

Jacob Nicklin - Worksheet 1

COMPLETED

by Jacob Nicklin

Submission date: 10-Jan-2024 04:41AM (UTC+0000)

Submission ID: 221292466

File name: 50551_Jacob_Nicklin_Jacob_Nicklin_-_Worksheet_1_COMPLETED_541134_1020486986.docx
(170.97K)

Word count: 444

Character count: 2039

WORKSHEET 1 – TESTING (Using Variables)

1. Original code:

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

void main()
/*
{
int age;
char mySurname[18], nationalInsurance[8];
printf("Please enter your Surname: ");
scanf("%s", &mySurname);
printf("\n\nPlease enter NIN: ");
scanf("%s", &nationalInsurance);

printf("Please enter your age, be realistic!: ");
scanf("%d", &age);

    while (age > 100)
r
```

2. Code Testing

For each test below, run your code and enter the values provided. Provide a screenshot of the programme running. Did the programme behave correctly or did it do something unexpected? Fill in a box for each test below.

Surname:

- i. Enter *JONES*
- ii. Leave blank
- iii. Enter *123*

National Insurance Number:

- i. Enter *PE 25 34 12 B*
- ii. Leave blank
- iii. Enter *AB 11 22 33 44 55 66 77* (ie too long for National Insurance number format)

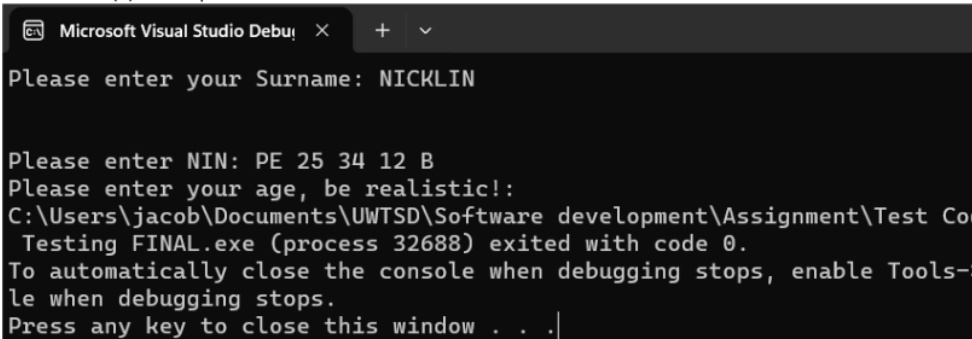
Age

- i. Enter *34*
- ii. Enter *150*
- iii. Enter *-4*
- iv. Enter a letter

Test:

Surname i. Enter *JONES*

Screenshot(s) of output:



```
Microsoft Visual Studio Debug Console
Please enter your Surname: NICKLIN

Please enter NIN: PE 25 34 12 B
Please enter your age, be realistic!:
C:\Users\jacob\Documents\UWTSD\Software development\Assignment\Test Code\Testing FINAL.exe (process 32688) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Environment->Automatically close console when debugging stops.
Press any key to close this window . . .|
```

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

NO

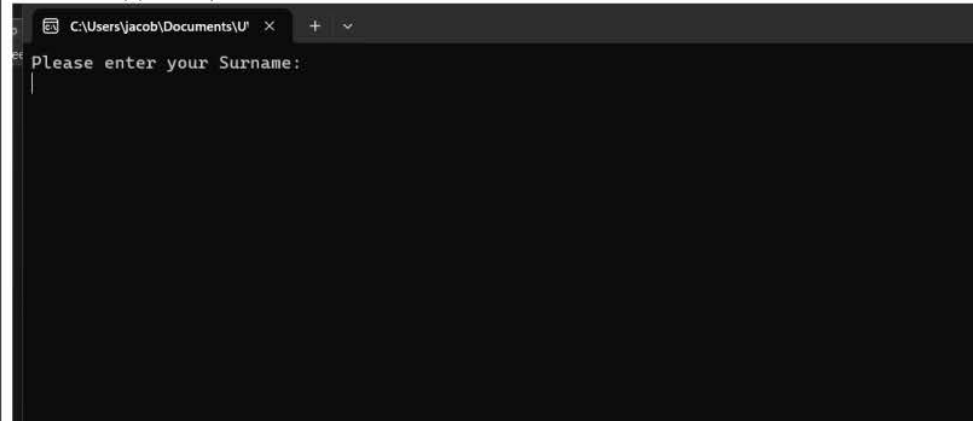
After entering the surname and the national insurance, the programme immediately ended and did not give a chance to input my age, this is most likely down to there being spaces in the NIN or the presence of both letters and numbers in the NIN.

I'll be looking to improve this by being able to input the NIN with spaces in it

Test:

Surname ii Leave blank

Screenshot(s) of output:



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

NO

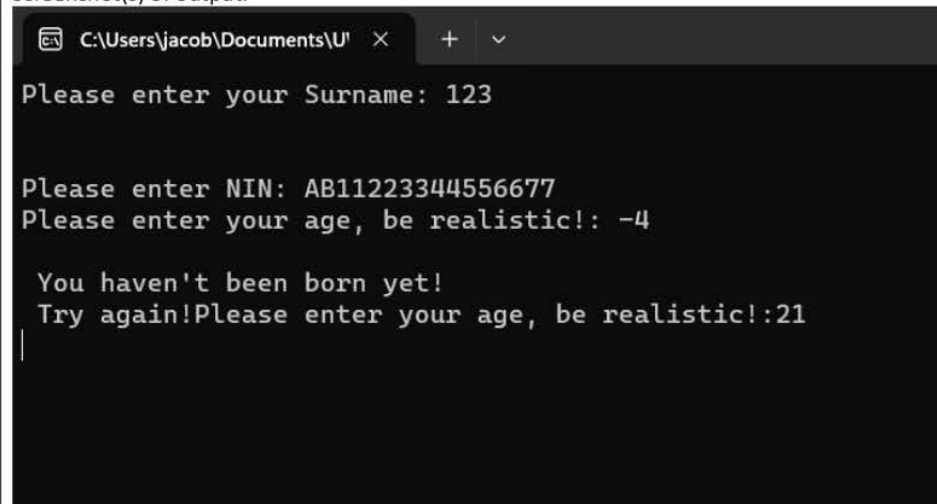
Displayed in the screenshot is the cursor below the input section. I experimented by hitting enter and the cursor just continued to work its way down the page. I can improve this by allowing spaces in my code or just having it end the programme since it's an invalid input. This won't apply to the national insurance as the national insurance needs spaces.

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

Test:

i. Enter 123

Screenshot(s) of output:



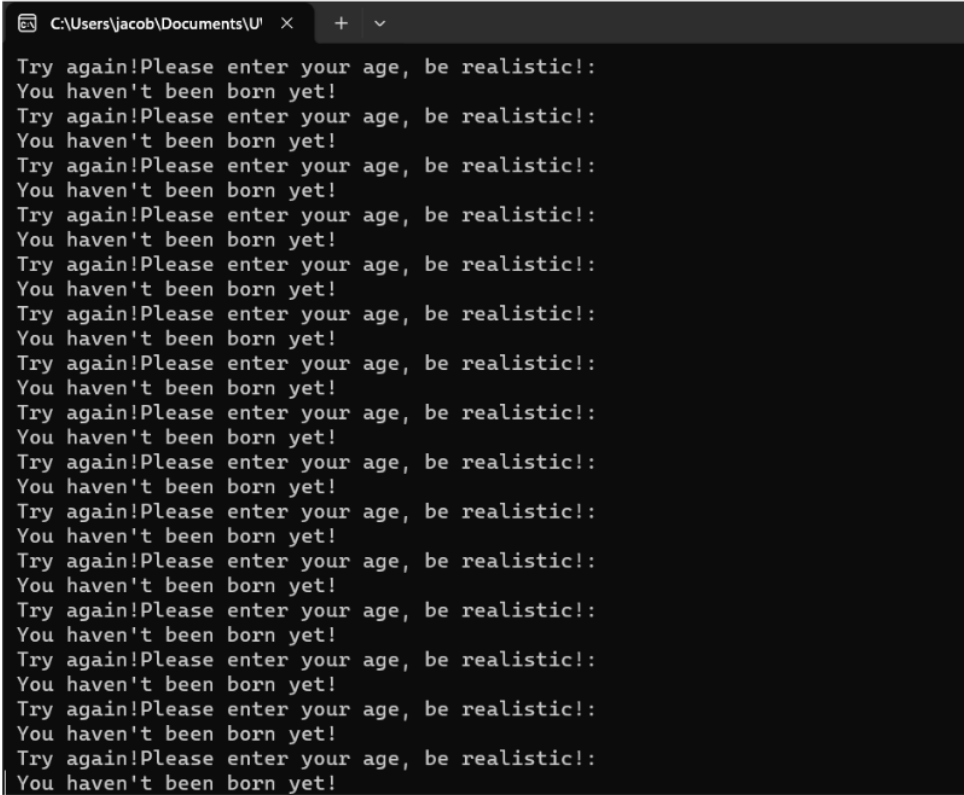
Result: Did the programme behave as expected ie syntactically and logically correct?

YES/NO

It behaved as a bit of both, the validation of the age actually worked and stopped the input of "-4". However, it did not stop 123 or stop the invalid syntax on the NIN, these will have to be amended in the updated code.

Test:
Enter a letter

Screenshot(s) of output:



