

# Matthew Coleman(2306158)

## Assignment testing 1-5

*by Matthew Coleman*

---

**Submission date:** 10-Jan-2024 12:57AM (UTC+0000)

**Submission ID:** 221280353

**File name:** 49755\_Matthew\_Coleman\_Matthew\_Coleman\_2306158\_Assignment\_testing\_1-5\_541134\_1669768227.docx (349.51K)

**Word count:** 2959

**Character count:** 15433

## **WORKSHEET 1a – TESTING (Using Variables)**

1. Original code:

```
printf("\n\nPlease enter your student number : ");
```

```
scanf("%d", &StudentNumber);
```

```
if (StudentNumber > 2999999 || StudentNumber < 2000000) {
```

```
    printf("please restart enter a Valid Number");
```

```
}
```

```
printf("\n\nPlases enter your Age : ");
```

```
scanf("%d", &Age);
```

```
if (Age > 70 || Age <= 17) {
```

```
    printf("please restart enter a Valid Number");
```

```
}
```

```
printf("\n\nplease enter your House number : ");
```

```
scanf("%s", &HouseNumber);
```

```
printf("\n\nplease enter your street name : ");
```

```
scanf("%s", StreetName);
```

```
printf("\n\nStudent Number: %d\n Student Age: %d \n Address: %s , %s \n", StudentNumber,  
Age, HouseNumber, StreetName);
```

```
}
```

## 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.

Student number:

- i. Enter *2341516*
- ii. Leave blank
- iii. Enter *-45*

House number:

- i. Enter 11A
- ii. Leave blank
- iii. Enter 123417899 (ie too long for a typical British street)

Age:

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

Test:

Student number i. Enter 2341516

Screenshot(s) of output:

```
Please enter your student number : 2341516
```

```
Plases enter your Age : 20
```

```
please enter your House number : 60
```

```
please enter your street name : ChurchRoad
```

```
Student Number: 2341516
```

```
Student Age: 20
```

```
Address: 60 , ChurchRoad
```

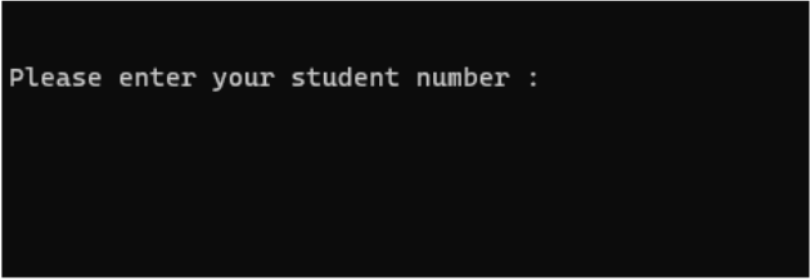
Result: Did the programme behave as expected ie syntactically and logically correct?

YES

Test:

Student number ii Leave blank

Screenshot(s) of output:



```
Please enter your student number :
```

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

NO

Code failed the test as it would not continue after attempting to enter a blank student number nor did it give an error message.

Test : Student Number Enter -45

Screenshot(s) of output:

```
Please enter your student number : -45
please restart enter a Valid Number

Plases enter your Age : 20

please enter your House number : 60

please enter your street name : ChurchRoad

Student Number: -45
Student Age: 20
Address: 60 , ChurchRoad
```

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

YES



Test: House Number Enter 11A

Screenshot(s) of output:

```
Please enter your student number : 2396158
```

```
Plases enter your Age : 20
```

```
please enter your House number : 11a
```

```
please enter your street name : chruchroad
```

```
Student Number: 2396158
```

```
Student Age: 20
```

```
Address: 11a , chruchroad
```

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

YES

Test: Enter House Number : Leave blank

Screenshot(s) of output:

```
Please enter your student number : 2306158

Pleases enter your Age : 20

please enter your House number :
```

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

NO (delete as appropriate)

If NO, what went wrong?

Code failed the test as it would not continue after attempting to enter a blank House number nor did it give an error message.



