

test sheet 3

by Josh Willis

Submission date: 27-Dec-2025 11:54PM (UTC+0000)

Submission ID: 271789858

File name: 58111_Josh_Willis_test_sheet_3_703116_774809364.docx (264.22K)

Word count: 1427

Character count: 7142

Name: Joshua Willis

Student Number: 2405267

WORKSHEET 3 – TESTING (Lap Times)

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 <math.h>

void main()
{
    float enteredLaps[5];
    float finalTotal=0;
    float targetTime;
    float fast = 1000;
    float slow = 0;
    int count = 1;
    int lapCount = 2;
    int countTarget = 0;

    printf("What is your target time to achieve?          Target = ");
    scanf("%f", &targetTime);

    printf("please enter your lap times (in seconds)\n\n");

    for (int i = 0; i < 5; i++)
    {
        printf("Lap time %d :",count);
```

```

scanf("%f", &enteredLaps[i]);
finalTotal = finalTotal + enteredLaps[i];
count = count + 1;
if (enteredLaps[i] < 30 || enteredLaps[i] > 90)
{
    printf("this lap time isn't possible. \nplease enter again");
    scanf("%f", &enteredLaps[i]);
}
}

printf("\n1. your overall time for all 5 laps was: %.3f\n\n", finalTotal);

finalTotal = finalTotal / 5;

printf("2. your average lap time was: %.3f\n\n", finalTotal);

for (int i = 0; i < 5; i++)
{
    if (fast > enteredLaps[i])
    {
        fast = enteredLaps[i];
    }
}

printf("3. your fastest lap was: %.3f\n\n", fast);

for (int i = 0; i < 5; i++)
{
    if (slow < enteredLaps[i])
    {
        slow = enteredLaps[i];
    }
}

```

```

printf("4. your slowest lap was: %.3f\n", slow);

for (int i = 0; i < 5; i++)
{
    if (enteredLaps[i] <= targetTime)
    {
        countTarget = countTarget + 1;
    }
}

printf("5. %d of your laps are equal or faster that your target\n", countTarget);

printf("6. \nlap 1 time was: %.3f\n", enteredLaps[0]);

for (int i = 0; i < 4; i++)
{
    if(enteredLaps[i+1] > enteredLaps[i])
    {
        printf("lap %d was slower than previous lap\n", lapCount);
    }
    else if (enteredLaps[i+1] < enteredLaps[i])
    {
        printf("lap %d was faster than previous lap\n", lapCount);
    }
    else if (enteredLaps[i+1] == enteredLaps[i])
    {
        printf("lap %d was equal than previous lap\n", lapCount);
    }
    lapCount = lapCount + 1;
}
}

```

2. Code Testing

For each test below, 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?

a) Test a lap time for your programme:

- i) 42 seconds
- ii) 2 seconds
- iii) -4 seconds
- iv) 53444 seconds
- v) Test the target lap time – show that the values entered are reasonable

b) Using the same five lap times as the sample output ie

Lap 1: 42.3
Lap 2: 42.1
Lap 3: 44
Lap 4: 44
Lap 5: 39.4

Examine each statistic (1-6) and comment on whether the correct answer(s) are displayed or not. (If any statistics were incomplete after Phase 1 (code completed in class) simply write "Unfinished")

c) Test two further sets of lap times of your own choice.

Template for tests:

Test: i – 42 seconds

Screenshot(s) of output:

```
What is your target time to achieve?           Target = 42
please enter your lap times (in seconds)

Lap time 1 :42
Lap time 2 :|
```

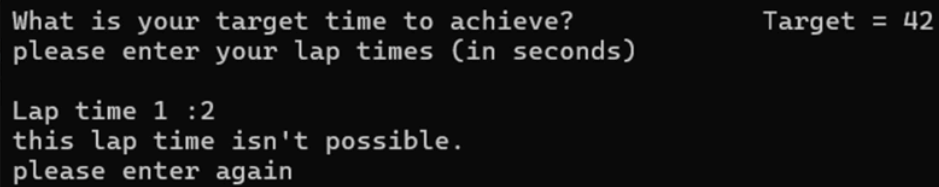
Result: Did the programme behave as expected ie syntactically and logically correct?

YES

The code continued how it should prompting for the next lap time after inputting 42 seconds into the programme

Test: ii – 2 seconds

Screenshot(s) of output:

A screenshot of a terminal window with a black background and white text. The text shows a program prompt asking for a target time, which is set to 42. It then asks for lap times. The user enters 2 seconds for the first lap, and the program responds that this is not possible and asks for a new input.

