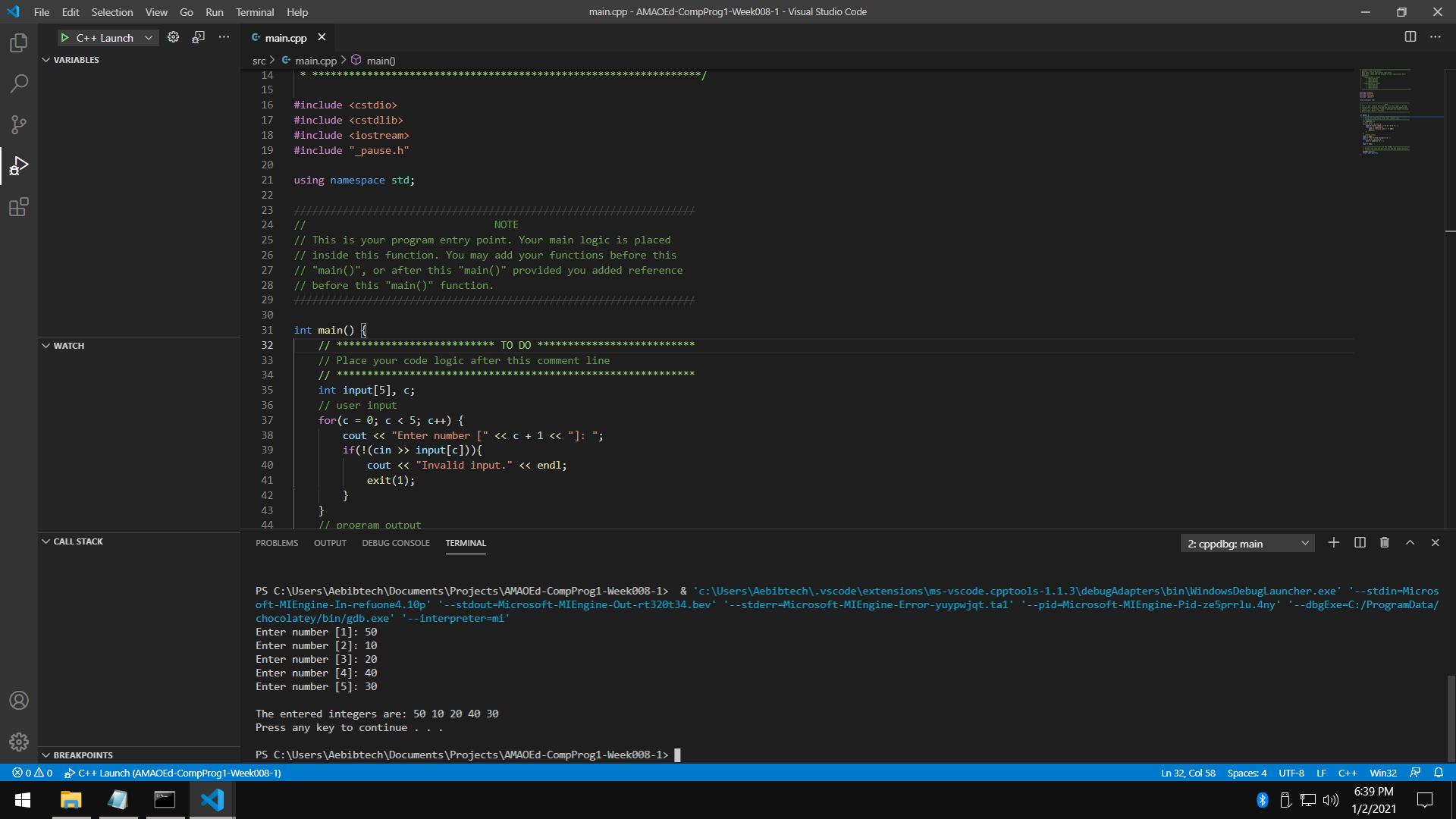
1. Write a program that will accept five (5) integers and display them to the users. You are limited to using only two (2) variables (including the array).
   1. Code