```
C:\Users\jacob\Documents\U' x + v
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
Try again!Please enter your age, be realistic!:
You haven't been born yet!
```

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

NO

Entering a letter went catastrophically wrong and caused the scanf function to freak  1sing an
if statement to check if an input is not equal to a number would possibly stop this?

```

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

int main()
{
    int age;
    char nationalInsurance[20];
    char mySurname[18];
    int SurnameL;

    printf("Please enter your Surname: ");
    fgets(mySurname, 18, stdin);
    SurnameL = strlen(mySurname);

    printf("\n\nPlease enter NIN: ");
    fgets(nationalInsurance, 20, stdin);

    printf("Please enter your age, be realistic!: ");
    scanf("%d", &age);

    while (age > 100)
    {
        printf("\nSorry, there's no way you're older than 100!\n Try again!");
        printf("Please enter your age, be realistic!:");
        scanf("%d", &age);
    }
    while (age <= 1)
    {
        printf("\n You haven't been born yet!\n Try again!");
        printf("Please enter your age, be realistic!:");
        scanf("%d", &age);
    }
    printf("\n\n-----\n\n");
    printf("\nyour surname is: %s ", mySurname);
    printf("\nYour NIN is: %s", nationalInsurance);
    printf("\nYour age is: %d", age);
    printf("\n\n-----\n\n");
}

```

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

Paste your revised code here:

Jacob Nicklin - Worksheet 1 COMPLETED

GRADEMARK REPORT

FINAL GRADE

/10

GENERAL COMMENTS

Worksheet 1:

Coding:

Be careful with the age validation. You could enter -4 and then 150 in that order and 150 would get through. Combine into one while with multiple conditions (see inline comment)

Good progress made on resolving problems with the tests, if you were going to implement this in a live system, you would need to look at the format of the National Insurance number itself, rather than just the space of the input

3.5/5

Testing:

Good, you have understood this section well and suitably presented your results.

5/5

Total: 8.5/10

PAGE 1

PAGE 2

PAGE 3

PAGE 4



Comment 1

This made me smile - I agree, but find a more academic expression :)



Comment 2

Combine to while (age>100 || age<1)

Jacob Nicklin - Worksheet 2

COMPLETED

by Jacob Nicklin

Submission date: 10-Jan-2024 03:25AM (UTC+0000)

Submission ID: 221289159

File name: 50551_Jacob_Nicklin_Jacob_Nicklin_-_Worksheet_2_COMPLETED_541134_959482715.docx (91.4K)

Word count: 696

Character count: 3315

WORKSHEET 2 – TESTING (Carrots)

1. Original code:

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

```
void main()