```
What is your target time to achieve?           Target = 42
please enter your lap times (in seconds)

Lap time 1 :2
this lap time isn't possible.
please enter again
```

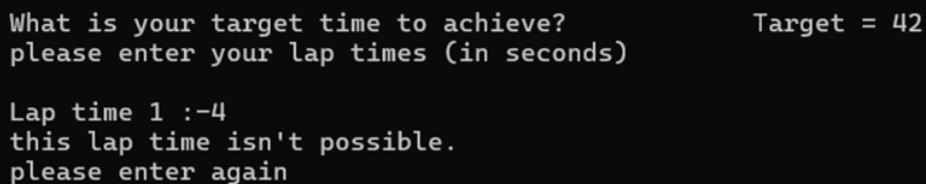
Result: Did the programme behave as expected ie syntactically and logically correct?

YES

After inputting 2 seconds into the lap time the programme behaved exactly how expected with it asking for a new input for the first lap time as 2 seconds is not a possible lap time

Test: iii - -4 seconds

Screenshot(s) of output:

A screenshot of a terminal window with a black background and white text. The text shows the same program prompt as in the previous test. The user enters -4 seconds for the first lap, and the program responds that this is not possible and asks for a new input.

```
What is your target time to achieve?           Target = 42
please enter your lap times (in seconds)

Lap time 1 :-4
this lap time isn't possible.
please enter again
```

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

YES

The programme behaved as expected as the input was out of the parameters of an achievable lap time and got the user to input a reasonable lap time

Test: iv – 53444 seconds

Screenshot(s) of output:

```
What is your target time to achieve?          Target = 42
please enter your lap times (in seconds)

Lap time 1 :53444
this lap time isn't possible.
please enter again
```

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

YES

The lap time entered is not a reasonable time as it is approximately 15 hours so the programme asked to input a time fitting to a suitable lap

Test: v – target times

Screenshot(s) of output:

```
What is your target time to achieve?          Target = 50
please enter your lap times (in seconds)

Lap time 1 :

What is your target time to achieve?          Target = -10
please enter your lap times (in seconds)

Lap time 1 :|

What is your target time to achieve?          Target = 9999
please enter your lap times (in seconds)

Lap time 1 :|
```

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

NO, what went wrong and why?

After inputting three different target times one being normal and two that were unreasonable there were no restrictions with the input the user can provide, and an improvement can be to add validation to ensure target times can only be between two specific times

Test: examine different statistic sections

Screenshot(s) of output:

```
What is your target time to achieve?           Target = 42
please enter your lap times (in seconds)

Lap time 1 :42.3
Lap time 2 :42.1
Lap time 3 :44
Lap time 4 :44
Lap time 5 :39.4

1. your overall time for all 5 laps was: 211.800
2. your average lap time was: 42.360
3. your fastest lap was: 39.400
4. your slowest lap was: 44.000
5. 1 of your laps are equal or faster than your target
6.
lap 1 time was: 42.300
lap 2 was faster than previous lap
lap 3 was slower than previous lap
lap 4 was equal than previous lap
lap 5 was faster than previous lap
```

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

Section 1) the first area of the statistics is the overall time to complete all 5 laps, and this has worked how it should have and added all the laps to produce a correct overall time

Section 2) for the average lap time this is also correct as the programme uses the overall time and divides that by 5 to come out with the right average lap time

Section 3) the 3rd statistic is for the fastest lap of all 5 laps, and the produced result is correct and has given the fastest out of all the other laps

Section 4) this section is all about the slowest lap the programme extracted the correct slowest lap and displayed it correctly in the order of the stats

Section 5) this section shows what laps are faster or equal to the target lap time and this came out correct and showed the number of laps that were fitting to the criteria of this area of stats

Section 6) the final section has produced all correct results of listing the faster or slower laps comparing to the previous lap and displayed as expected

Test: examine different statistic sections

Screenshot(s) of output:

```
What is your target time to achieve?           Target = 45
please enter your lap times (in seconds)

Lap time 1 :43.5
Lap time 2 :46.3
Lap time 3 :45.1
Lap time 4 :42
Lap time 5 :47.8