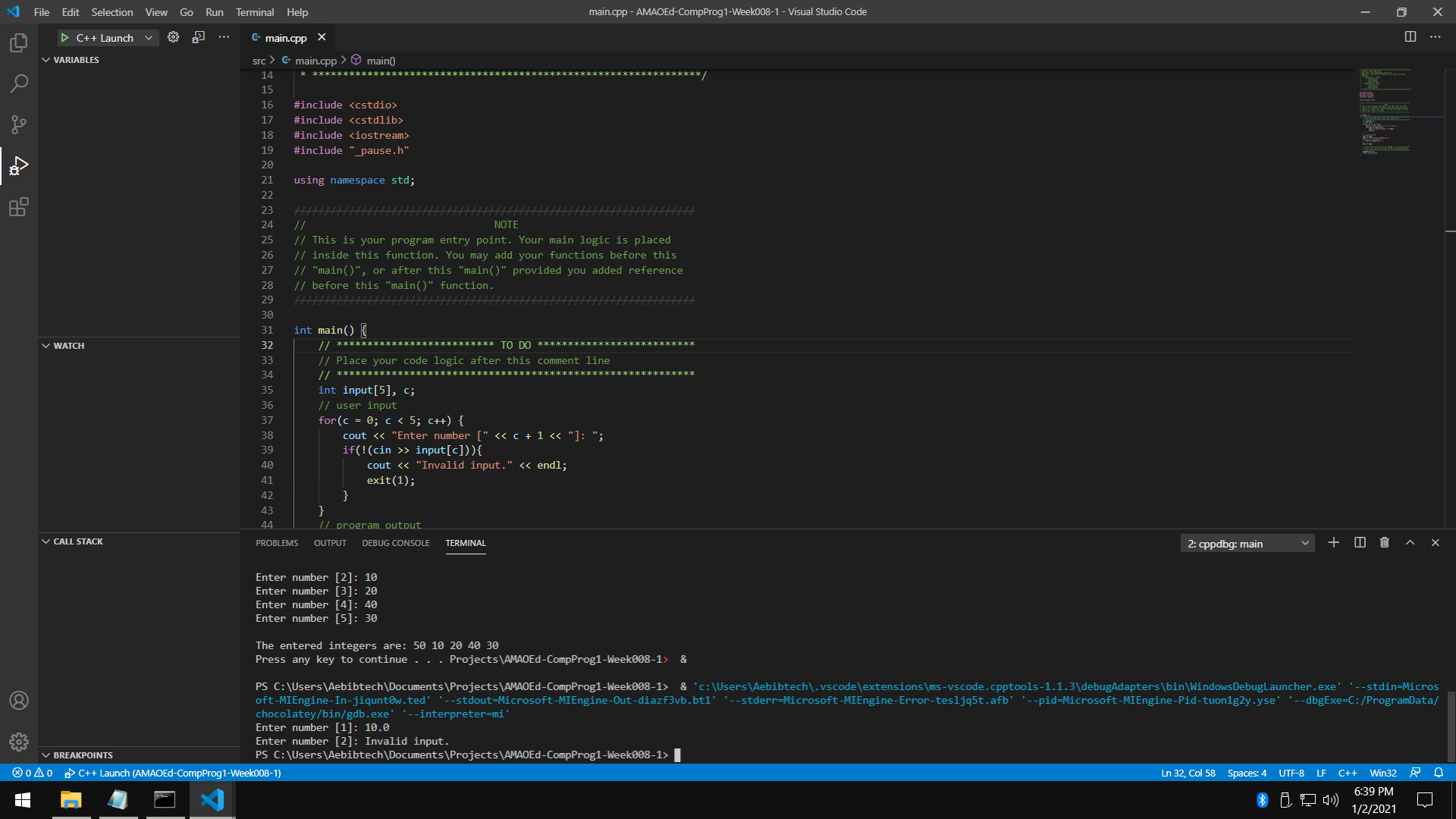
* 1. Sample Input 1



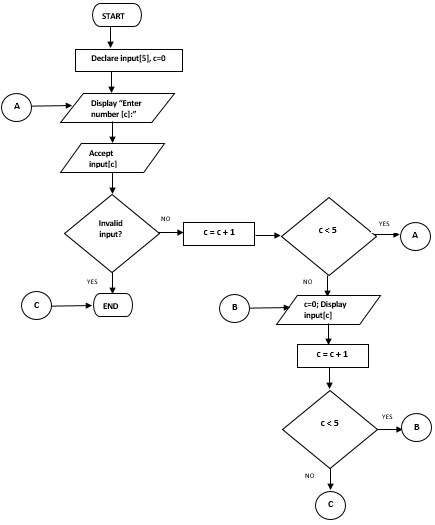
* 1. Sample Input 2



* 1. Sample Input 3



* 1. Flowchart



* 1. Pseudocode

DECLARE input [5], c

SET c AS 0

WHILE c < 5 DO

INPUT input[c]

IF (!IsValid(input[c])) THEN

OUTPUT “Invalid input.”

END

SET c AS c + 1

LOOP

SET c AS 0

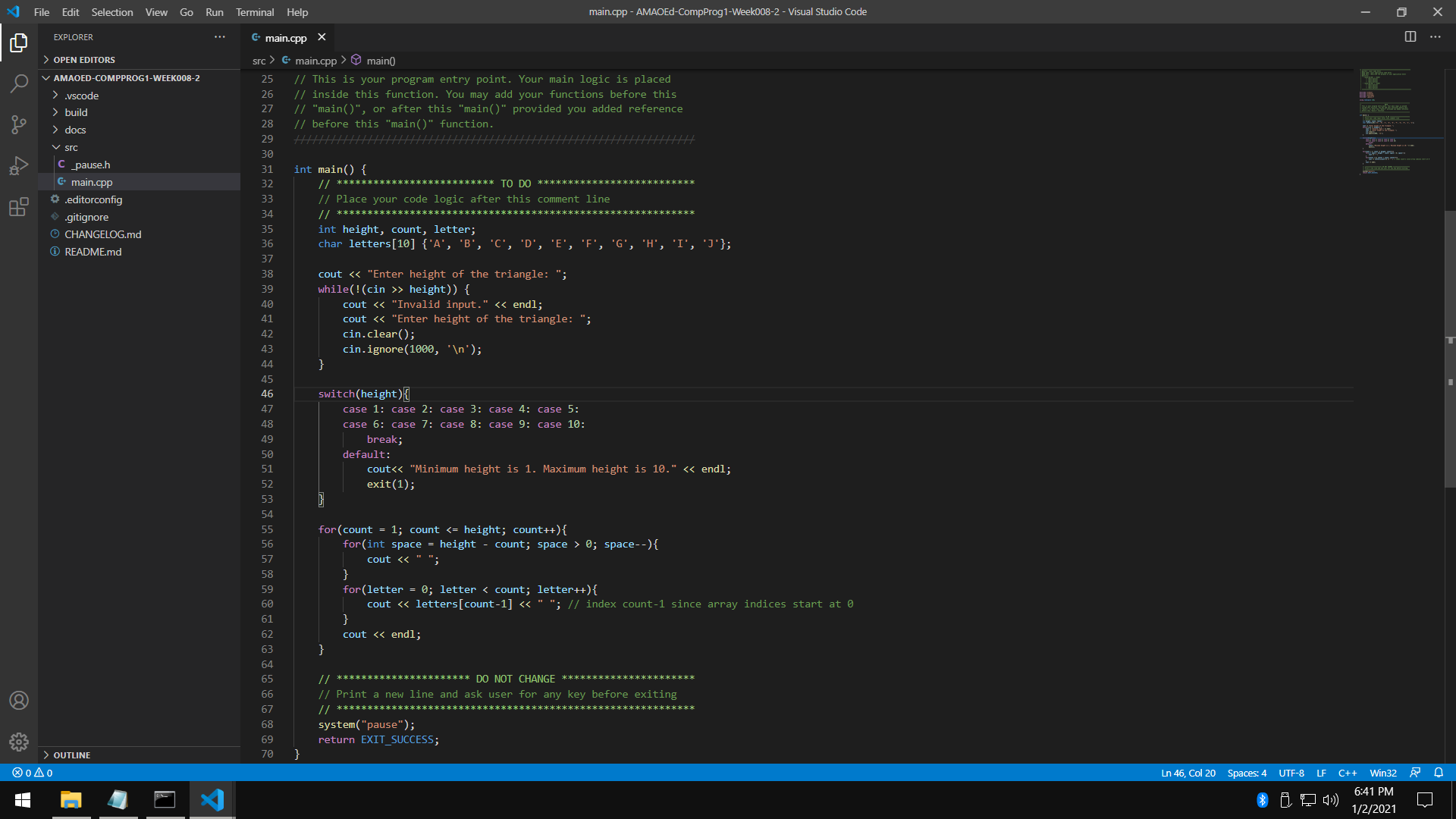
WHILE c<5 DO

OUTPUT input[c]