Test: Enter House Number : Enter 123417899 (ie too long for a typical British street)

Screenshot(s) of output:

```
Please enter your student number : 2306158
```

```
Plases enter your Age : 20
```

```
please enter your House number : 123417899
```

```
please enter your street name : churchroad
```

```
Student Number: 2306158
```

```
Student Age: 20
```

```
Address: 123417899 , churchroad
```

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

NO

Code failed the test as it did not sterilize the data and accepted it and even corrupted the data stack as it was larger than the variables hard code allocation.

Test: Age enter: Enter 34

Screenshot(s) of output:

```
Please enter your student number : 2306158

Pleases enter your Age : 34

please enter your House number : 60

please enter your street name : Churchroad

Student Number: 2306158
Student Age: 34
Address: 60 , Churchroad
```

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

YES

Test: Age Enter 150

Screenshot(s) of output:

```
Please enter your student number : 2306158

Pleases enter your Age : 150
please restart enter a Valid Number

please enter your House number : 60

please enter your street name : ChurchRoad

Student Number: 2306158
Student Age: 150
Address: 60 , ChurchRoad
```

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

YES



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

Test: Age Enter -4

Screenshot(s) of output:

```
Please enter your student number : 2306158

Pleases enter your Age : -4
please restart enter a Valid Number

please enter your House number : 60

please enter your street name : ChurchRoad

Student Number: 2306158
Student Age: -4
Address: 60 , ChurchRoad
```

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

YES



Test: Age Enter a letter

Screenshot(s) of output:

```
Please enter your student number : 2306158

Pleases enter your Age : a
please restart enter a Valid Number

please enter your House number :

please enter your street name : ChurchRoad

Student Number: 2306158
Student Age: -858993460
Address: a , ChurchRoad
```

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

Yes

The code succeeded in detecting false data and providing an error message However the faulty data caused a jump in the code, however the user was instructed to restart from the error so not an issue.

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.)

(fixed input validation but couldn't complete blank space validation with the time allotted, but i presume there must be a null string check function somewhere in the library)

Paste your revised code here:

```
#define _CRT_SECURE_NO_WARNINGS
```

```
#include <iostream>
```



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

```
#include <string.h>
```

```
int main()
{
    int StudentNumber = 99999999999999, Age;
    char HouseNumber[5], StreetName[30];

    printf("\n\nPlease enter your student number : ");
    scanf("%d", &StudentNumber);

    if (StudentNumber > 2999999 || StudentNumber < 2000000) {

        printf("please restart, enter a Valid Number");
    }

    printf("\n\nPlases enter your Age : ");
    scanf("%d", &Age);

    if (Age > 70 || Age <= 17) {
         6

        printf("please restart, enter a Valid Number");
    }
}
```

```
}
```

```
printf("\n\nplease enter your House number : ");
```

```
scanf("%s", &HouseNumber);
```

```
if (strlen(HouseNumber) > 4 ) {
```

```
    printf("please restart, enter a Valid Number");
```

```
}
```

```
printf("\n\nplease enter your street name : ");
```

```
scanf("%s", StreetName);
```

```
printf("\n\nStudent Number: %d\n Student Age: %d \n Address: %s , %s \n", StudentNumber,  
Age, HouseNumber, StreetName);
```

```
}
```

