

Worksheet 4 - Jake Richards

by Jake Richards

Submission date: 09-Jan-2026 02:32AM (UTC+0000)

Submission ID: 272622846

File name: 64894_Jake_Richards_Worksheet_4_-_Jake_Richards_703116_124581723.docx (420.9K)

Word count: 1419

Character count: 7133

Name: _____

Student Number: _____

WORKSHEET 4 – TESTING (Dice Duel)

1. Original code:

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

```
#define _CRT_SECURE_NO_WARNINGS
#include <math.h>
#include <windows.h>
#include <stdio.h>
#include <time.h>

void main()
{
    int high = 6, low = 1;
    int dice1, dice2, dice3;
    int compdice1, compdice2, compdice3;
    int diceTotal, compTotal;
    int diceWins = 0, compWins = 0, roundNo = 0;
    int roundStats[24]; // draw = 1, dice = 2, comp = 3

    srand(time(NULL));

    char choice;
    printf("Welcome to Best of 12 Dice Roll!\n");
    printf("Are you ready to roll your dice? (Y/N) ");
    scanf(" %c", &choice);

    while (toupper(choice) == 'Y')
    {
        dice1 = rand() % (high - low + 1) + low;
        dice2 = rand() % (high - low + 1) + low;
        dice3 = rand() % (high - low + 1) + low;
        compdice1 = rand() % (high - low + 1) + low;
        compdice2 = rand() % (high - low + 1) + low;
        compdice3 = rand() % (high - low + 1) + low;

        diceTotal = dice1 + dice2 + dice3;
        compTotal = compdice1 + compdice2 + compdice3;

        printf("You rolled....\n");
        Sleep(1000);
        printf("%d %d %d", dice1, dice2, dice3);
```

```

printf("\n\nThe computer rolled....\n");
Sleep(1000);
printf("%d %d %d", compdice1, compdice2, compdice3);

printf("\n\nYour total is %d....", diceTotal);
printf("\n\nThe computer's total is %d....", compTotal);

if (diceTotal > compTotal)
{
    printf("\n\nCongratulations! You won!");
    diceWins = diceWins + 1;
    roundStats[roundNo] = 2;
}
else if (diceTotal < compTotal)
{
    printf("\n\nThe computer won!");
    compWins = compWins + 1;
    roundStats[roundNo] = 3;
}
else if (diceTotal == compTotal)
{
    printf("\n\nIt's a draw!");
    roundStats[roundNo] = 1;
}

printf("\n\nYour score is: %d. The computer's score is: %d", diceWins,
compWins);

if (compWins == 12)
{
    printf("\n\nThe computer has won!");
    break;
}
else if (diceWins == 12)
{
    printf("\n\nYou have won!");
    break;
}

roundNo = roundNo + 1;
printf("\n\nCurrent Statistics:");
for (int i = 0; i <= roundNo; i++)
{
    if (roundStats[i] == 1)
    {
        printf("\nRound %d: You drew.", i + 1);
    }
    else if (roundStats[i] == 2)
    {
        printf("\nRound %d: You won!", i + 1);
    }
}

```

```

        else if (roundStats[i] == 3)
        {
            printf("\nRound %d: The computer won!", i + 1);
        }
    }
    printf("\n\nAre you ready to roll your dice again? (Y/N) ");
    scanf(" %c", &choice);
}
printf("\n\nThank you for playing Best of 12 Dice Roll!");
printf("\n\nRound Statistics:");
for (int i = 0; i < roundNo; i++)
{
    if (roundStats[i] == 1)
    {
        printf("\nRound %d: You drew.", i + 1);
    }
    else if (roundStats[i] == 2)
    {
        printf("\nRound %d: You won!", i + 1);
    }
    else if (roundStats[i] == 3)
    {
        printf("\nRound %d: The computer won!", i + 1);
    }
}
}

```

2. 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*

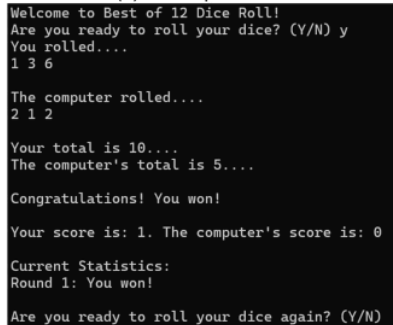
Part b)

- 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:

Test: ii. User wins round
Screenshot(s) of output: 
Result: Did the programme behave as expected ie syntactically and logically correct? YES Responded as appropriate to when the user won the round. Also displayed a running statistic of each round.