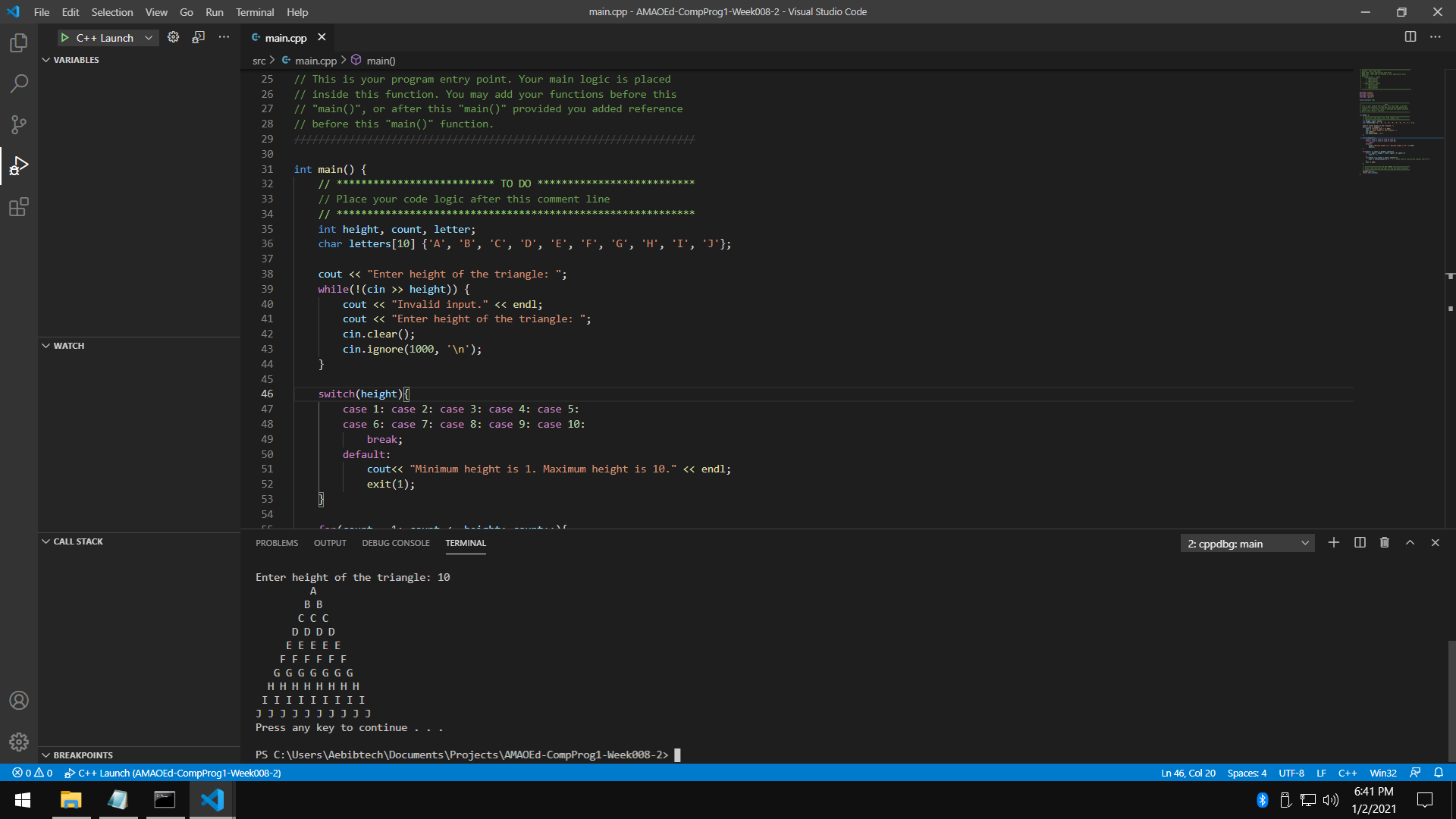
SET C AS c + 1

LOOP

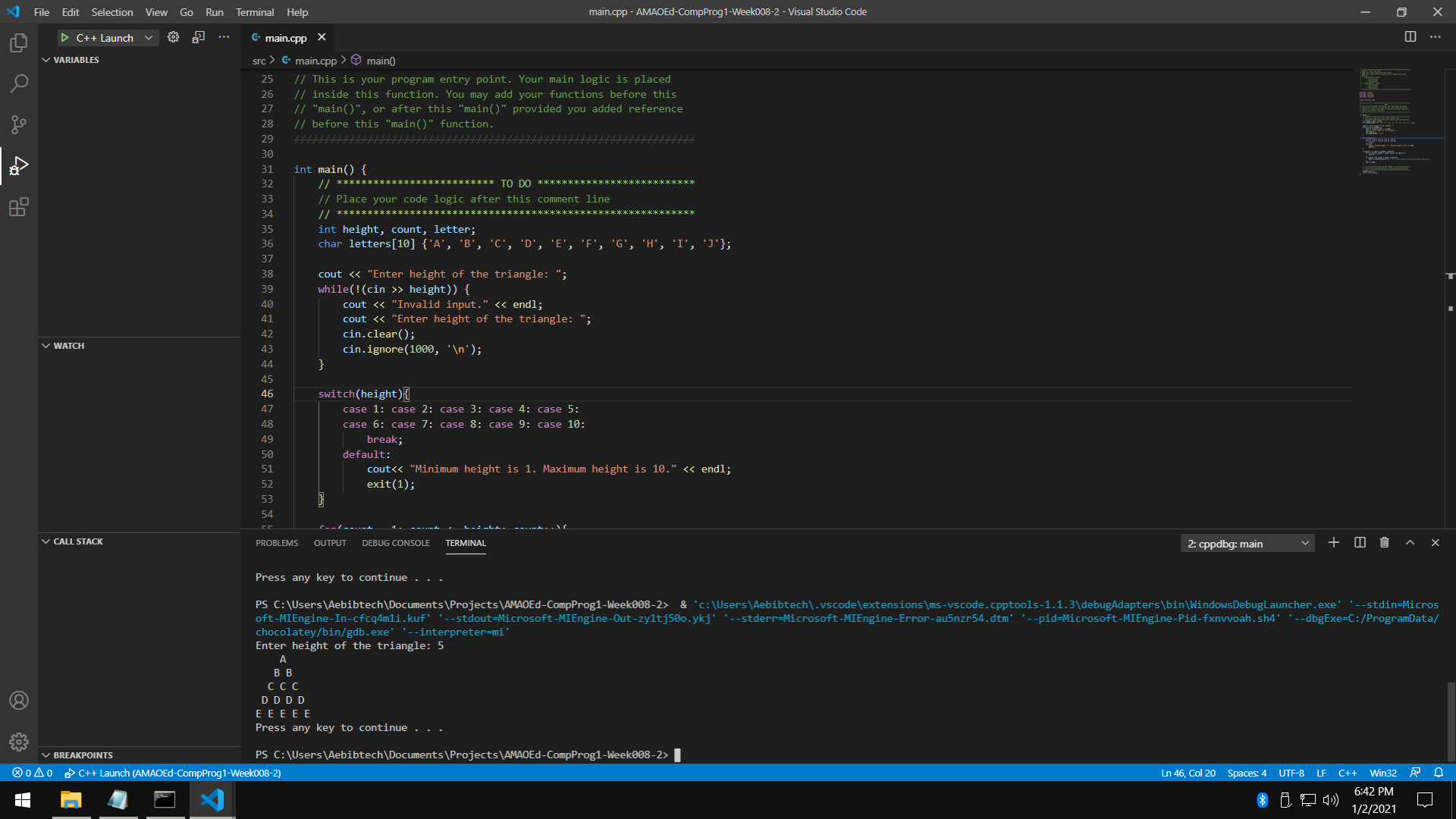
1. Write a program that will display an equilateral triangle with a height depending on the user. The minimum height is 1, the maximum height is 10. Use an array to display the specific character on the specific row. The array will be: { 0 := “A”, 1:= “B”, 2 := “C”, 3 := “D”, 4 := “E”, 5 := “F”, 6 := “G”, 7 := “H”, 8 := “I”, 9 := “J” }. You are limited to four (4) variables only (including the array).
   1. Code