## **WORKSHEET 2a – TESTING (Circle/Triangle)**

1. Original code:

```
#define _CRT_SECURE_NO_WARNINGS
```

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

```
#include <math.h>
```

```
#include <string.h>
```

```
#include <stdlib.h>
```

```
#include <iostream>
```



```
float circleradiusA()
```

```
{
```

```
    float radius, area;
```

```
    scanf("%f", &radius);
```

```
    area = 3.14 * (radius * radius);
```

```
    return area;
```

```
}
```

```
float trianglearea()
```

```
{
```

```
    float area, Height, Base;
```

```
    scanf("%f %f", &Height,&Base);
```

```
    area = ((Base * Height) / 2);
```

```
    return area;
```

```
}
```

```
//float R1W, R1H, R1A;
```

```
int main()
```

```
{
```

```
    //find circle area usinf R
```

```
    // find trinagle area using H and B
```

```
    //take Traingle away from circle to get shaded area
```

```
    float circlearea1;
```

```
float triangleA;

float shadedarea;


printf("\nlet's find the area of the circle");
printf("\nplease enter the radius: ");
circlearea1 = circlearadiusA();
printf("the circles area is %f \n", circlearea1);


printf("\nplease enter the dimensions of the triangle\n");
printf("\nHeight then base please :");
triangleA = trianglearea();
printf("Triangle area : %f", triangleA);


shadedarea = (circlearea1 - triangleA);


printf("\n\nyour Shaded area is : %f", shadedarea);


}
```

## 2. Code Testing

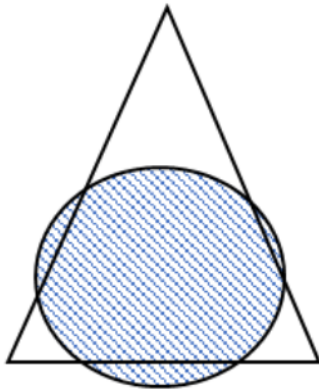
For each test below, run your code and enter the values provided. Provide a screenshot of the programme output. Did the programme behave correctly or did it do something unexpected? Fill in a box for each test below.

Input:

- i. Enter dimensions: Radius: 5
- ii. Enter dimensions: Radius: -5
- iii. Enter dimensions: Base: 4.5
- iv. Enter dimensions: Base: -18
- v. Enter dimensions: Height: 15.4
- vi. Enter dimensions Height: -3

Logic:

- i. Enter dimensions: Radius 10, Base 6, height 8. The result should be **290.2** (314.2 for the area of the circle minus 24 for the area of the triangle)
- ii. Enter dimensions: Radius 2, Base 6, height 8. The result is logically not possible. The base is overflowing the circle (approximation shown below). Does your program recognise and account for this?



Test:

- i) Input - Enter dimensions: Radius: 5

Screenshot(s) of output:

```
let's find the area of the circle
please enter the radius: 5
the circles area is 78.500000

please enter the dimensions of the triangle
Height then base please :5
5
Triangle area : 12.500000
your Shaded area is : 66.000000
```

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

YES

Test:

ii) Enter dimensions: Radius: - 5

```
let's find the area of the circle
please enter the radius: -5
the circles area is 78.500000

please enter the dimensions of the triangle

Height then base please :5
5
Triangle area : 12.500000

your Shaded area is : 66.000000
```

Screenshot(s) of output:

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

Yes  8

Test: Enter dimensions: Base: 4.5

```
let's find the area of the circle
please enter the radius: 5
the circles area is 78.500000

please enter the dimensions of the triangle

Height then base please :5 4.5
Triangle area : 11.250000

your Shaded area is : 67.250000
```

Screenshot(s) of output:

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

yes

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

Test: Base:      Enter dimensions: Base: -18

```
let's find the area of the circle
please enter the radius: 5
the circles area is 78.500000

please enter the dimensions of the triangle

Height then base please :5 -18
Triangle area : -45.000000

your Shaded area is : 123.500000
```

Screenshot(s) of output:

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

NO

Because you can't have a negative area so it doesn't make sense from an outcome oriented perspective.

Test: Enter dimensions: Height: 15.4

```
let's find the area of the circle
please enter the radius: 5
the circles area is 78.500000

please enter the dimensions of the triangle

Height then base please :15.4 5
Triangle area : 38.500000

your Shaded area is : 40.000000
```

Screenshot(s) of output:

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

YES

Test: Enter dimensions Height: -3

```
let's find the area of the circle
please enter the radius: 5
the circles area is 78.500000

please enter the dimensions of the triangle

Height then base please :-3 5
Triangle area : -7.500000

your Shaded area is : 86.000000
```

Screenshot(s) of output:

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

NO

Because you can't have a negative area so it doesn't make sense from an outcome oriented perspective.