{
    int width, length;
        float oneCarrot = 64;
        float total;
        float totalCarrots;
    printf("please enter the width of the bed: ");
    scanf("%d", &width);
    while (width <= 0)
    {
        printf("\n\nGive appropriate measurements or I'll explode!");
        scanf("%d", &width);
    }

    printf("\n\nPlease enter the length of the bed: ");
        scanf("%d", &length);
    while (length <= 0)
    {
        printf("Give appropriate measurements or I'll explode!");
        scanf("%d", length);
    }
    total = (width * length);

    printf("Your bed is %f cm", &total);

    totalCarrots = total / 64;

    while (oneCarrot < 64)
    {
        floor(totalCarrots);
        printf(" your total amount of carrots you cna fit in the area is %f", totalCarrots);
    }
}
```

2. Code Testing

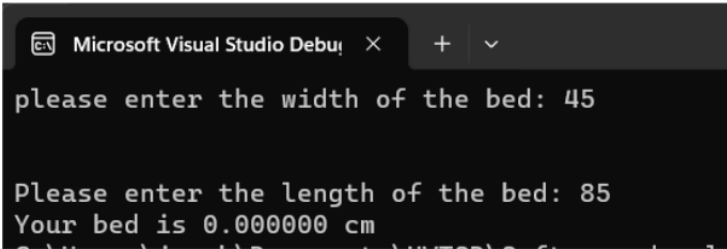
For each test below, run your code and enter the values provided. Provide a screenshot of the programme output. Did the programme behave correctly or did it do something unexpected? Fill in a box for each test below.

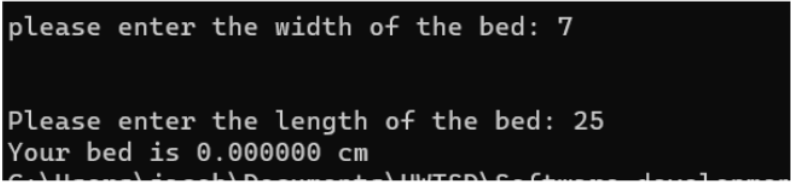
Input:

- i. Enter dimensions: Width 45 Length 85
- ii. Enter dimensions: Width 7 Length 25
- iii. Enter dimensions: Width -16 Length -20

Logic:

- i. Enter dimensions: Width 27 Length 30. The result should be 9 carrots.
- ii. Enter dimensions: Width 32 Length 32. The result should be 16 carrots.
- iii. Enter dimensions: Width 7 Length 16. The result should either be 0 carrots, or you may have accounted for this in your validation at the input phase.

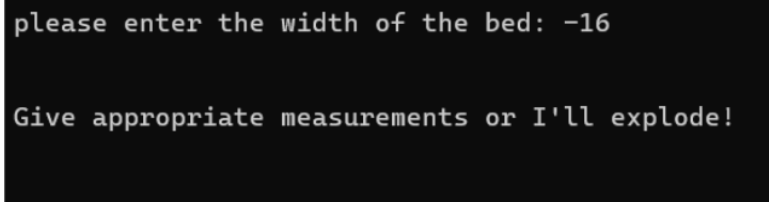
Test: Input - Enter dimensions: width 45 length 85
Screenshot(s) of output: 
Result: Did the programme behave as expected ie syntactically and logically correct? NO If NO, what went wrong? The first and most notable error is the complete failure of the total area result. My first mistake in that was use a float variable instead of an integer in the area and also not including other variables that would've allowed me to calculate the total area.

Test: Enter dimensions: width 7 length 25
Screenshot(s) of output: 
Result: Did the programme behave as expected ie syntactically and logically correct? NO (delete as appropriate) If NO, what went wrong? Again, the calculation is completely wrong, which means that due to me missing a whole variable in my code, the code does not function as it should! I will be amending it correctly within the test phase.

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

Test: Enter dimensions: Width -16 Length -20

Screenshot(s) of output:



```
please enter the width of the bed: -16

Give appropriate measurements or I'll explode!
```

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

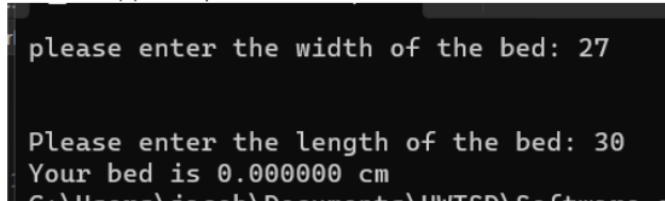
YES/NO

My programme was designed to not to accept anything less than 0 so it's very clear that the validation on that has worked very nicely and it gives an option to enter again (albeit there's no prompt message)

I would like to improve on this by allowing entry on negative numbers but also making sure that no letters or symbols can get through that loophole.

Test: Enter dimensions: Width 27 Length 30. The result should be 9 carrots.

Screenshot(s) of output:

A screenshot of a terminal window with a black background and white text. The text shows the program's prompts and output for the test case: 'please enter the width of the bed: 27', 'Please enter the length of the bed: 30', and 'Your bed is 0.000000 cm'.

```
please enter the width of the bed: 27

Please enter the length of the bed: 30
Your bed is 0.000000 cm
```

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

NO

The input once again did not bring any calculation to the amount of carrots that could fit inside the vegetable patch, so the outcome was not 9 carrots.

To improve upon this, I'll first need to fix the calculation part of the programme and then also change a lot of the variables to integers from floats as I can use that to stop entry on half a carrot or stunted carrots (e.g. 1.5 carrots) which don't count.

Test: Enter dimensions: Width 32 Length 32. The result should be 16 carrots.

Screenshot(s) of output:

```
please enter the width of the bed: 32

Please enter the length of the bed: 32
Your bed is 0.000000 cm
C:\Users\jacob\Documents\UWTSD\Software
```

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

NO: As expected, the calculation failed due to it missing an integer, I will need to double check that all my calculations within the programme are also correct to avoid this in the testing.

- i. Test: Enter dimensions: Width 7 Length 16. The result should either be 0 carrots, or you may have accounted for this in your validation at the input phase.

Screenshot(s) of output:

```
please enter the width of the bed: 7

Please enter the length of the bed: 16
Your bed is 0.000000 cm
C:\Users\jacob\Documents\UWTSD\Software
```

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

YES/NO

Although the Test asks for a result of 0 carrots, it's very clear to see that from the other tests that the 7 and the 16 were not calculated and the programme jumped straight to 0, although the test does mention making sure to account for validation, so adding a check for anything that goes into the negative numbers or decimals is important.

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 <Windows.h>

void main()
{
    int EntryW, EntryL;
    int total;
    printf("Hello! Welcome to the carrot calculator! please put in WHOLE\nNUMBERS ONLY\n\n...Or I'll explode :)");
    Sleep(3000);
    system("cls");

    printf("please enter the width of your carrot bed: ");
    scanf("%d", &EntryW);
    while (EntryW <= 0)
    {
        printf("\nPlease put in a positive number");
        printf("\n\nPlease enter the width of the bed: ");
        scanf("%d", &EntryW);
    }
    if (EntryW)
        Sleep(2000);
    system("cls");
    printf("Please enter the length of your carrot bed: ");
    scanf("%d", &EntryL);
    while (EntryL <= 0)
```

Jacob Nicklin - Worksheet 2 COMPLETED

GRADEMARK REPORT

FINAL GRADE

/10

GENERAL COMMENTS

Worksheet 2:

Coding:

The width and length validation should actually be less than 8 - which is the minimum, not 0.

you have not addressed the logic of the number of carrots that can fit in the bed - each needs and 8 x 8 square. If you do length*width you get the total area of the bed. This will calculate that you can grow more carrots than you can in reality. You need to relate the carrot bed dimensions to the 8cm minimum dimensions on length and width. It's not enough to say carrot needs 64 - one of the 64s in total could be the sum of leftover strips of bed.

2.5/5

Testing: Good discussion. Happy that you've identified the main points.

5/5

Total: 7.5

PAGE 1

PAGE 2



Comment 1

Passive voice: A mistake was made in using a float variable



Comment 2

Passive voice: that would have allowed calculation of the total area

PAGE 3



Comment 3

Passive voice: the code will be amended

PAGE 4

PAGE 5

PAGE 6

PAGE 7



Comment 4

8 is the minimum



Comment 5

8

Jacob Nicklin - Worksheet 3 COMPLETED

by Jacob Nicklin

Submission date: 10-Jan-2024 03:33AM (UTC+0000)

Submission ID: 221289497

File name: 50551_Jacob_Nicklin_Jacob_Nicklin_-_Worksheet_3_COMPLETED_541134_1026695693.docx
(166.62K)

Word count: 439

Character count: 2132

WORKSHEET 3 – TESTING (Arrays)

1. Original code:

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

void main()
{
    char products[5][20];
    float price[5];
    int count = 1;
    float total;
    float staffPrice;

    for (int i = 0; i < 5; i++)
    {

        printf("Please enter product %d: ", count++);
        scanf("%s", &products[i]);
        printf("\n And enter your price: ");
        scanf("%f", &price[i]);
        printf("\n\n");

    }
    printf("Products Entered: ");

    for (int i = 0; i < 5; i++)
    {
        printf("\n %s: \x9c%.2f\n", products[i], price[i]);
    }
}
```

2. Code Testing

For each test below, run your code and enter the values provided. Provide a screenshot of the programme output. Did the programme behave correctly or did it do something unexpected? Fill in a box for each test below.

- i. Enter the 5 normal values as shown on the example screenshot (on the question sheet)
ie *pizza 3.45, chips 1.20, tea 1.35, coffee 1.82* and *banana 0.17*
Double check the staff discount for the above values, is it correct?
- ii. Enter *pizza -2.34 chips -1.10, tea -1.22, coffee -2.05* and *banana -0.18*
- iii. Enter your own values, but leave some of them blank

Test: i. Enter the 5 normal values as shown on the example ie <i>pizza 3.45, chips 1.20, tea 1.35, coffee 1.82</i> and <i>banana 0.17</i> Double check the staff discount for the above values, is it correct?
--

Screenshot(s) of output:

```
Microsoft Visual Studio Debug Console
Please enter product 1: pizza
And enter your price: 3.45

Please enter product 2: chips
And enter your price: 1.20

Please enter product 3: tea
And enter your price: 1.35

Please enter product 4: coffee
And enter your price: 1.82

Please enter product 5: banana
And enter your price: 0.17

Products Entered:
pizza: £3.45
chips: £1.20
tea: £1.35
coffee: £1.82
banana: £0.17