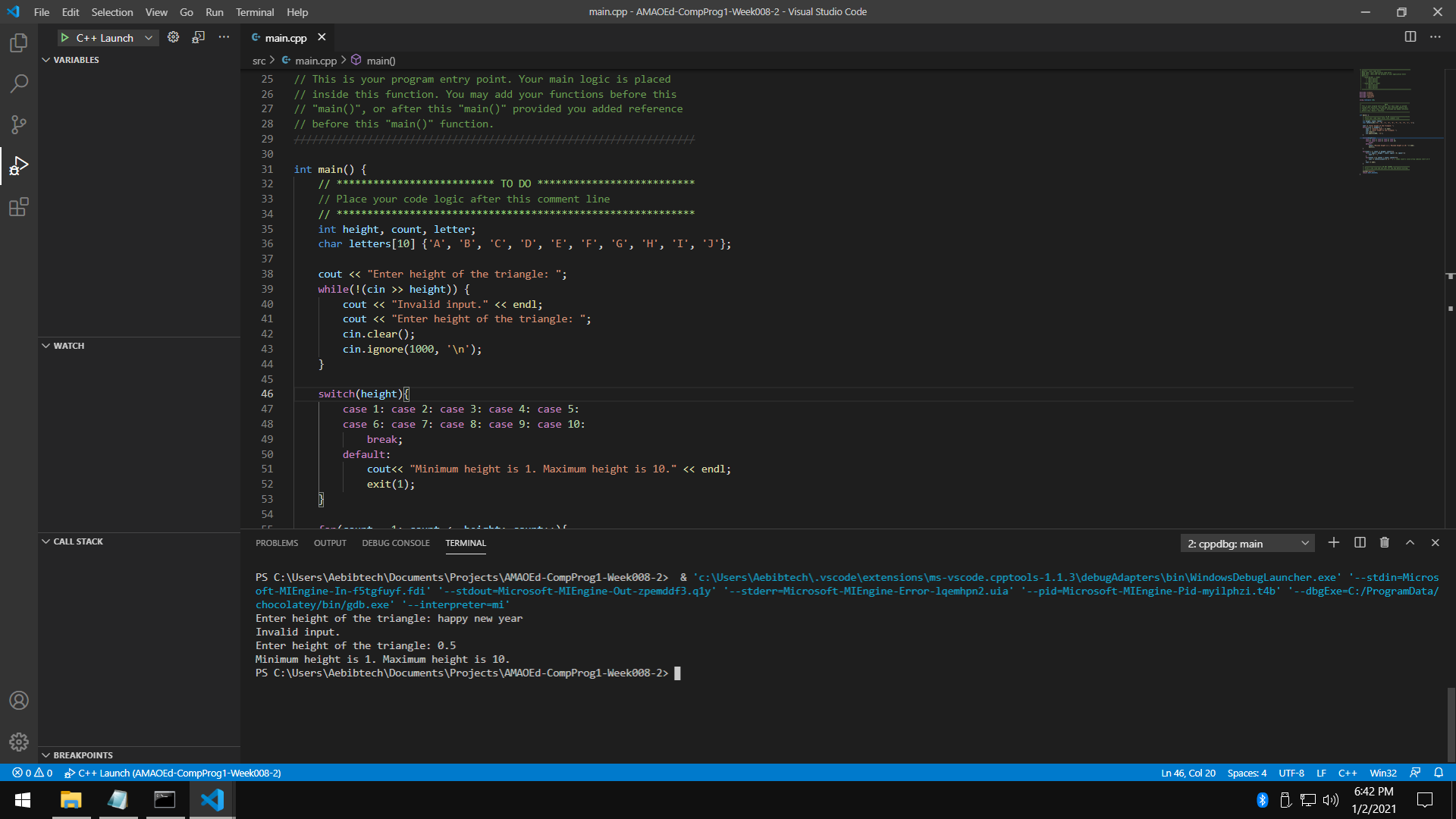
* 1. Sample Input 1



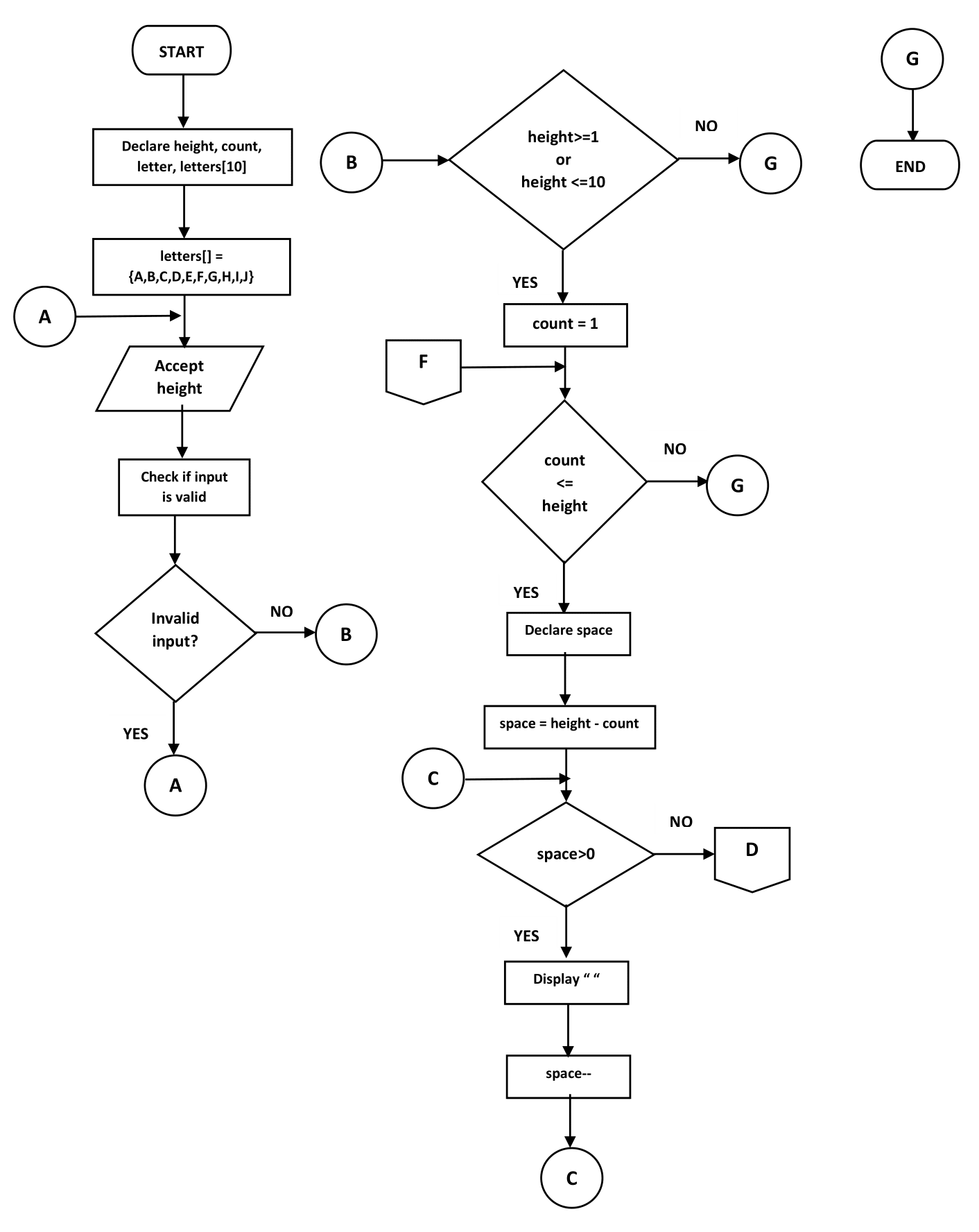
* 1. Sample Input 2

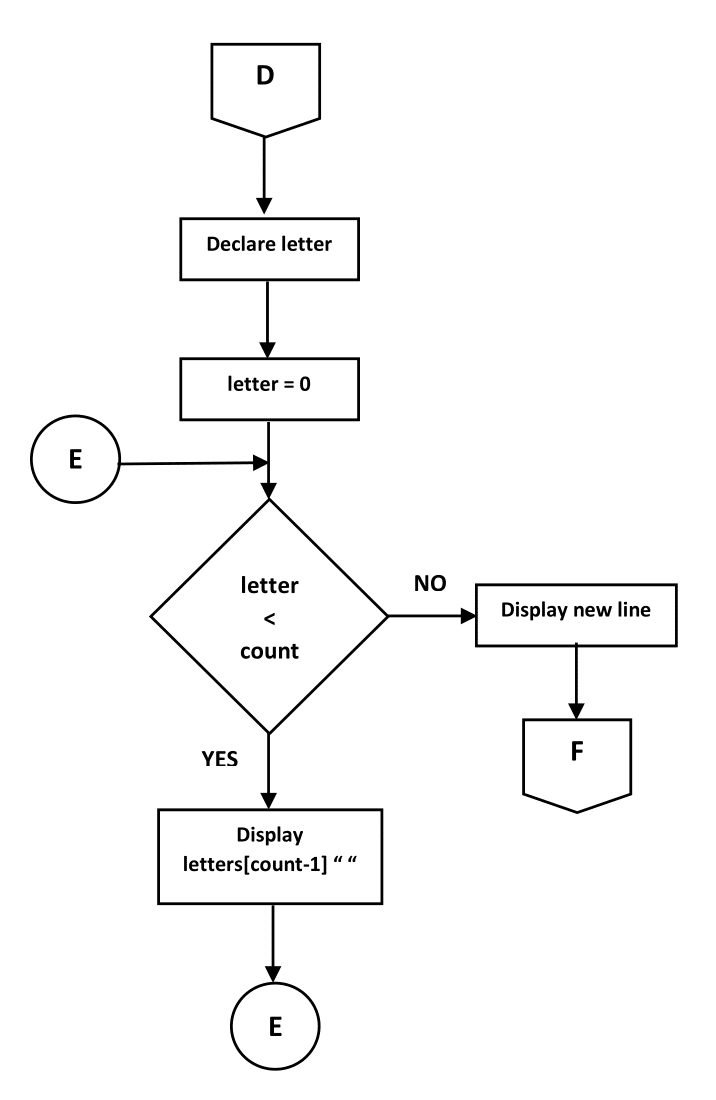


* 1. Sample Input 3



* 1. Flowchart





* 1. Pseudocode

DECLARE height, count, letter, letters[10]

SET letters AS {A,B,C,D,E,F,G,H,I,J}

INPUT height

WHILE IsInvalid(height) DO

OUTPUT “Invalid input.”

