

2331962 Olcay Sivas

by Olcay Sivas

Submission date: 09-Jan-2026 01:17PM (UTC+0000)

Submission ID: 272674678

File name: 54350_Olcay_Sivas_2331962_Olcay_Sivas_703116_921122781.docx (3.37M)

Word count: 1748

Character count: 8335

Name: Olcay Sivas

Student Number: 2331962

WORKSHEET 4 – TESTING (Dice Duel)

1. Original code:

```
2. #define _CRT_SECURE_NO_WARNINGS
3. #include <stdio.h>
4. #include <string.h>
5. #include <math.h>
6. #include <time.h>
7. #include <stdlib.h>
8.
9. void main()
10. {
11.     int myNo1, myNo2, myNo3, comp1, comp2, comp3, myTotal, compTotal;
12.     char choice;
13.     int myWin, compWin, targetWin, score, round;
14.
15.     round = 1;
16.     myWin = 0;
17.     compWin = 0;
18.     targetWin = 5;
19.
20.     printf("Are you ready to roll your dice? (y/n)");
21.     scanf(" %c", &choice);
22.
23.
24.     int r1Me = 0, r2Me = 0, r3Me = 0, r4Me = 0, r5Me = 0;
25.     int r1Co = 0, r2Co = 0, r3Co = 0, r4Co = 0, r5Co = 0;
26.
27.
28.
29.
30.     srand(time(0));
31.
32.     while (choice == 'y')
33.     {
34.         myNo1 = rand() % 6 + 1;
35.         myNo2 = rand() % 6 + 1;
36.         myNo3 = rand() % 6 + 1;
37.         myTotal = myNo1 + myNo2 + myNo3;
38.
```

```

39.         printf("You rolled...\n");
40.         printf("%d %d %d\n", myNo1, myNo2, myNo3);
41.         printf("Now the computer will roll...\n");
42.
43.         comp1 = rand() % 6 + 1;
44.         comp2 = rand() % 6 + 1;
45.         comp3 = rand() % 6 + 1;
46.         compTotal = comp1 + comp2 + comp3;
47.
48.         printf("%d %d %d\n", comp1, comp2, comp2);
49.
50.         printf("You rolled %d and computer rolled %d\n", myTotal, compTotal);
51.         if (compTotal > myTotal)
52.         {
53.             printf("The computer won.\n");
54.             compWin = compWin + 1;
55.         }
56.         else if (myTotal > compTotal)
57.         {
58.             printf("You won\n");
59.             myWin = myWin + 1;
60.         }
61.         else
62.         {
63.             printf("Draw\n");
64.         }
65.         if (round == 1)
66.         {
67.             r1Me = myTotal;
68.             r1Co = compTotal;
69.         }
70.         else if (round == 2)
71.         {
72.             r2Me = myTotal;
73.             r2Co = compTotal;
74.         }
75.         else if (round == 3)
76.         {
77.             r3Me = myTotal;
78.             r3Co = compTotal;
79.         }
80.         else if (round == 4)
81.         {
82.             r4Me = myTotal;
83.             r4Co = compTotal;
84.         }
85.         else if (round == 5)
86.         {

```

```

87.             r5Me = myTotal;
88.             r5Co = compTotal;
89.         }
90.         printf("Current scores are you %d computer %d\n", myWin, compWin);
91.
92.
93.
94.
95.
96.         if (myWin == 5)
97.         {
98.             printf("Congratulations, you won.\n");
99.             choice == 'n';
100.             break;
101.         }
102.         else if (compWin == 5)
103.         {
104.             printf("You lost.\n");
105.             choice == 'n';
106.             break;
107.         }
108.
109.
110.         else
111.         {
112.             printf("Do you wanna go again?(y/n)\n");
113.             scanf(" %c", &choice);
114.         }
115.     }
116.     printf("Round stats: \n");
117.     printf("Round 1: ");
118.     if (r1Me > r1Co)
119.     {
120.         printf("was won by you\n");
121.     }
122.     else if (r1Co > r1Me)
123.     {
124.         printf("was won by computer\n");
125.     }
126.     else
127.     {
128.         printf("was a draw\n");
129.     }
130.
131.     printf("Round 2: ");
132.     if (r2Me > r2Co)
133.     {
134.         printf("was won by you\n");

