

2331962 Olcay Sivas

by Olcay Sivas

Submission date: 03-Dec-2025 11:48AM (UTC+0000)

Submission ID: 269432286

File name: 54350_Olcay_Sivas_2331962_Olcay_Sivas_703116_235400879.docx (185.47K)

Word count: 849

Character count: 4752

Name: Olcay Sivas

Student Number: 2331962

WORKSHEET 2 – TESTING (Tiles)

1. Original code:

[Copy and paste your original code from Moodle into this box. Do not change it.]

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <string.h>
#include <math.h>

void main()
{
    float wheight, wwidth, theight, twidth, ftotal;
    int total;

    printf("What is the height of your wall: ");
    scanf("%f", &wheight);
    while (wheight <= 0)
    {
        printf("Height must be a positive number:\n ");
        scanf("%f", &wheight);
    }

    printf("What is the width of your wall: ");
    scanf("%f", &wwidth);
    while (wwidth <= 0)
    {
        printf("Width must be a positive number:\n ");
        scanf("%f", &wwidth);
    }

    theight = ceil(wheight / 20);
    twidth = ceil(wwidth / 20);
    ftotal = theight * twidth;

    printf("You need %2f tiles", ftotal);
}
```


2. Code Testing

For each test below (i – vii), run your code and enter the values provided. Provide a screenshot of the of what happened when the programme ran. Did the programme behave correctly or did it do something unexpected, and if so, why?

Input:

- i. Enter dimensions: Width 40 Length 45
- ii. Enter dimensions: Width 100000 Length 200000
- iii. Enter dimensions: Width -16 Length -20

Logic:

- iv. Enter dimensions: Width 80 Length 80. The result should be 16 tiles
- v. Enter dimensions: Width 40 Length 45. The result should be 6 tiles
- vi. Enter dimensions: Width 85 Length 90. The result should be 25 tiles.
- vii. Enter dimensions: Width 15 Length 15. The result should be 1 tile.

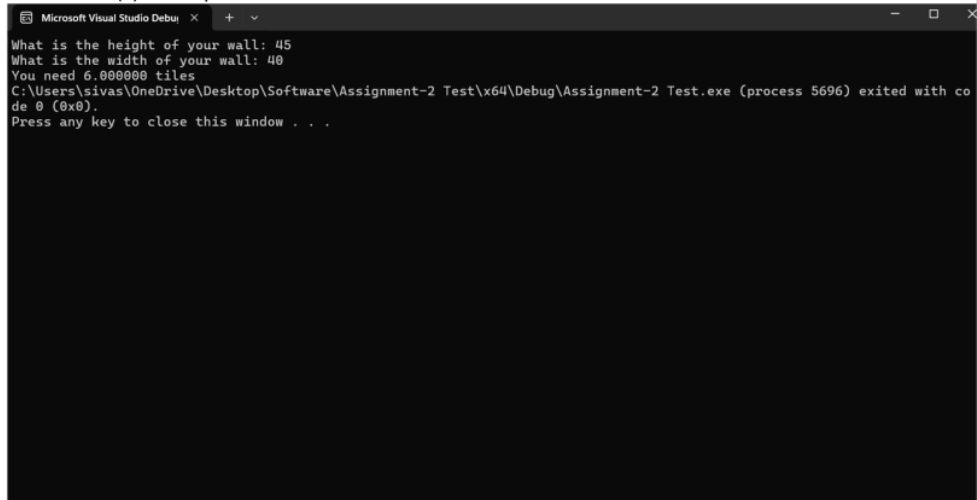
Note: If you decide to extend your program to allow a tile to be cut such that 2 pieces can be used from the same tile, your results may differ from the above. In this case, explain your logic as part of your answer)

Fill in a box for each test above. You should present 7 separate boxes, 1 for each test. (Note: the first box is provided as a template)

Test i:

Input - **Enter dimensions: width 40 length 45**

Screenshot(s) of output:



```
Microsoft Visual Studio Debug  x  +  v  -  □  x
What is the height of your wall: 45
What is the width of your wall: 40
You need 6.000000 tiles
C:\Users\sivas\OneDrive\Desktop\Software\Assignment-2 Test\x64\Debug\Assignment-2 Test.exe (process 5696) exited with code 0 (0x0).
Press any key to close this window . . .
```

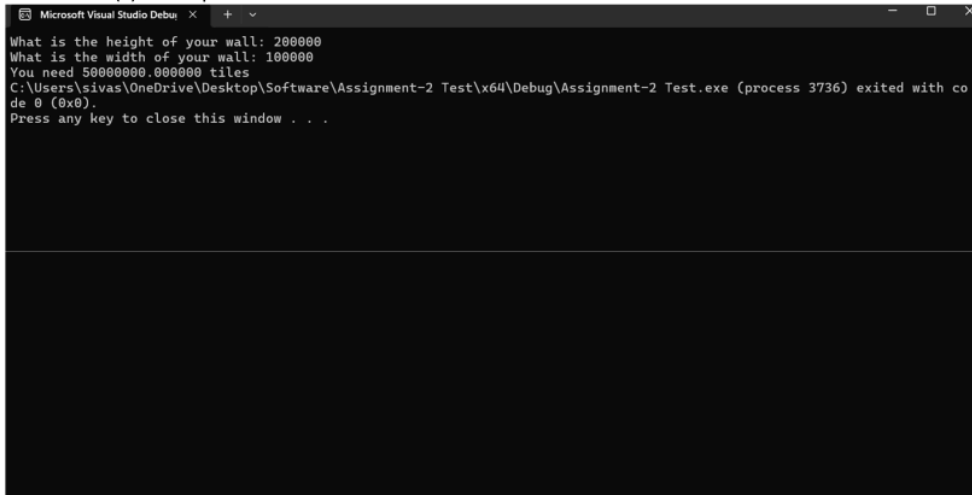
Result: Did the programme behave as expected ie syntactically and logically correct?
Yes, the tiles were calculated by the program correctly and given the result of 6.

Continue the rest of the tests below. (Copy and paste a box for each one)

Test: ii

Enter dimensions: Width 100000 Length 200000

Screenshot(s) of output:

A screenshot of the Microsoft Visual Studio Debug console window. The window title is "Microsoft Visual Studio Debug". The output text is as follows:

```
What is the height of your wall: 200000
What is the width of your wall: 100000
You need 50000000.000000 tiles
C:\Users\sivas\OneDrive\Desktop\Software\Assignment-2 Test\x64\Debug\Assignment-2 Test.exe (process 3736) exited with code 0 (0x0).
Press any key to close this window . . .
```

Result: Did the programme behave as expected ie syntactically and logically correct?

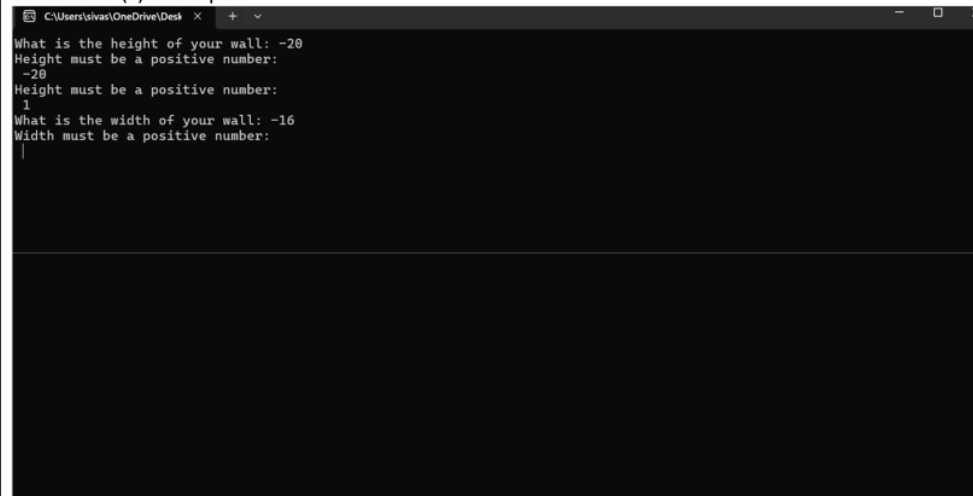
Yes, given amount been accepted and calculated by the program and the big number of 50.000.000 tiles been calculated. Even though it's not a realistic number it worked in function.

No, realistically thinking the amount of 50.000.000 tiles shouldn't have been calculated because the program is set to calculate amount of tiles a homeowner would've needed.

Test: iii

Enter dimensions: Width -16 Length -20

Screenshot(s) of output:



```
C:\Users\sivas\OneDrive\Desktop > .\Test.exe
What is the height of your wall: -20
Height must be a positive number:
-20
Height must be a positive number:
1
What is the width of your wall: -16
Width must be a positive number:
|
```

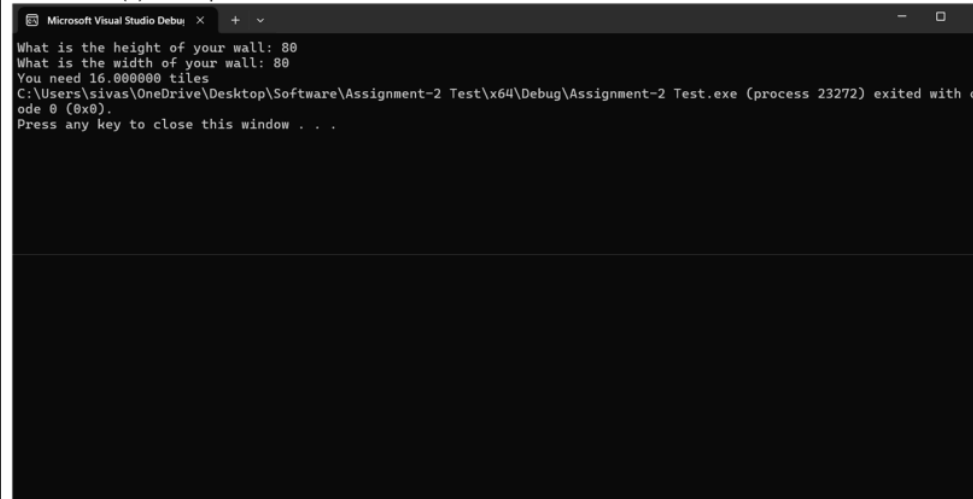
Result: Did the programme behave as expected ie syntactically and logically correct?

Yes, the negative values were rejected correctly by program and user been asked to put valid (positive) inputs.

Test: iv

Enter dimensions: Width 80 Length 80. The result should be 16 tiles

Screenshot(s) of output:

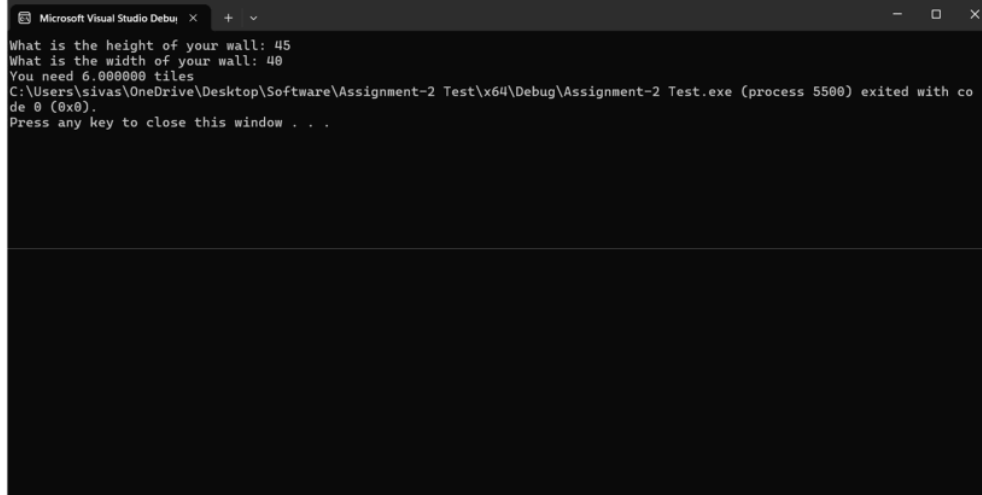