INPUT height

LOOP

IF height <1 OR height >10 THEN

OUTPUT “Minimum height is 1. Maximum height is 10.”

ELSE

SET count AS 1

WHILE count<=height DO

DECLARE space

SET space AS height - count

WHILE space > 0 DO

OUTPUT “ ”

SET space AS space - 1

LOOP

SET letter AS 0

WHILE letter < count DO

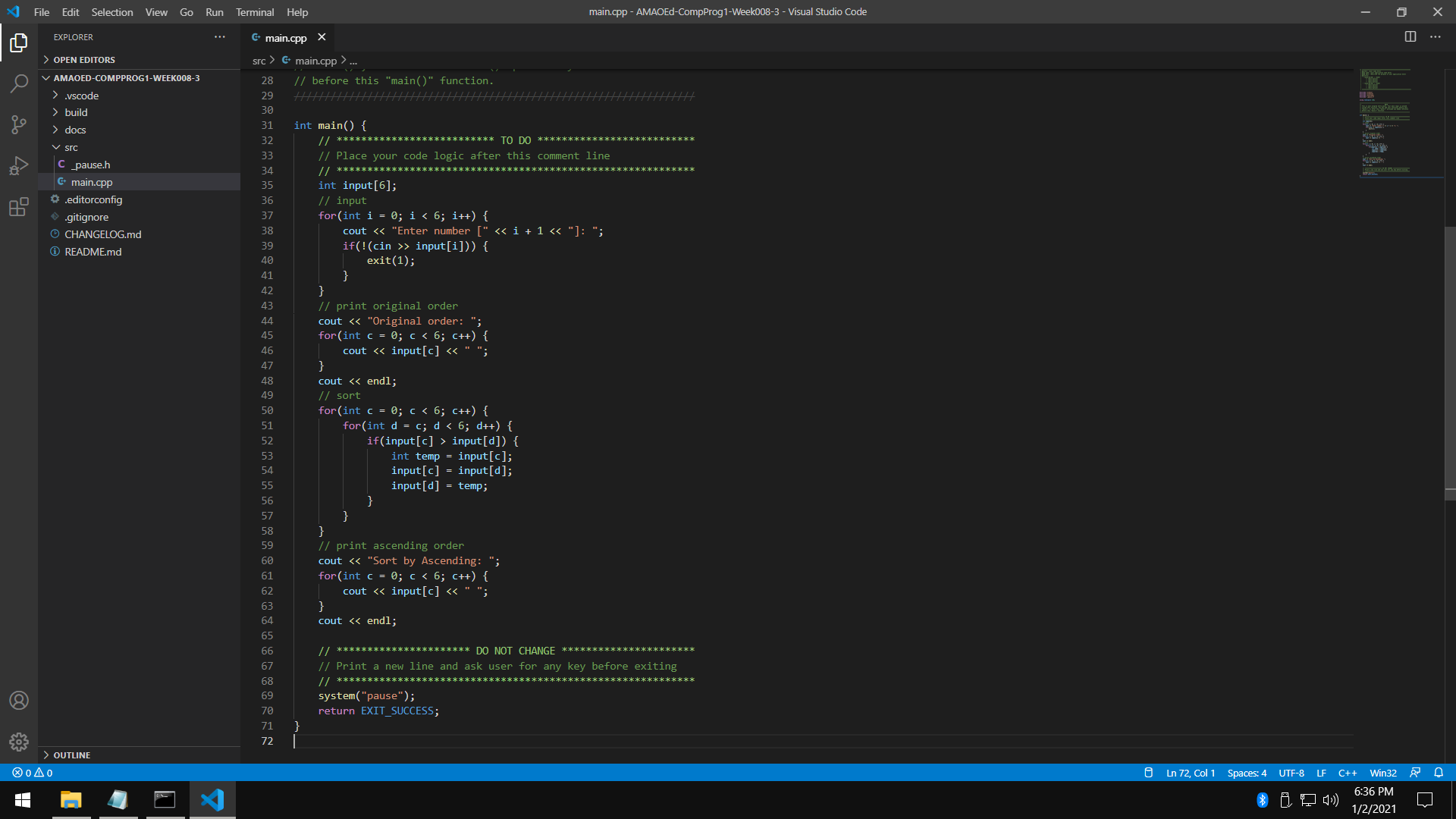
OUTPUT “letters[count - 1] “

LOOP

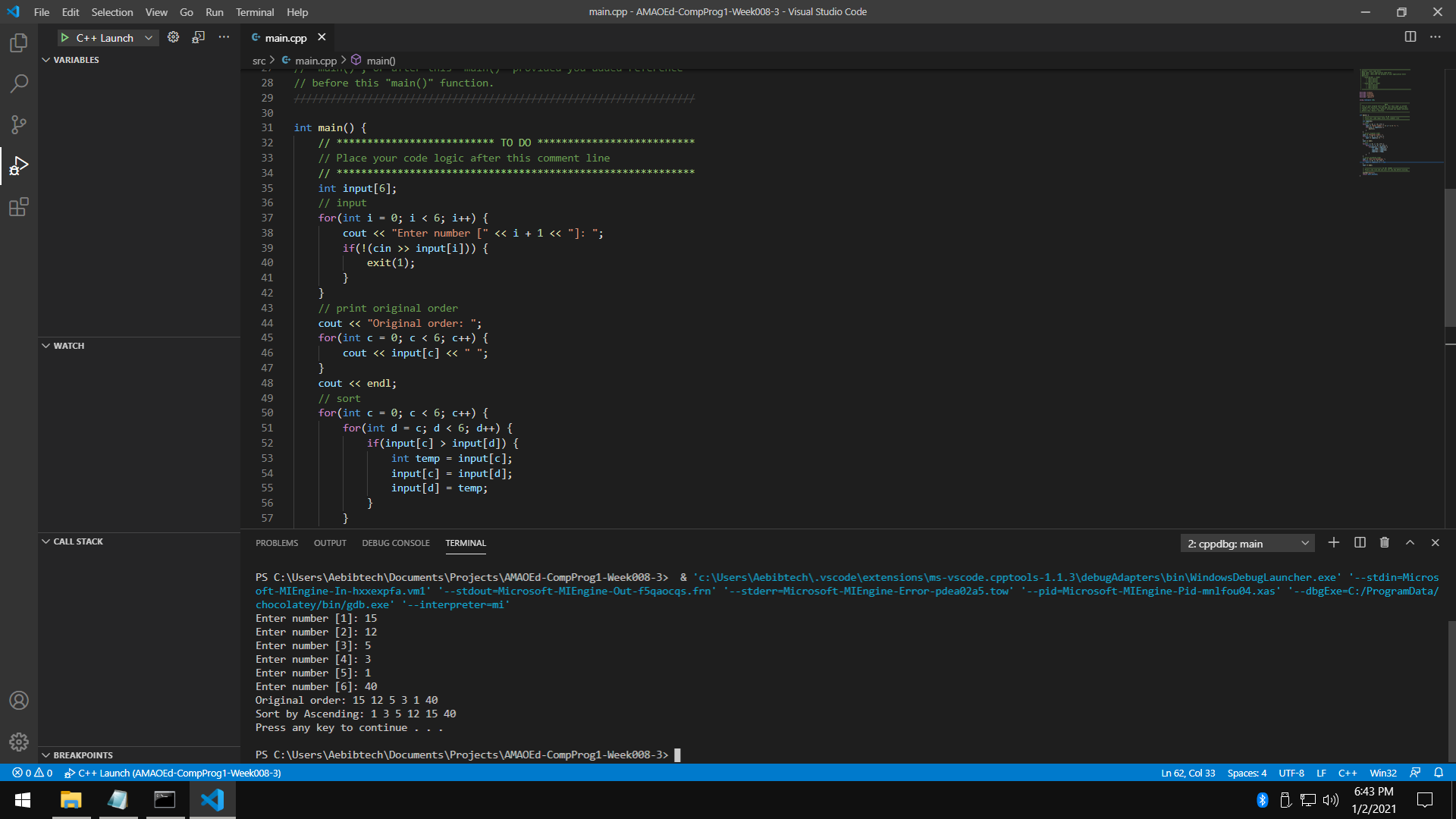
OUTPUT “\n”