```

```

135.         }
136.         else if (r2Co > r2Me)
137.         {
138.             printf("was won by computer\n");
139.         }
140.         else
141.         {
142.             printf("was a draw\n");
143.         }
144.
145.         printf("Round 3: ");
146.         if (r3Me > r3Co)
147.         {
148.             printf("was won by you\n");
149.         }
150.         else if (r3Co > r3Me)
151.         {
152.             printf("was won by computer\n");
153.         }
154.         else
155.         {
156.             printf("was a draw\n");
157.         }
158.
159.         printf("Round 4: ");
160.         if (r4Me > r4Co)
161.         {
162.             printf("was won by you\n");
163.         }
164.         else if (r4Co > r4Me)
165.         {
166.             printf("was won by computer\n");
167.         }
168.         else
169.         {
170.             printf("was a draw\n");
171.         }
172.
173.         printf("Round 5: ");
174.         if (r5Me > r5Co)
175.         {
176.             printf("was won by you\n");
177.         }
178.         else if (r5Co > r5Me)
179.         {
180.             printf("was won by computer\n");
181.         }
182.         else

```

```
183.         {  
184.             printf("was a draw\n");  
185.         }  
186.  
187.     }
```

188. Code Testing

For each test below (i – ix), run your code and show the results of the scenarios below. Present a screenshot of what happened when the programme ran. Did the programme behave correctly or did it do something unexpected, and if so, why?

Part a)

- i. Computer wins round
- ii. User wins round
- iii. Draw
- iv. "Go again" – enter *y*
- v. "Go again" – enter *n*
- vi. "Go again" – enter *Y*
- vii. "Go again" – enter *b*

Part b)

- viii. Show score/score tracking (depending on how you have implemented your program)
- ix. Show an overall win (however this is achieved eg first to 5, best of 12 etc)

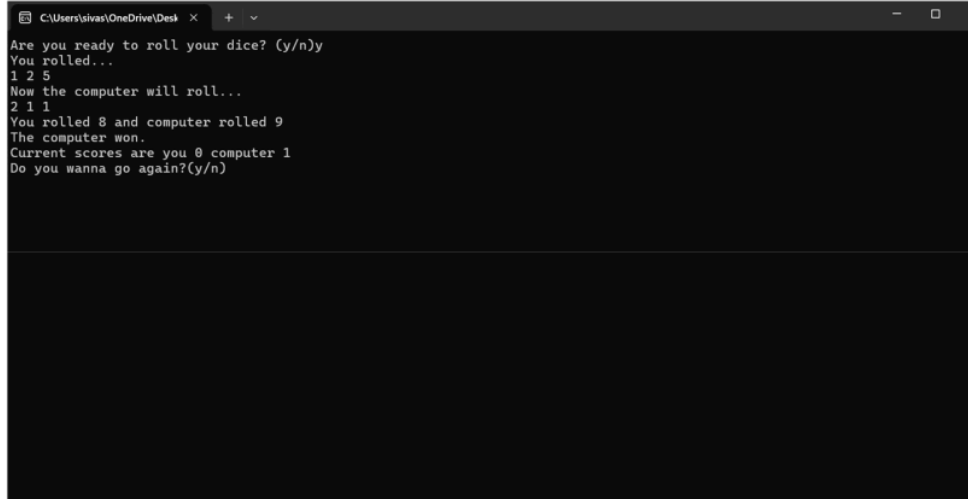
Part c)

x. Analyse your code and write a short paragraph commenting on the efficiency of the original code. Could efficiency be improved and if so comment on what is currently inefficient?

Template for tests:

i. Test: i. Computer wins round

Screenshot(s) of output:

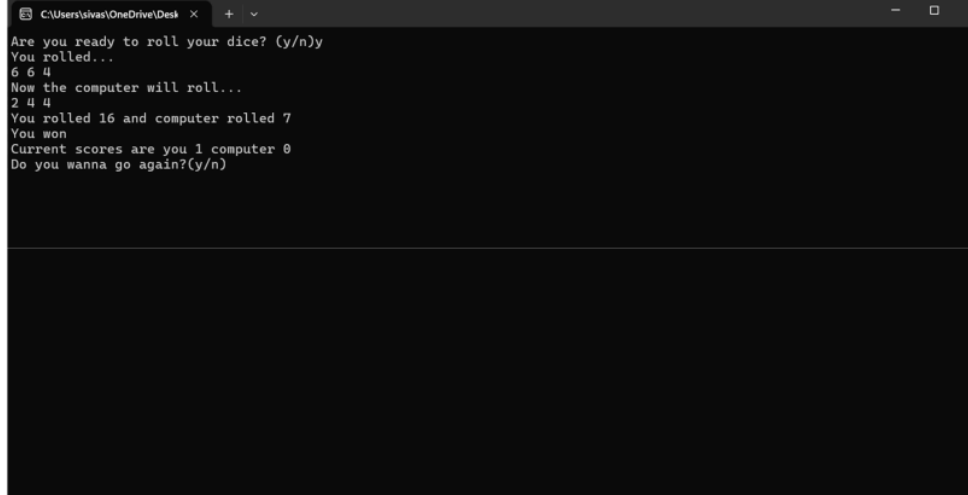


```
C:\Users\slivas\OneDrive\Desktop > python dice.py
Are you ready to roll your dice? (y/n)y
You rolled...
1 2 5
Now the computer will roll...
2 1 1
You rolled 8 and computer rolled 9
The computer won.
Current scores are you 0 computer 1
Do you wanna go again?(y/n)
```