1. your overall time for all 5 laps was: 224.700
2. your average lap time was: 44.940
3. your fastest lap was: 42.000
4. your slowest lap was: 47.800
5. 2 of your laps are equal or faster that your target
6.
lap 1 time was: 43.500
lap 2 was slower than previous lap
lap 3 was faster than previous lap
lap 4 was faster than previous lap
lap 5 was slower than previous lap
```

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

All the sections for this set of lap times behaved correctly and has produced all the information properly and each sections result has a correct result correlating to the lap times that have been inputted by the user

Test: examine different statistic sections

Screenshot(s) of output:

```
What is your target time to achieve?           Target = 40
please enter your lap times (in seconds)

Lap time 1 :39.5
Lap time 2 :40.2
Lap time 3 :40
Lap time 4 :41
Lap time 5 :38.9

1. your overall time for all 5 laps was: 199.600
2. your average lap time was: 39.920
3. your fastest lap was: 38.900
4. your slowest lap was: 41.000
5. 3 of your laps are equal or faster that your target
6.
lap 1 time was: 39.500
lap 2 was slower than previous lap
lap 3 was faster than previous lap
lap 4 was slower than previous lap
lap 5 was faster than previous lap
```

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

For the final set of lap times for the test the programme completed all the statistics correctly and logically producing an understandable list of sections with correctly extracted and calculated stats to fit the inputted lap times of this programme

3. Consider the results of the above tests 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 <math.h>

void main()
{
    float enteredLaps[5];
    float finalTotal=0;
    float targetTime=0;
    float fast = 1000;
    float slow = 0;
    int count = 1;
    int lapCount = 2;
    int countTarget = 0;

    printf("What is your target time to achieve?           Target = ");
    scanf("%f", &targetTime);
    while (targetTime < 30 || targetTime > 90)
    {
        if (targetTime < 30 || targetTime > 90)
        {
            printf("this is an unreasonable lap time. \n please enter again: ");
            scanf("%f", &targetTime);
        }
    }
    printf("please enter your lap times (in seconds)\n\n");

    for (int i = 0; i < 5; i++)
    {
        printf("Lap time %d :",count);
        scanf("%f", &enteredLaps[i]);
        finalTotal = finalTotal + enteredLaps[i];
        count = count + 1;
        while (enteredLaps[i] < 30 || enteredLaps[i] > 90)
        {
            if (enteredLaps[i] < 30 || enteredLaps[i] > 90)
            {
                printf("this lap time isn't possible. \nplease enter again: ");
                scanf("%f", &enteredLaps[i]);
            }
        }
    }

    printf("\n1. your overall time for all 5 laps was: %.3f\n\n", finalTotal);
```

```

finalTotal = finalTotal / 5;

printf("2. your average lap time was: %.3f\n\n", finalTotal);

for (int i = 0; i < 5; i++)
{
    if (fast > enteredLaps[i])
    {
        fast = enteredLaps[i];
    }
}
printf("3. your fastest lap was: %.3f\n\n", fast);

for (int i = 0; i < 5; i++)
{
    if (slow < enteredLaps[i])
    {
        slow = enteredLaps[i];
    }
}
printf("4. your slowest lap was: %.3f\n\n", slow);

for (int i = 0; i < 5; i++)
{
    if (enteredLaps[i] <= targetTime)
    {
        countTarget = countTarget + 1;
    }
}
printf("5. %d of your laps are equal or faster that your target\n\n", countTarget);

printf("6. \nlap 1 time was: %.3f\n", enteredLaps[0]);

for (int i = 0; i < 4; i++)
{
    if(enteredLaps[i+1] > enteredLaps[i])
    {
        printf("lap %d was slower than previous lap\n", lapCount);
    }
    else if (enteredLaps[i+1] < enteredLaps[i])
    {
        printf("lap %d was faster than previous lap\n", lapCount);
    }
    else if (enteredLaps[i+1] == enteredLaps[i])
    {
        printf("lap %d was equal than previous lap\n", lapCount);
    }
    lapCount = lapCount + 1;
}
}

```


FINAL GRADE

17/0

GENERAL COMMENTS

WORKSHEET 3

Coding- Might be an idea to tell the user what the laptime range is to improve the user interface ("Please enter lap time 30-60") Good overall. There is some scope to combine the for loop for 3 and 4 (for example) for efficiency.

The if on line 35 isn't really needed as it's covered by the while loop that it's inside. Good improvement on the validation of the target lap time

8.5/10

Testing - Good, suitably discussed, you could have started what the limits of the target lap time was for clarity.

9/10

Overall- 17.5/20

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12