LOOP

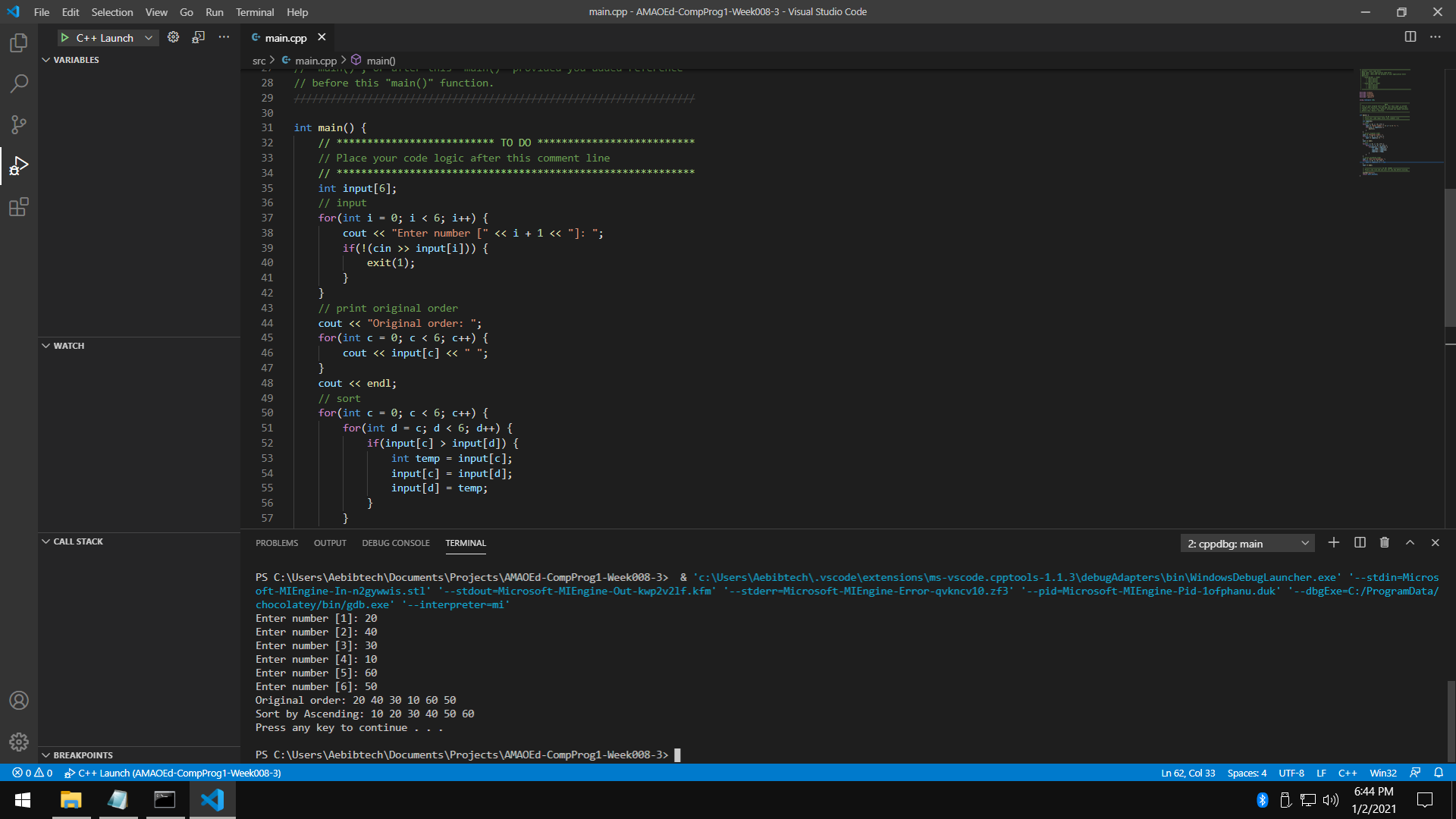
1. Write a program that will ascendingly sort six (6) integers from the user. Use only four (4) variables (including the array).
   1. Code