Result: Did the programme behave as expected ie syntactically and logically correct?
Yes, the program correctly identified that the computer won and added it to the score.

i. Test: ii. User wins round

Screenshot(s) of output:

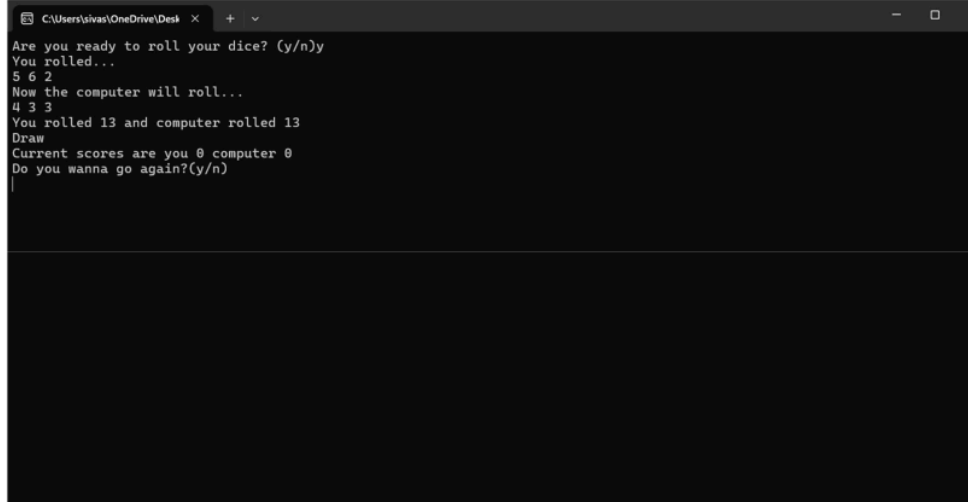


```
C:\Users\sivas\OneDrive\Desktop >
Are you ready to roll your dice? (y/n)y
You rolled...
6 6 4
Now the computer will roll...
2 4 4
You rolled 16 and computer rolled 7
You won
Current scores are you 1 computer 0
Do you wanna go again?(y/n)
```

Result: Did the programme behave as expected ie syntactically and logically correct?
Yes, the program was correctly able to tell that the user won the round over the computer and the result was added to the score.

i. Test: iii. Draw

Screenshot(s) of output:



```
C:\Users\shras\OneDrive\Desktop
Are you ready to roll your dice? (y/n)y
You rolled...
5 6 2
Now the computer will roll...
4 3 3
You rolled 13 and computer rolled 13
Draw
Current scores are you 0 computer 0
Do you wanna go again?(y/n)
```

Result: Did the programme behave as expected ie syntactically and logically correct?
Yes, the program was able to correctly identify the situation as draw.

i. Test: iv. "Go again" – enter y

Screenshot(s) of output:

```
C:\Users\sivas\OneDrive\Desktop >
Are you ready to roll your dice? (y/n)y
You rolled...
5 3 3
Now the computer will roll...
5 3 3
You rolled 11 and computer rolled 14
The computer won.
Current scores are you 0 computer 1
Do you wanna go again?(y/n)
y
You rolled...
4 3 6
Now the computer will roll...
2 5 5
You rolled 13 and computer rolled 12
You won
Current scores are you 1 computer 1
Do you wanna go again?(y/n)
|
```

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

Yes, the program correctly continued with the next dice game and the score was calculated.