(Copy and paste box above for each of the rest of the tests)

Test: i. Computer wins round

Screenshot(s) of output:

```
Are you ready to roll your dice again? (Y/N) y
You rolled....
6 2 1

The computer rolled....
1 3 6

Your total is 9....
The computer's total is 10....

The computer won!

Your score is: 1. The computer's score is: 1

Current Statistics:
Round 1: You won!
Round 2: The computer won!

Are you ready to roll your dice again? (Y/N) |
```

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

YES

Responded as appropriate to when the computer won the round. Also displayed a running statistic of each round.

Test: iii. Draw

Screenshot(s) of output:

```
Are you ready to roll your dice again? (Y/N) y
You rolled....
1 3 4

The computer rolled....
4 2 2

Your total is 8....
The computer's total is 8....

It's a draw!

Your score is: 6. The computer's score is: 2

Current Statistics:
Round 1: You won!
Round 2: The computer won!
Round 3: You won!
Round 4: You won!
Round 5: You won!
Round 6: You won!
Round 7: You won!
Round 8: The computer won!
Round 9: You drew.

Are you ready to roll your dice again? (Y/N) y
```

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

YES

Responded as appropriate to when both drew the same total. Also displayed a running statistic of each round.

Test: iv. "Go again" – enter y

Screenshot(s) of output:

```
Are you ready to roll your dice again? (Y/N) y
You rolled....
5 4 4

The computer rolled....
4 4 6

Your total is 13....
The computer's total is 14....

The computer won!

Your score is: 6. The computer's score is: 3

Current Statistics:
Round 1: You won!
Round 2: The computer won!
Round 3: You won!
Round 4: You won!
Round 5: You won!
Round 6: You won!
Round 7: You won!
Round 8: The computer won!
Round 9: You drew.
Round 10: The computer won!

Are you ready to roll your dice again? (Y/N)
```

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

YES

Redrew the dice resulting in a win for the computer.

Test: v. "Go again" – enter n

Screenshot(s) of output:

```
Are you ready to roll your dice again? (Y/N) n

Thank you for playing Best of 12 Dice Roll!

Round Statistics:
Round 1: You won!
Round 2: The computer won!
Round 3: You won!
Round 4: You won!
Round 5: You won!
Round 6: You won!
Round 7: You won!
Round 8: The computer won!
Round 9: You drew.
Round 10: The computer won!
C:\Users\Jake\Desktop\University Year 1\Software
xited with code 0 (0x0).
Press any key to close this window . . .
```

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

YES

Ended the game, presented round statistics for every round played.

Test: vi. "Go again" – enter Y

Screenshot(s) of output:

```
Are you ready to roll your dice again? (Y/N) Y
You rolled....
5 4 6

The computer rolled....
2 3 1

Your total is 15...
The computer's total is 6....

Congratulations! You won!

Your score is: 1. The computer's score is: 1

Current Statistics:
Round 1: The computer won!
Round 2: You won!

Are you ready to roll your dice again? (Y/N)
```

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

YES

Redrew the dice and played the game. User won.

Test: vii. "Go again" – enter b

Screenshot(s) of output:

```
Are you ready to roll your dice again? (Y/N) b

Thank you for playing Best of 12 Dice Roll!

Round Statistics:
Round 1: The computer won!
Round 2: You won!
C:\Users\Jake\Desktop\University Year 1\Software
xited with code 0 (0x0).
Press any key to close this window . . .|
```

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

No

Should've asked for a value of either Y/N but instead decided to end the game outright.

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

Screenshot(s) of output:

```
Your score is: 4. The computer's score is: 10

Current Statistics:
Round 1: The computer won!
Round 2: The computer won!
Round 3: You won!
Round 4: You won!
Round 5: The computer won!
Round 6: The computer won!
Round 7: The computer won!
Round 8: You won!
Round 9: The computer won!
Round 10: The computer won!
Round 11: The computer won!
Round 12: The computer won!
Round 13: You drew.
Round 14: You won!
Round 15: The computer won!
```

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

Yes

After each round, it adds on to the current score tracking, also does the same when the game ends.