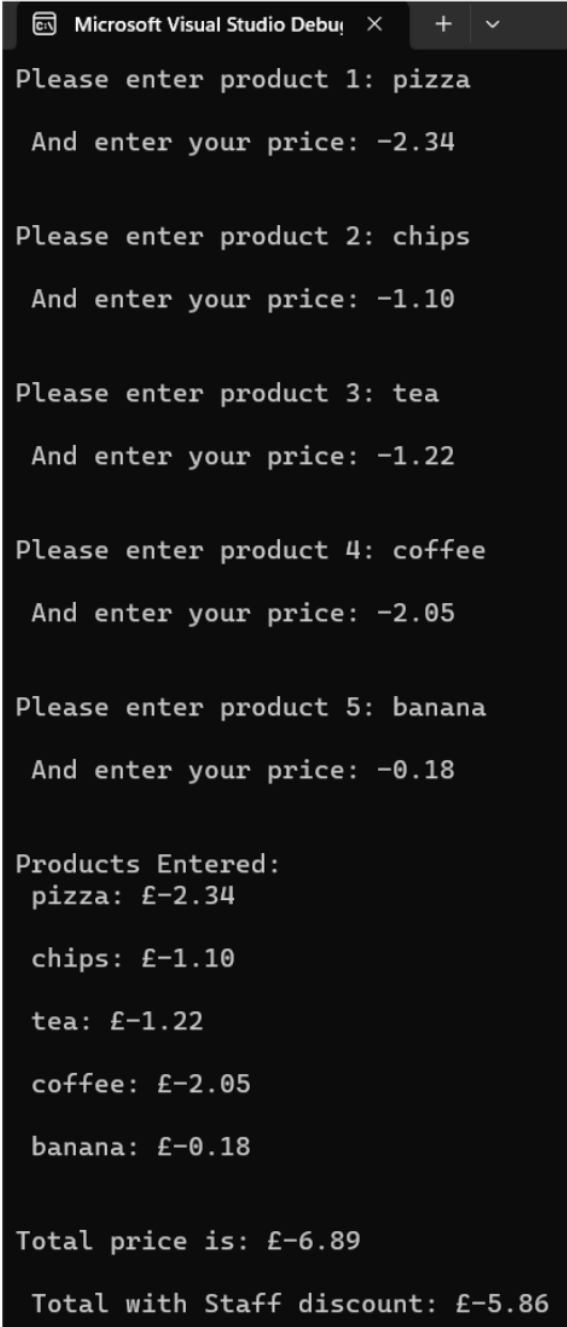
Total price is: £7.99

Total with Staff discount: £6.79
```

Result: Did the programme behave as expected ie syntactically and logically correct?
YES
The inputs were all entered correctly and the calculations for the total price and the staff discount worked perfectly too!

Test: ii. Enter *pizza* -2.34 *chips* -1.10, *tea* -1.22, *coffee* -2.05 and *banana* -0.18

Screenshot(s) of output:

A screenshot of the Microsoft Visual Studio Debug Console window. The window has a title bar with the Visual Studio icon, the text 'Microsoft Visual Studio Debug Console', and standard window controls (close, maximize, and a dropdown arrow). The console output is as follows:

```
Please enter product 1: pizza
And enter your price: -2.34

Please enter product 2: chips
And enter your price: -1.10

Please enter product 3: tea
And enter your price: -1.22

Please enter product 4: coffee
And enter your price: -2.05

Please enter product 5: banana
And enter your price: -0.18

Products Entered:
pizza: £-2.34
chips: £-1.10
tea: £-1.22
coffee: £-2.05
banana: £-0.18

Total price is: £-6.89

Total with Staff discount: £-5.86
```

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

YES/NO

It worked perfectly and calculated all the values in their minuses and even into the staff discount. However, I will need to take this as a precaution and use a ≤ 0 to prevent numbers lower than 0 being entered.

Test: iii) Enter a few of your own values, leave some of the inputs blank

Screenshot(s) of output:

Please enter product 1: Pineapp

And enter your price: 6.77

Please enter product 2:

p

And enter your price:

l

Please enter product 3:

And enter your price: 45

Please enter product 4: carrots

And enter your price: 4.55

Please enter product 5: orange

And enter your price: 6.77

Products Entered:

Pineapple: £6.77

p: £-107374176.00

l: £45.00

carrots: £4.55

orange: £6.77

Total price is: £-107374104.00

Total with Staff discount: £-91267992.00

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

NO

The programme would not let me input blanks and I eventually had to input random letters just to continue, once I reached the total pricing, the calculations went haywire and gave completely



incorrect numbers, There's no surefire way to stop a user from entering blanks or random characters but I can stop them from entering negative numbers, validating the code a bit more.

Any other tests you feel may be relevant, or reveal a strength or deficiency in your original code, please detail below:

Test:
Screenshot(s) of output:
Result: Did the programme behave as expected ie syntactically and logically correct? YES/NO (delete as appropriate) If NO, what went wrong?

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>

void main()
{
    char products[5][20];
    float price[5];
    int count = 1;
    float total;
    float staffPrice;

    for (int i = 0; i < 5; i++)
    {

        printf("Please enter product %d: ", count++);
        scanf("%s", &products[i]);
        printf("\n And enter your price: ");
        scanf("%f", &price[i]);
        printf("\n\n");
        while (price[i] <= 0)
        {
            printf("You can't enter a negative number onto the
menu!\n That would be giving the customer money!");
            printf("\n\nPlease enter your price!: ");
            scanf("%f", &price[i]);
        }
    }
}
```

Jacob Nicklin - Worksheet 3 COMPLETED

GRADEMARK REPORT

FINAL GRADE

/20

GENERAL COMMENTS

Worksheet 3:

Coding:

You could revise for efficiency. You need to loop total = price[0]+price[1]+price[2]+price[3]+price[4] rather than hard code.

Also the letter/numbers issue (see below)

7/10

Testing:

You could have look at entering letters and numbers in the wrong place also

8/10

Total: 15/20

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5



Comment 1

Has to be no as prices can't be negative

PAGE 6



Comment 2

A more academic word would be nice :)

PAGE 7

PAGE 8

PAGE 9

Jacob Nicklin - Worksheet 4

COMPLETE

by Jacob Nicklin

Submission date: 10-Jan-2024 03:14AM (UTC+0000)

Submission ID: 221288675

File name: 50551_Jacob_Nicklin_Jacob_Nicklin_-_Worksheet_4_COMPLETE_541134_2119102226.docx
(125.54K)

Word count: 551

Character count: 2485

8WORKSHEET 4 – TESTING (Arrays)

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

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <Windows.h>
#include <string.h>
#include "Header.h"

void main()
{
    int Ran1, Ran2, Ran3, points = 0;
    char PlayAgain = 'Y';