i. Test: v. "Go again" – enter *n*

Screenshot(s) of output:

```
Microsoft Visual Studio Debug >
Are you ready to roll your dice? (y/n)y
You rolled...
3 2 4
Now the computer will roll...
6 4 4
You rolled 9 and computer rolled 11
The computer won.
Current scores are you 0 computer 1
Do you wanna go again?(y/n)
n
Round stats:
Round 1: was won by computer
Round 2: was a draw
Round 3: was a draw
Round 4: was a draw
Round 5: was a draw

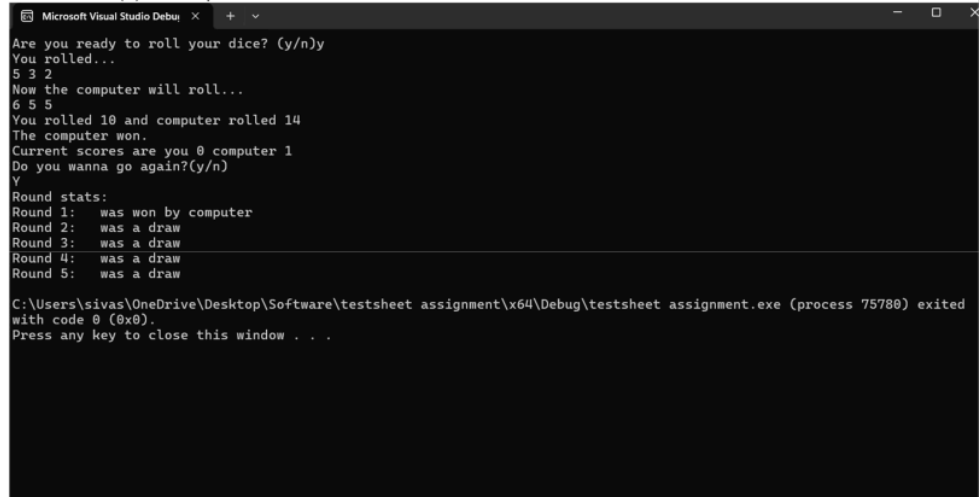
C:\Users\sivas\OneDrive\Desktop\Software\testsheet assignment\x64\Debug\testsheet assignment.exe (process 84648) exited
with code 0 (0x0).
Press any key to close this window . . .|
```

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

Yes, the program correctly stopped the loop after the 'n' input had been entered. Conversely, the results of rounds 2,3,4 and 5 were shown as a draw even though these rounds never been played.

i. Test: vi. "Go again" – enter Y

Screenshot(s) of output:



```
Microsoft Visual Studio Debu  x  +  v
Are you ready to roll your dice? (y/n)y
You rolled...
5 3 2
Now the computer will roll...
6 5 5
You rolled 10 and computer rolled 14
The computer won.
Current scores are you 0 computer 1
Do you wanna go again?(y/n)
Y
Round stats:
Round 1: was won by computer
Round 2: was a draw
Round 3: was a draw
Round 4: was a draw
Round 5: was a draw

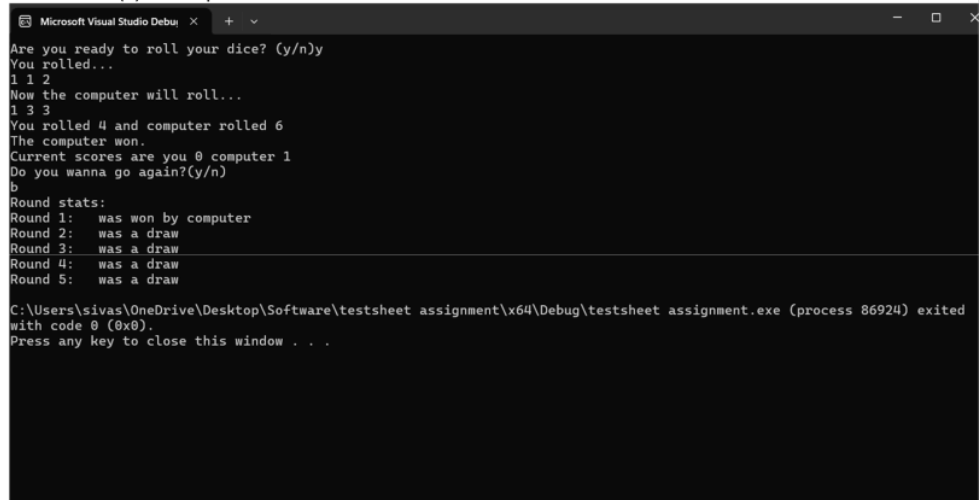
C:\Users\sivas\OneDrive\Desktop\Software\testsheet assignment\x64\Debug\testsheet assignment.exe (process 75780) exited
with code 0 (0x0).
Press any key to close this window . . .
```

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

No, the user was expecting the program to continue with the dice game but the program was closed due to an error with scanf expecting a lowercase 'y' instead.