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

Screenshot(s) of output:

```
Are you ready to roll your dice again? (Y/N) y
You rolled....
3 4 6

The computer rolled....
1 2 6

Your total is 13....
The computer's total is 9....

Congratulations! You won!

Your score is: 12. The computer's score is: 9
You have won!

Thank you for playing Best of 12 Dice Roll!

Round Statistics:
Round 1: You won!
Round 2: You won!
Round 3: The computer won!
Round 4: The computer won!
Round 5: You won!
Round 6: The computer won!
Round 7: You won!
Round 8: You won!
Round 9: The computer won!
Round 10: The computer won!
Round 11: You won!
Round 12: You won!
Round 13: The computer won!
Round 14: You won!
Round 15: You drew.
Round 16: You won!
Round 17: The computer won!
Round 18: The computer won!
Round 19: You won!
Round 20: The computer won!
Round 21: You won!
C:\Users\Jake\Desktop\University Year 1\Software D
Press any key to close this window . . .]
```

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

No

Final round (in this instance 22) does not appear in the final score.

3. 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 :

[Copy and paste your full revised code here – note, do not use a screenshot]

```
#define _CRT_SECURE_NO_WARNINGS
#include <math.h>
#include <windows.h>
#include <stdio.h>
#include <time.h>

void main()
{
    int high = 6, low = 1;
    int dice1, dice2, dice3;
    int compdice1, compdice2, compdice3;
    int diceTotal, compTotal;
    int diceWins = 0, compWins = 0, roundNo = 0;
    int roundStats[24]; // draw = 1, dice = 2, comp = 3

    srand(time(NULL));

    char choice;
    printf("Welcome to Best of 5 Dice Roll!\n");
    printf("Are you ready to roll your dice? (Y/N) ");
    scanf(" %c", &choice);

    while (toupper(choice) == 'Y')
    {
        dice1 = rand() % (high - low + 1) + low;
        dice2 = rand() % (high - low + 1) + low;
        dice3 = rand() % (high - low + 1) + low;
        compdice1 = rand() % (high - low + 1) + low;
        compdice2 = rand() % (high - low + 1) + low;
        compdice3 = rand() % (high - low + 1) + low;

        diceTotal = dice1 + dice2 + dice3;
        compTotal = compdice1 + compdice2 + compdice3;

        printf("You rolled...\n");
        Sleep(1000);
        printf("%d %d %d", dice1, dice2, dice3);

        printf("\n\nThe computer rolled...\n");
        Sleep(1000);
        printf("%d %d %d", compdice1, compdice2, compdice3);

        printf("\n\nYour total is %d....", diceTotal);
        printf("\n\nThe computer's total is %d....", compTotal);
    }
}
```

```

if (diceTotal > compTotal)
{
    printf("\n\nCongratulations! You won!");
    diceWins = diceWins + 1;
    roundStats[roundNo] = 2;
}
else if (diceTotal < compTotal)
{
    printf("\n\nThe computer won!");
    compWins = compWins + 1;
    roundStats[roundNo] = 3;
}
else if (diceTotal == compTotal)
{
    printf("\n\nIt's a draw!");
    roundStats[roundNo] = 1;
}

roundNo = roundNo + 1;
printf("\n\nCurrent Statistics:");
for (int i = 0; i <= roundNo; i++)
{
    if (roundStats[i] == 1)
    {
        printf("\nRound %d: You drew.", i + 1);
    }
    else if (roundStats[i] == 2)
    {
        printf("\nRound %d: You won!", i + 1);
    }
    else if (roundStats[i] == 3)
    {
        printf("\nRound %d: The computer won!", i + 1);
    }
}

printf("\n\nYour score is: %d. The computer's score is: %d", diceWins, compWins);

if (compWins == 5)
{
    printf("\n\nThe computer has won!");
    break;
}
else if (diceWins == 5)
{
    printf("\n\nYou have won!");
    break;
}

printf("\n\nAre you ready to roll your dice again? (Y/N) ");
scanf(" %c", &choice);

```

```
}
printf("\n\nThank you for playing Best of 5 Dice Roll!");
printf("\n\nRound Statistics:");
for (int i = 0; i < roundNo; i++)
{
    if (roundStats[i] == 1)
    {
        printf("\nRound %d: You drew.", i + 1);
    }
    else if (roundStats[i] == 2)
    {
        printf("\nRound %d: You won!", i + 1);
    }
    else if (roundStats[i] == 3)
    {
        printf("\nRound %d: The computer won!", i + 1);
    }
}
}
```

Worksheet 4 - Jake Richards

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

WORKSHEET 4

16/0

Coding- A good solution. Good to see use of array for the round stats. You could also have used an array for the dice rolls for each player, to streamline things, you could then include adding along with print in a for loop (for example)

It would also be good to streamline the section that tells you whether you drew, won of the computer won, as this code repeats below. Possibly a function would be handy here.

8/10

Testing - Generally a good discussion, but don't forget to say why -For exam "Final round does not appear in the final score". Why not?

8.5/10

Overall- 16.5/20

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11