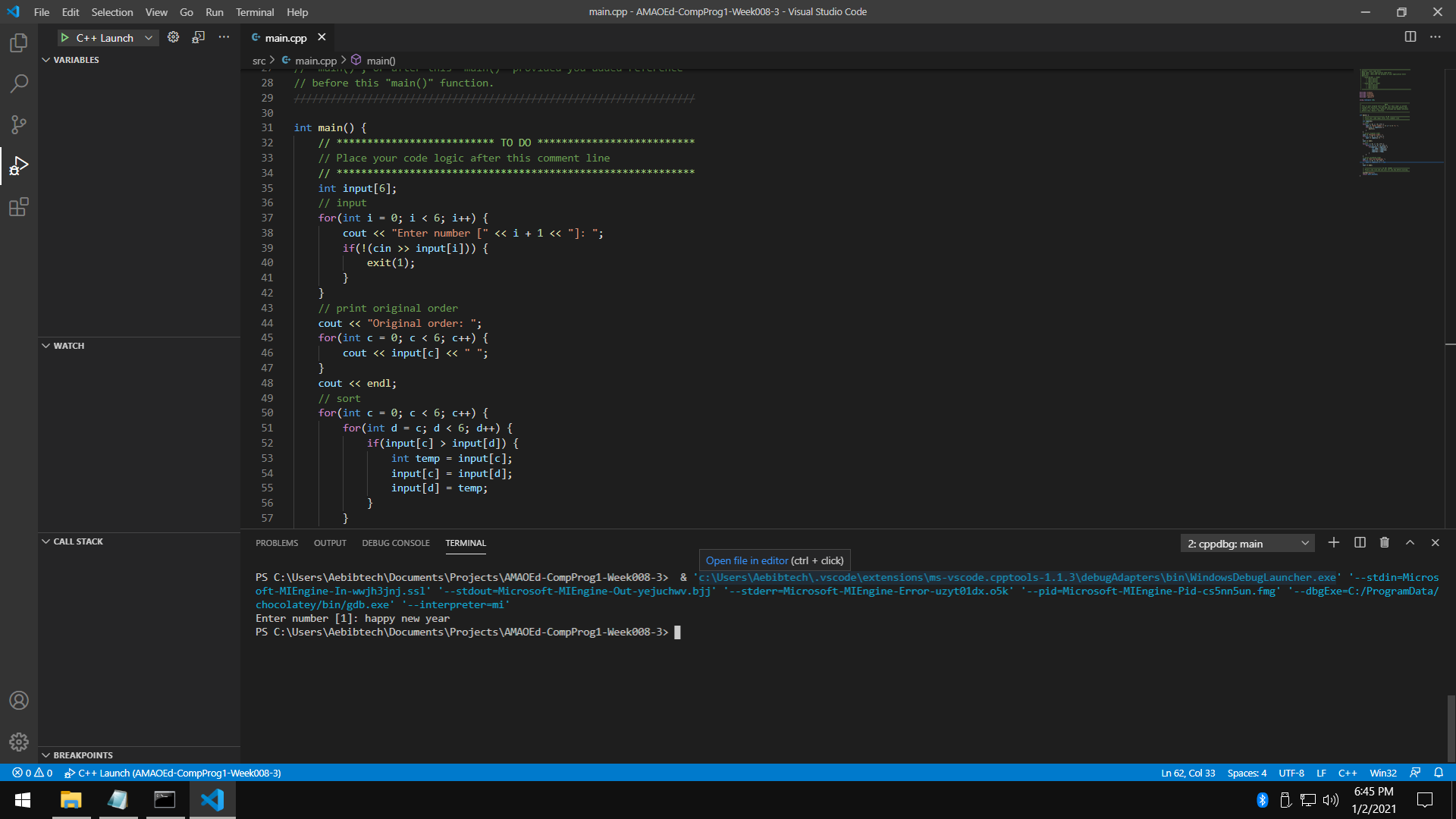
* 1. Sample Input 1



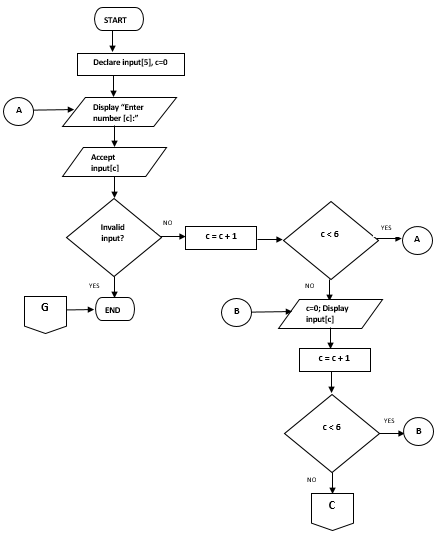
* 1. Sample Input 2

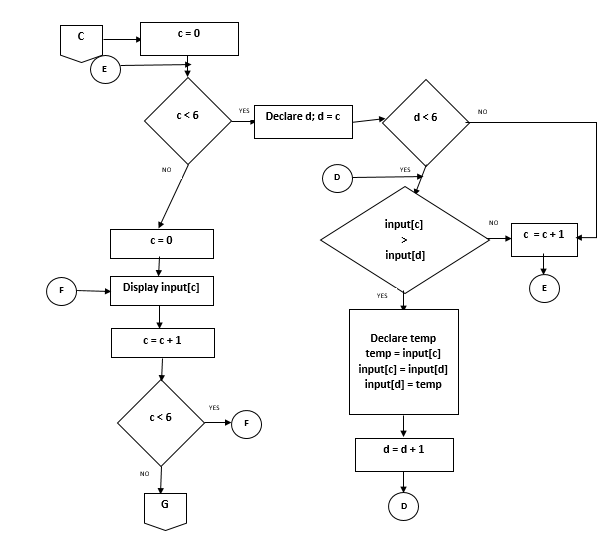


* 1. Sample Input 3



* 1. Flowchart





* 1. Pseudocode

DECLARE input[], i, c

SET i AS 0

WHILE i < 6 DO

INPUT input[i]

IF IsInvalid(input[i]) THEN

CALL Exit()

END

SET i AS i + 1

LOOP

SET c AS 0

WHILE c < 6 DO

OUTPUT input[i]

SET c AS c + 1

LOOP

SET c AS 0

WHILE c < 6 DO

DECLARE d

SET d AS c

WHILE d < 6 DO

IF input[c] > input[d] THEN

DECLARE temp

SET temp AS input[c]

SET input[c] AS input[d]

SET input[d] AS temp

END

LOOP

LOOP

1. What can you conclude from this activity?

I can conclude from this activity that loops can be used to access and modify array elements. Programmers must always keep in mind that an array’s first element has an index of 0 to avoid confusion when accessing or modifying array elements through loops.