```
Microsoft Visual Studio Debug Console > .\Test.exe
What is the height of your wall: 80
What is the width of your wall: 80
You need 16.000000 tiles
C:\Users\sivas\OneDrive\Desktop\Software\Assignment-2 Test\Debug\Assignment-2 Test.exe (process 23272) exited with code 0 (0x0).
Press any key to close this window . . .
```

Result: Did the programme behave as expected ie syntactically and logically correct?
Yes, the tiles been calculated correctly by program and given the result of 16.

Test: v
Enter dimensions: Width 40 Length 45. The result should be 6 tiles

Screenshot(s) of output:

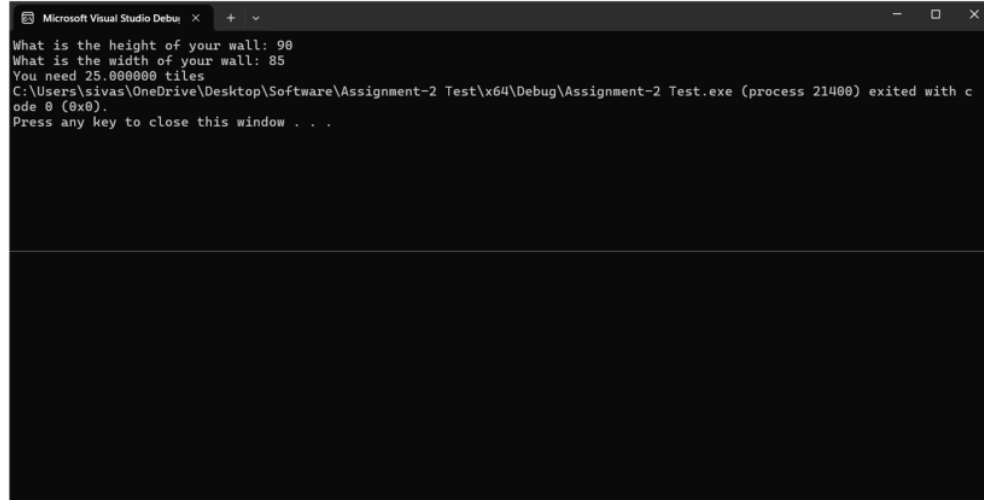


```
Microsoft Visual Studio Debug  x + -
What is the height of your wall: 45
What is the width of your wall: 40
You need 6.000000 tiles
C:\Users\sivas\OneDrive\Desktop\Software\Assignment-2 Test\x64\Debug\Assignment-2 Test.exe (process 5500) exited with code 0 (0x0).
Press any key to close this window . . .
```

Result: Did the programme behave as expected ie syntactically and logically correct?
Yes, the tiles been calculated correctly and given the result of 6. Successfully rounded up the rows (2.25 to 3).

Test: vi
Enter dimensions: Width 85 Length 90. The result should be 25 tiles.

Screenshot(s) of output:



```
Microsoft Visual Studio Debu  x  +  v  -  □  x
What is the height of your wall: 90
What is the width of your wall: 85
You need 25.000000 tiles
C:\Users\sivas\OneDrive\Desktop\Software\Assignment-2 Test\x64\Debug\Assignment-2 Test.exe (process 21400) exited with c
ode 0 (0x0).
Press any key to close this window . . .
```

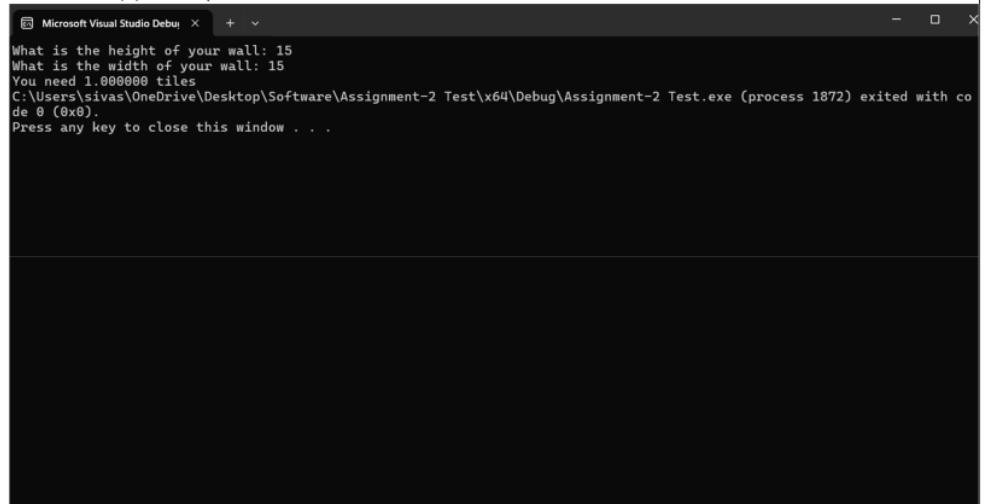
Result: Did the programme behave as expected ie syntactically and logically correct?

Yes, the tiles were calculated correctly by the program and both dimensions rounded up (4.25 rounded up to 5 and 4.5 rounded up to 5).

Test: vii

Enter dimensions: Width 15 Length 15. The result should be 1 tile

Screenshot(s) of output:



```
Microsoft Visual Studio Debu  x  +  v  -  □  x
What is the height of your wall: 15
What is the width of your wall: 15
You need 1.000000 tiles
C:\Users\sivas\OneDrive\Desktop\Software\Assignment-2 Test\x64\Debug\Assignment-2 Test.exe (process 1872) exited with co
de 0 (0x0).
Press any key to close this window . . .
```

Result: Did the programme behave as expected ie syntactically and logically correct?

Yes, the tiles were calculated correctly and even though 15 is smaller than 20 the correct answer of 1 tile been given.

3. Consider the results of the above tests and revisit your code in order to solve any problems discovered at the testing phase. (Note: you may not be able to solve all of them, but solve as many as you can.)

Paste your revised code here:

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <string.h>
#include <math.h>

void main()
{
    float wheight, wwidth, theight, twidth, ftotal;
    int total;

    printf("What is the height of your wall: ");
    scanf("%f", &wheight);
    while (wheight <= 0 || wheight >= 10000)
    {
        printf("Height must be a positive realistic number:\n ");
        scanf("%f", &wheight);
    }

    printf("What is the width of your wall: ");
    scanf("%f", &wwidth);
    while (wwidth <= 0 || wwidth >= 10000)
    {
        printf("Width must be a positive and realistic number:\n ");
        scanf("%f", &wwidth);
    }

    theight = ceil(wheight / 20);
    twidth = ceil(wwidth / 20);
    ftotal = theight * twidth;

    printf("\n-----\n");
```

```
printf("TILING ESTIMATE\n");
printf("-----\n");
printf("Wall Dimensions: %.0fcm x %.0fcm\n", wheight, wwidth);
printf("Tiles required for height: %.0f\n", theight);
printf("Tiles required for width: %.0f\n", twidth);
printf("-----\n");
printf("Total Tiles Required: %.0f\n", ftotal);
printf("-----\n");
}
```

FINAL GRADE

GENERAL COMMENTS

WORKSHEET 2

9/0

Coding- Good algorithm You could also consider that where there is less than half a tile in the final calculation, the other half could be utilised (as the pattern on this particular tile can be used in any direction) and this would save a tile.

Good validation

4/5

Testing - Good discussion, you have considered the results well. 5/5

Overall - 9/10

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