    srand(time(NULL));
    printf("Let's spin the Fruit Machine!");
    Sleep(3000);
    system("cls");
    while (PlayAgain == 'Y')
    {
```

1. Original code:

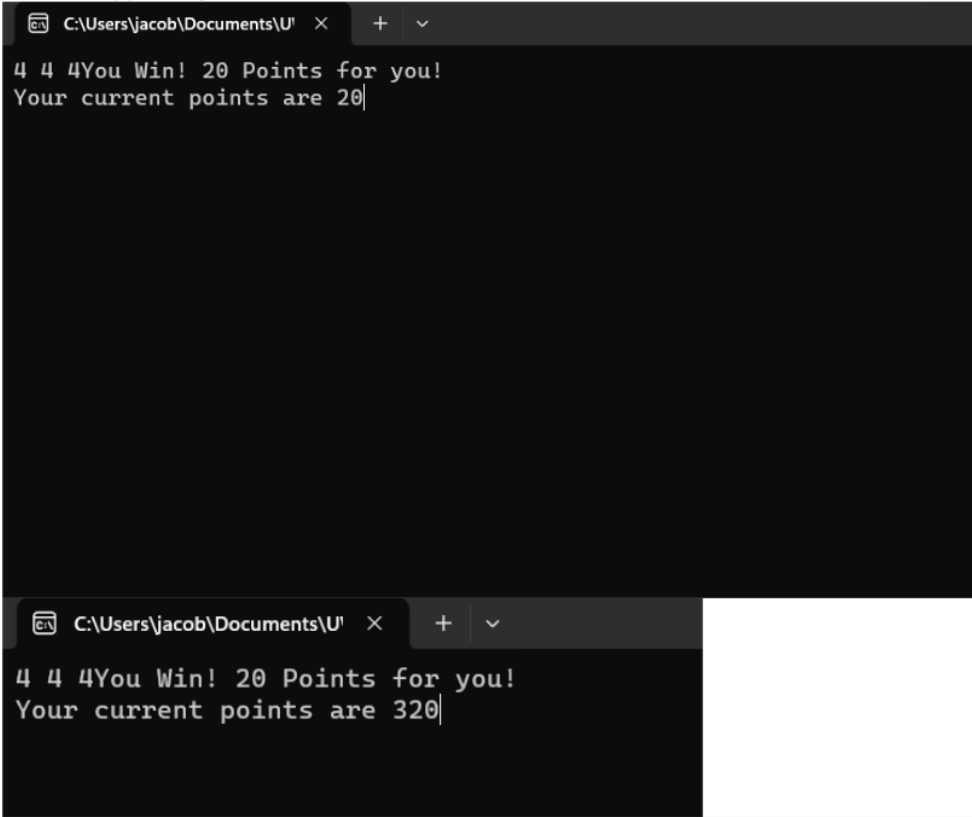
2. Code Testing

For each test below, run your code and enter the values provided. Provide a screenshot of the programme output. Did the programme behave correctly or did it do something unexpected? Fill in a box for each test below.

Test the following values for your fruit machine:

- i. 4 4 4 (20 points)
- ii. 1 3 2 (0 points)
- iii. 5 5 3 (100 points)
- iv. 5 2 5 (100 points)
- v. 1 5 5 (100 points)
- vi. 5 5 5 (2000 points)

vii. Enter "N" when user is asked whether they wish to make another spin

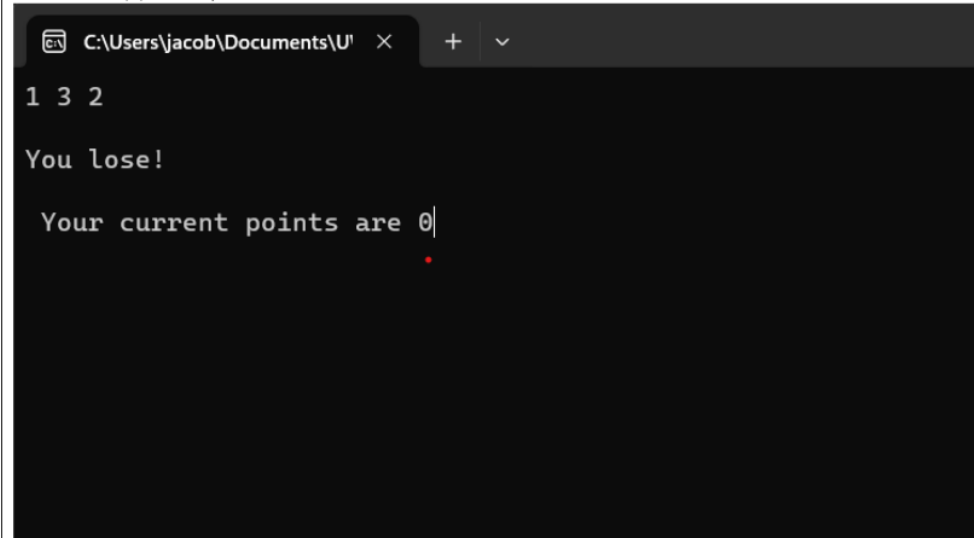
Test: i. 444 (20 points)
Screenshot(s) of output:

Result: Did the programme behave as expected ie syntactically and logically correct? NO

If NO, what went wrong?

Nothing went wrong with the random number generation itself, although as you can see in the second screenshot that because I have not put in any input or choice to play again into the “while” loop, the code loops automatically, constantly pulling the test values and updating the score like it should. Another thing I missed is a new line between the number generation and the “You Win” printf statement.

Test: ii. 1 3 2 (0 points)

Screenshot(s) of output:



```
C:\Users\jacob\Documents\U' x + v
1 3 2
You lose!
Your current points are 0|
```

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

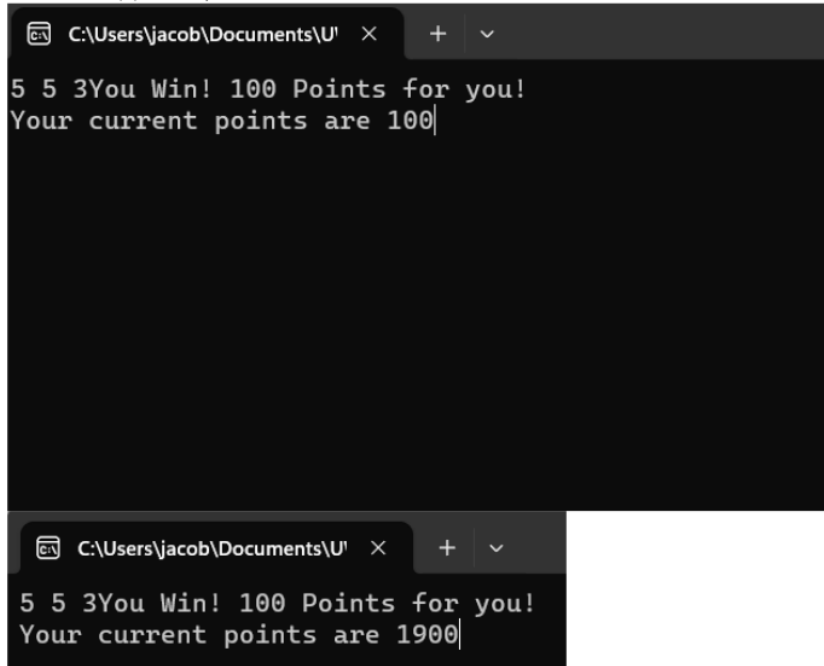
YES

The test Values updated the score to zero.

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

Test: 5 5 3 (100 points)

Screenshot(s) of output:



The image shows two screenshots of a terminal window. The top screenshot shows the output '5 5 3You Win! 100 Points for you!' and 'Your current points are 100|'. The bottom screenshot shows the output '5 5 3You Win! 100 Points for you!' and 'Your current points are 1900|'. Both screenshots have a dark background and a light-colored text. The terminal window has a title bar that reads 'C:\Users\jacob\Documents\U' and a tab icon.