Test: Enter dimensions: Radius 10, Base 6, height 8. The result should be **290.2**  
(314.2 for the area of the circle minus 24 for the area of the triangle)

Screenshot(s) of output:

```
let's find the area of the circle
please enter the radius: 10
the circles area is 314.000000

please enter the dimensions of the triangle

Height then base please :8 6
Triangle area : 24.000000

your Shaded area is : 290.000000
```

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

NO

As the output wasn't precise enough so i believe during data handling/presentation i must have clipped the last digits. Possibly when converting integers to floats or vice versa.

Test: Enter dimensions: Radius 2, Base 6, height 8. The result is logically not possible. The base is overflowing the circle (approximation shown below). Does your program recognise and account for this?

Screenshot(s) of output:

```
let's find the area of the circle
please enter the radius: 2
the circles area is 12.560000

please enter the dimensions of the triangle

Height then base please :8 6
Triangle area : 24.000000

your Shaded area is : -11.440000
```

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

NO

Code did not pick up the logical fallacy, and also presented a remaining are that is within the negatives

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 <math.h>

#include <string.h>

#include <stdlib.h>

#include <iostream>
```

```
float radius;
```

```
float circleradiusA()
{
    float area;

    scanf("%f", &radius);

    area = 3.14159265358979323846 * (radius * radius);

    return area;
}
```

```
float trianglearea()
{
    float area, Height=-1, Base=-1;
```

```
printf("\nIf values are no logically compatable then data will be re requested\n");
```

```
while( Height<0 || Base<0 || Base>(radius*2)){
```

```
    printf("\nplease enter the dimensions of the triangle\n");
```

```
    printf("\nHeight then base please :");
```

```
    scanf("%f %f", &Height, &Base);
```

```
}
```

```
area = ((Base * Height) / 2);
```

```
return area;
```

```
}
```

```
//float R1W, R1H, R1A;
```

```
int main()
```

```
{
```

```
    //find circle area usinf R
```

```
    // find trinagle area using H and B
```

```
//take Traingle away from circle to get shaded area
```

```
float circlearea1;
```

```
float triangleA = -10;
```

```
float shadedarea;
```

```
printf("\nlet's find the area of the circle");
```

```
printf("\nplease enter the radius: ");
```

```
circlearea1 = circleradiusA();
```

```
printf("the circles area is %0.1f \n", circlearea1);
```

```
triangleA = trianglearea();
```

```
printf("Triangle area : %0.1f", triangleA);
```

```
shadedarea = (circlearea1 - triangleA);
```

```
printf("\n\nyour Shaded area is : %0.1f", shadedarea);
```

```
}
```

### **WORKSHEET 3 – TESTING (Arrays)**

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*

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

iv. Double check the total+VAT price at the end, is it correct relative to the total price?

Test: i. Enter the 5 normal values as shown on the example

ie *pizza 3.45, chips 1.20, tea 1.35, coffee 1.82 and banana 0.17*

Screenshot(s) of output:

```
Welcome to the : SA1 Campus canteen!
```

```
pleases enter Item 1 Name : pizza
```

```
Please enter the price for Item 03.45
```

```
pleases enter Item 2 Name : chips
```

```
Please enter the price for Item 11.20
```

```
pleases enter Item 3 Name : tea
```

```
Please enter the price for Item 21.35
```

```
pleases enter Item 4 Name : coffee
```

```
Please enter the price for Item 31.82
```

```
pleases enter Item 5 Name : banana
```

```
Please enter the price for Item 40.17
```

```
Item 0 : pizza at 3.450000  
Item 1 : chips at 1.200000  
Item 2 : tea at 1.350000  
Item 3 : coffee at 1.820000  
Item 4 : banana at 0.170000
```

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

No

It works but the counter increments are incorrectly offset by 1 and needs to add up the total price of the items at the end and also the staff discount version below that.(%15)

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

Screenshot(s) of output:

```
Welcome to the : SA1 Campus canteen!