Test: vii. "Go again" – enter b

Screenshot(s) of output:



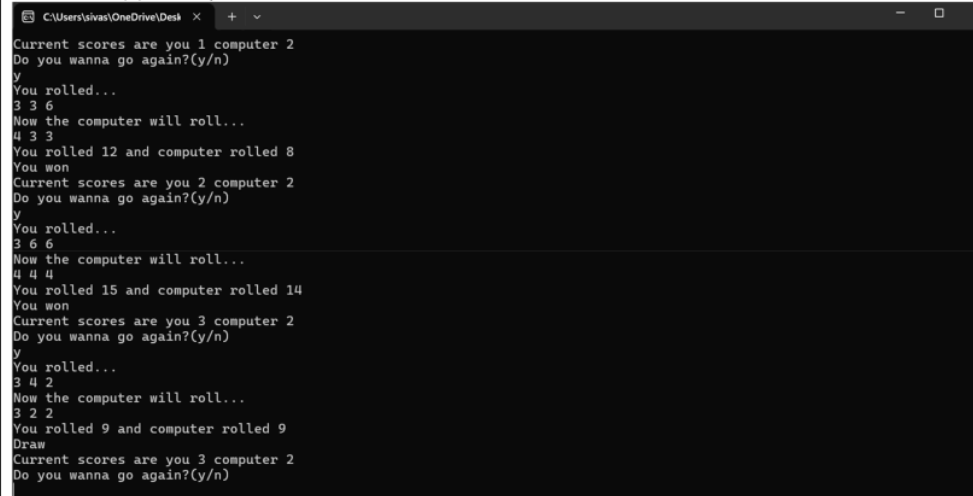
```
Microsoft Visual Studio Debu  x  +  v
Are you ready to roll your dice? (y/n)y
You rolled...
1 1 2
Now the computer will roll...
1 3 3
You rolled 4 and computer rolled 6
The computer won.
Current scores are you 0 computer 1
Do you wanna go again?(y/n)
b
Round stats:
Round 1: was won by computer
Round 2: was a draw
Round 3: was a draw
Round 4: was a draw
Round 5: was a draw

C:\Users\sivas\OneDrive\Desktop\Software\testsheet assignment\x64\Debug\testsheet assignment.exe (process 86924) 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 input 'b' was identified as incorrect by the program and loop was stopped but even though 'b' was a wrong input the remaining rounds been incorrectly captured as draw due to a scanf error.

- i. Test: viii. Show score/score tracking (depending on how you have implemented your program)

Screenshot(s) of output:



```
C:\Users\sivas\OneDrive\Desktop X + v
Current scores are you 1 computer 2
Do you wanna go again?(y/n)
y
You rolled...
3 3 6
Now the computer will roll...
4 3 3
You rolled 12 and computer rolled 8
You won
Current scores are you 2 computer 2
Do you wanna go again?(y/n)
y
You rolled...
3 6 6
Now the computer will roll...
4 4 4
You rolled 15 and computer rolled 14
You won
Current scores are you 3 computer 2
Do you wanna go again?(y/n)
y
You rolled...
3 4 2
Now the computer will roll...
3 2 2
You rolled 9 and computer rolled 9
Draw
Current scores are you 3 computer 2
Do you wanna go again?(y/n)
```

Result: Did the programme behave as expected ie syntactically and logically correct?
Yes, the program identified all inputs correctly and calculated a correct score tracking by telling the user the score.

- i. Test: ix. Show an overall win (however this is achieved eg first to 5, best of 12 etc)

Screenshot(s) of output:

```
Microsoft Visual Studio Debu x + v
Current scores are you 3 computer 4
Do you wanna go again?(y/n)
y
You rolled...
4 6 4
Now the computer will roll...
6 2 2
You rolled 14 and computer rolled 11
You won
Current scores are you 4 computer 4
Do you wanna go again?(y/n)
y
You rolled...
5 1 3
Now the computer will roll...
4 4 4
You rolled 9 and computer rolled 12
The computer won.
Current scores are you 4 computer 5
You lost.
Round stats:
Round 1: was won by computer
Round 2: was a draw
Round 3: was a draw
Round 4: was a draw
Round 5: was a draw
C:\Users\sivas\OneDrive\Desktop\Software\testsheets assignment\x64\Debug\testsheets assignment.exe (process 81828) exited
with code 0 (0x0).
Press any key to close this window . . .
```

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

No, the program logic for checking an "Overall Win" (first to 5) was working but the **Round Stats** at the end were broken. Rounds 2 to 5 incorrectly shown as draws because the round counter variable was never incremented.

Test: x. Analyse your code and write a short paragraph commenting on the efficiency of the original code. Could efficiency be improved and if so comment on what is currently inefficient?

The original code could be described as 'functional' but it was also efficient because it used the same logic for every round using and just changing the individual variables. Instead, Arrays and a Loop could be used to reduce the code length and increase efficiency. This also made the code hard to update e.g. if 10 rounds were to be used, it would require writing five more else if blocks. Similarly, there was a logic error in the original code where the round variable never incremented which meant the statistics for rounds two to five were never recorded. A typo also meant the second dice for the computer printed twice (comp2) instead of showing the third dice (comp3).

189. Consider the results of the above tests and your analysis of program efficiency and revisit your original code in order to address any tests that failed and/or improve code efficiency.

(Note: you may not be able to solve all of the problems that were encountered, but solve as many as you can.)

Paste your full revised code here :

```

#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <time.h>
#include <stdlib.h>

void main()
{
    int myNo1, myNo2, myNo3, comp1, comp2, comp3, myTotal, compTotal;
    char choice;
    int myWin = 0, compWin = 0, round=0;
    int myHistory[100];
    int compHistory[100];

    printf("Are you ready to roll your dice? (y/n) ");
    scanf(" %c", &choice);

    srand(time(0));

    while (choice == 'y' || choice == 'Y')
    {
        round++;

        myNo1 = rand() % 6 + 1;
        myNo2 = rand() % 6 + 1;
        myNo3 = rand() % 6 + 1;
        myTotal = myNo1 + myNo2 + myNo3;

        printf("\n--- Round %d ---\n", round);
        printf("You rolled: %d %d %d (Total: %d)\n", myNo1, myNo2, myNo3, myTotal);

        comp1 = rand() % 6 + 1;
        comp2 = rand() % 6 + 1;
        comp3 = rand() % 6 + 1;
        compTotal = comp1 + comp2 + comp3;

        printf("Computer rolled: %d %d %d (Total: %d)\n", comp1, comp2, comp3,
compTotal);

        if (round <= 100)
        {
            myHistory[round - 1] = myTotal;
            compHistory[round - 1] = compTotal;
        }

        if (compTotal > myTotal)
        {
            printf("Result: The computer won this round.\n");
            compWin++;
        }
        else if (myTotal > compTotal)
        {

```

```

        printf("Result: You won this round!\n");
        myWin++;
    }
    else
    {
        printf("Result: It's a draw.\n");
    }

    printf("Current Score -> You: %d | Computer: %d\n", myWin, compWin);

    if (myWin == 5)
    {
        printf("\nCONGRATULATIONS! You reached 5 wins first!\n");
        break;
    }
    else if (compWin == 5)
    {
        printf("\nGAME OVER. The computer reached 5 wins first.\n");
        break;
    }

    printf("\nDo you want to play the next round? (y/n): ");
    scanf(" %c", &choice);
}

printf("\n===== \n");
printf("  GAME SUMMARY   \n");
printf("===== \n");

for (int i = 0; i < round; i++)
{
    printf("Round %d: ", i + 1);
    if (myHistory[i] > compHistory[i])
    {
        printf("Won by YOU\n");
    }
    else if (compHistory[i] > myHistory[i])
    {
        printf("Won by COMPUTER\n");
    }
    else
    {
        printf("DRAW\n");
    }
}
printf("===== \n");
}

```

FINAL GRADE

17/0

GENERAL COMMENTS

WORKSHEET 4

Coding- The initial coding is lengthy due to lack of array (you've essentially unpacked a loop!) so it's good to see that things have improved in the final code by bringing in an array to hold the scores.You've also improved the exit from the game (so it's not showing "was a draw" when the round wasn't played)

You could also consider holding the user numbers and the computer numbers in an array each too, to streamline things. Good solution overall.

8/10

Testing - Good discussion overall. Occasionally a bit difficult to follow the explanation. 9/10

Overall- 17/20

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15