```
5 5 3You Win! 100 Points for you!  
Your current points are 100|
```

```
5 5 3You Win! 100 Points for you!  
Your current points are 1900|
```

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

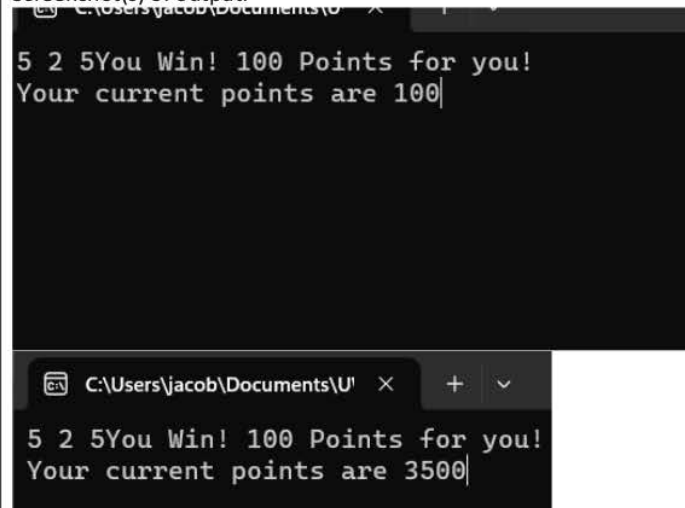
NO

If NO, what went wrong?

Again, the random number generation worked perfectly. I have the same errors as the 4 4 4 values which is that I have no code which stops the while loop from constantly looping every few seconds. I also forgot to put a "/n" next to "You Win" on this one as well.

Test: 5 2 5 (100 points)

Screenshot(s) of output:



```
5 2 5You Win! 100 Points for you!  
Your current points are 100|
```

```
5 2 5You Win! 100 Points for you!  
Your current points are 3500|
```

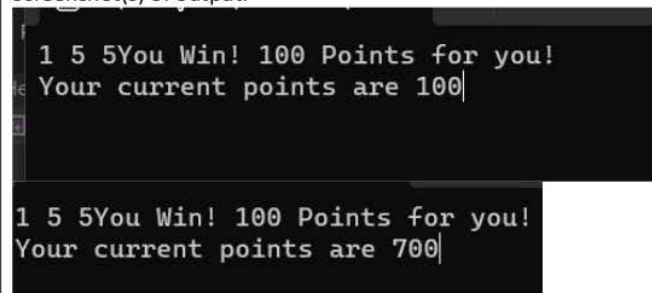
Result: Did the programme behave as expected ie syntactically and logically correct?

NO

Again, same errors as the previous ones, there is no input barrier to stop the programme for automatically looping

Test: 1 5 5 (100 points)

Screenshot(s) of output:



```
1 5 5You Win! 100 Points for you!  
Your current points are 100|
```

```
1 5 5You Win! 100 Points for you!  
Your current points are 700|
```

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

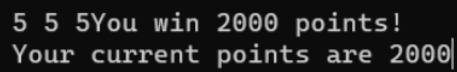
NO

Once more, the lack of any input leads to the score updating itself without any user input, also there is another lack of "\n" next to "you win"

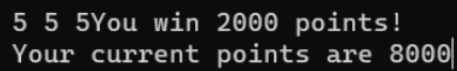
Any other tests you feel may be relevant, or reveal a strength or deficiency in your original code, please detail below:

Test: 5 5 5 (2000 points)

Screenshot(s) of output:



```
5 5 5You win 2000 points!  
Your current points are 2000|
```



```
5 5 5You win 2000 points!  
Your current points are 8000|
```

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

NO:

To round things out, the test did exactly the same as the last 5, the lack of any user input causes the loop to continue on forever. This will be one of the first things I fix in my updated code in question 3.

```

#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <Windows.h>
#include <string.h>
#include <ctype.h>
#include "Header.h"

void main()
{
    int Ran1, Ran2, Ran3, points = 0;
    char PlayAgain = 'Y';

    srand(time(NULL));
    printf("Let's spin the Fruit Machine!");
    Sleep(3000);
    system("cls");
    while (PlayAgain == 'Y' || PlayAgain == 'y')
    {

        Ran1 = RandomGeneration(1, 5);
        printf("%d ", Ran1);

        Ran2 = RandomGeneration(1, 5);
        printf("%d ", Ran2);

        Ran3 = RandomGeneration(1, 5);
        printf("%d", Ran3);

        if (Ran1 == Ran2 && Ran2 == Ran3 )
        {
            points = points + 20;
            printf("\n\nYou win 20 points!\n");
            printf("Your current points are %d\n", points);

        }

        else if (Ran1 == Ran2)
        {
            points = points + 100;
            printf("\n\nYou Win! 100 Points for you!\n");
            printf("Your current points are %d\n", points);

        }

        else if (Ran1 == Ran3)
        {
            points = points + 100;
            printf("\n\nYou Win! 100 Points for you!\n");
            printf("Your current points are %d", points);

        }

        else if (Ran2 == Ran3)
        {
            points = points + 100;
            printf("\n\nYou Win! 100 Points for you!\n");
            printf("Your current points are %d", points);