pleases enter Item 1 Name : pizza
Please enter the price for Item 0-2.34

pleases enter Item 2 Name : chips
Please enter the price for Item 1-1.10

pleases enter Item 3 Name : tea
Please enter the price for Item 2-1.22

pleases enter Item 4 Name : coffee -2.05
Please enter the price for Item 3

pleases enter Item 5 Name : banana -0.18
Please enter the price for Item 4

Item 0 : pizza at -2.340000
```

```
Item 1 : chips at -1.100000
```

```
Item 2 : tea at -1.220000
```

```
Item 3 : coffee at -2.050000
```

```
Item 4 : banana at -0.180000
```

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

NO

Accepted negative values so doesn't make sense and will need to be validated

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

Screenshot(s) of output:

```
Welcome to the : SA1 Campus canteen!
```

```
pleases enter Item 1 Name : pizza
```

```
Please enter the price for Item 0 2.50
```

```
pleases enter Item 2 Name :
```

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

NO

Couldn't validate blank inputs and wouldn't move on until a non blank input is given

Test: iv) Double check the total+VAT price at the end, is it correct relative to the total price?

Screenshot(s) of output:

```
Welcome to the : SA1 Campus canteen!
```

```
pleases enter Item 1 Name : pizza
```

```
Please enter the price for Item 03.45
```

```
pleases enter Item 2 Name : chips
```

```
Please enter the price for Item 11.20
```

```
pleases enter Item 3 Name : tea
```

```
Please enter the price for Item 21.35
```

```
pleases enter Item 4 Name : coffee
```

```
Please enter the price for Item 31.82
```

```
pleases enter Item 5 Name : banana
```

```
Please enter the price for Item 40.17
```

```
Item 0 : pizza at 3.450000
Item 1 : chips at 1.200000
Item 2 : tea at 1.350000
Item 3 : coffee at 1.820000
Item 4 : banana at 0.170000
```

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

NO

Program doesn't add up total at all nor the staff discounted version

1. 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 <stdlib.h>
```

```
int main()
```

```
{  
  
    char CanteenName[11] = { "SA1 Campus" };  
  
    float Prices[5];  
    char Names[5][30];  
    int i;  
    float Total, EmpDisc = 0.85;  
  
    printf("Welcome to the : %s canteen!\n\n", CanteenName);  
  
    for (i = 0; i < 5; i++) {  
  
        printf("\n\npleases enter Item %d Name : ", i + 1);  
        scanf("%s", &Names[i]);  
  
        printf("\nPlease enter the price for Item %d : ", i + 1);  
        scanf("%f", &Prices[i]);  
  
        while (Prices[i] < 0) {
```

```
        printf("\nPlease enter a positive price value");

        printf("\nPlease enter the price for Item %d : ", i + 1);

        scanf("%f", &Prices[i]);

    }

}

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

    printf("\n\n Item %d : %s at %f", i, Names[i], Prices[i]);

}

Total = (Prices[0] + Prices[1] + Prices[2] + Prices[3] + Prices[4]);

printf("\n\nyour Total is : %F", Total);

Total = (Total * EmpDisc);

printf("\n and with discount your total is : %f", Total);

}
```

## **WORKSHEET 4 – TESTING (Arrays)**

1. Original code:

```
#define _CRT_SECURE_NO_WARNINGS

#include <stdio.h>

#include <math.h>

#include <string.h>

#include <stdlib.h>

#include <time.h>

//#include "Header.h"


// use a random number generater to produce 3 seperate random numbers


//Use IF statements to compare instead potentially[

// compare the numbers to determine a likness value

// then a switch satement to then determin points

//]


// can have acumaltive or non acumulative score

// keeps asking to play until N is entered
```

```
int main()
{
    int rand1, rand2, rand3;

    srand(time(NULL));

    rand1 = RandomNo();

    rand2 = RandomNo();

    rand3 = RandomNo();

    if (rand1 == rand2 && rand2 == rand3);

}
```

```
int RandomNo()
{

    int randno;

    int low = 1;
```

```
int high = 5;

randno = rand() % (high - low + 1) + low;

printf("\nYour lucky number is = %d", randno);

return randno;

}
```

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.