        }

        else if (Ran1 == 5 && Ran2 == 5 && Ran3 == 5)
        {
            points = points + 2000;
            printf("\n\nYou Win! 2000 Points for you!\n");
            printf("Your current points are %d", points);
        }
    }
}

```

Jacob Nicklin - Worksheet 4 COMPLETE

GRADEMARK REPORT

FINAL GRADE

/20

GENERAL COMMENTS

Worksheet 4:

Coding:

It would have been better to use an array which would have meant you could have improved code efficiency throughout, and you'd also have the option of using loops - particularly useful for counting 5s rather than long if statements. Nevertheless, good answer with a cumulative score

8/10

Testing:

Testing is OK, what happens if you enter something other than "Y" or "N"? Although you did address this in the code later.

9/10

Total: 17/20

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

Jacob Nicklin-Worksheet 5 Complete

by Jacob Nicklin

Submission date: 10-Jan-2024 03:09AM (UTC+0000)

Submission ID: 221288428

File name: 50551_Jacob_Nicklin_Jacob_Nicklin-Worksheet_5_Complete_541134_215454019.docx (95.89K)

Word count: 308

Character count: 1554

WORKSHEET 5 – TESTING (Treasure Hunt)

For this worksheet, there are several possible approaches to the problem.

Create your own test set to show the strengths and/or limitations of your solution. Use as many of the boxes below as you need.

Test:

1. Enter choice to play: Yes or No
2. Print the map
3. Allow user to input number and update the map accordingly
4. Repeat step 3 until they win

Screenshot(s) of output:

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

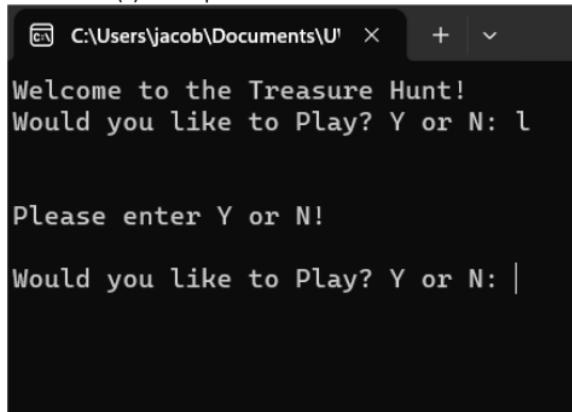
YES

The choice to input yes or no worked perfectly and would end the programme if invalid syntax was used.

The Map generated perfectly too, allowing for input afterwards and updated the map accordingly, replacing the incorrect squares with 8s and the winning square with 5 and stopping the programme afterwards!

Test: Entering wrong letters into "Y or N" input

Screenshot(s) of output:



```
C:\Users\jacob\Documents\U' x + v
Welcome to the Treasure Hunt!
Would you like to Play? Y or N: l

Please enter Y or N!
Would you like to Play? Y or N: |
```

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

YES

The choice to input yes or no worked perfectly and would end the programme if invalid syntax was used. It then gave the options back up, showcasing the validation.

Test: Entering negative numbers into the map

Screenshot(s) of output:

```
Welcome to the Treasure Hunt!  
Would you like to Play? Y or N: l
```

```
Please enter Y or N!
```

```
Would you like to Play? Y or N: Y
```

```
Awesome! Here is the map!
```

```
0 0 0 0  
0 0 0 0  
0 0 0 0  
0 0 0 0
```

```
Please enter a number (0-15): -57
```

```
Stay within the numbers given!
```

```
Please enter a number (0-15): |
```

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

YES

The input of negative numbers led to the programme reading "stay within the numbers given!" and providing the option of 0-15 again and allowing user input, once again, it is validated.

Test: Entering a letter into the position

Screenshot(s) of output:



```
C:\Users\jacob\Documents\U' x + v

Please enter a number (0-15): 0 0 0 0
0 0 0 8
0 0 0 0
0 0 0 0

Please enter a number (0-15): 0 0 0 0
0 0 0 8
0 0 0 0
0 0 0 0

Please enter a number (0-15): 0 0 0 0
0 0 0 8
0 0 0 0
0 0 0 0

Please enter a number (0-15): 0 0 0 0
0 0 0 8
0 0 0 0
0 0 0 0

Please enter a number (0-15): 0 0 0 0
0 0 0 8
0 0 0 0
0 0 0 0

Please enter a number (0-15): 0 0 0 0
0 0 0 8
0 0 0 0
0 0 0 0
```

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

NO

Upon entering a letter into the position input, the programme crashed and ended up repeating the screenshot over and over again. There's no real way to stop a user from entering a letter into a int type variable.

```

#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#include <time.h>
/*global*/
int map[16] = { 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0 };

int main()
{
    /*values*/
    srand(time(NULL));

    char choice;
    int position = 0;
    int WrongTile = 8;
    int RightTile = rand() % 15;
    int count = 0;
    int coordinate;

    printf("Welcome to the Treasure Hunt!\n");
    /*Choosing to Play*/
    printf("Would you like to Play? Y or N: ");
    scanf("%c", &choice);
    choice = tolower(choice);
    while (choice != 'y' && choice != 'n')
    {
        printf("\n\nPlease enter Y or N!");
        printf("\n\nWould you like to Play? Y or N: ");
        scanf(" %c", &choice);
        choice = tolower(choice);
    }
    switch (choice)
    {
    case 'y':
        printf("\n\n");
        break;
    case 'n':
        printf("\n-----\n");
        exit(0);
        break;
    default:
        break;
    }

    //first print of the map
    printf("Awesome! Here is the map!\n\n");
    for (coordinate = 0; coordinate < 16; coordinate++)
    {

        printf("%d ", map[coordinate]);
        count++;

        if (count == 4)
        {
            printf("\n");
            count = count - 4;
        }
    }
}

```


Jacob Nicklin-Worksheet 5 Complete

GRADEMARK REPORT

FINAL GRADE

/30

GENERAL COMMENTS

Worksheet 5:

Coding:

There's a few errors on the way through, for example while (position=WrongTile) - this should be a double equals when comparing.

The user is able to choose the same position twice it might be nice to tell them "You've already chosen that location, try somewhere else"

You have got a decent solution here, but I was looking for function work - the brief does say "at least one function" - I did not mean just main()!

9/15

Testing:

A bit on the short side. You need to be a bit more thorough and make sure you have presented all possibilities. I can't see a win shown here?

9/15

Total: 18/30

**Organisation of report, writing style,
presentation of work:**

It was a bit hard to find your code at times as it had gone off the bottom of the text box - make sure all of your work is displayed so that the marker doesn't miss anything.

Your writing is a little informal at times, this will come with practice. Avoid the use of "I" in academic reports, instead use the passive voice. I've put a few examples inline to show you (there are some on Worksheet 2)

6/10

Overall total for assignment: 72%

Test mark: 69%

Overall mark for module: 71%

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6