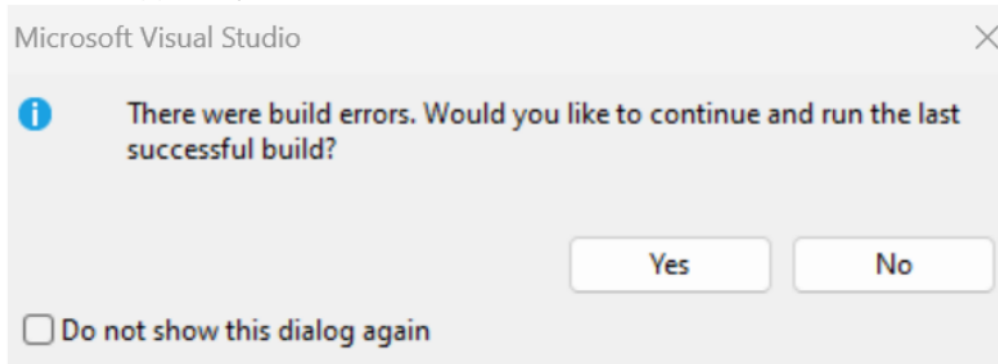
Test the following values for your fruit machine:

- 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:



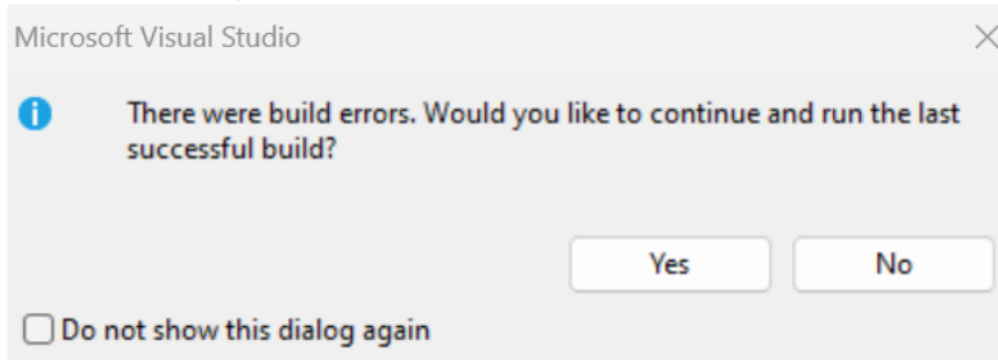
Result: Did the programme behave as expected ie syntactically and logically correct?

NO

Code cannot compile

Test: ii. 1 3 2 (0 points)

Screenshot(s) of output:



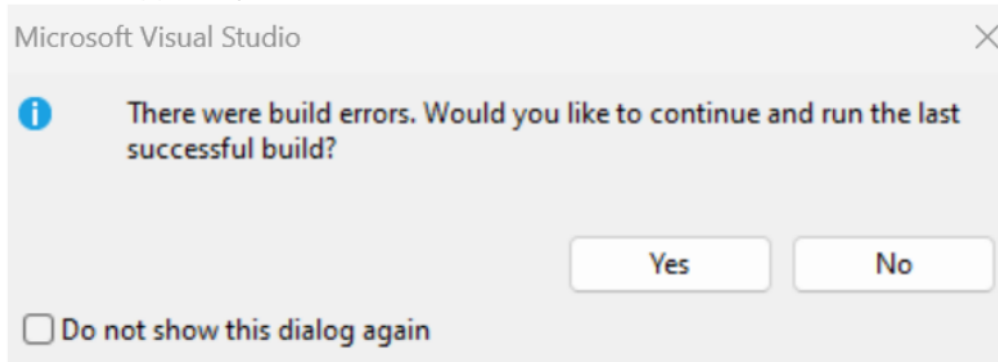
Result: Did the programme behave as expected ie syntactically and logically correct?

NO

Code cannot compile

Test: 5 5 3 (100 points)

Screenshot(s) of output:



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

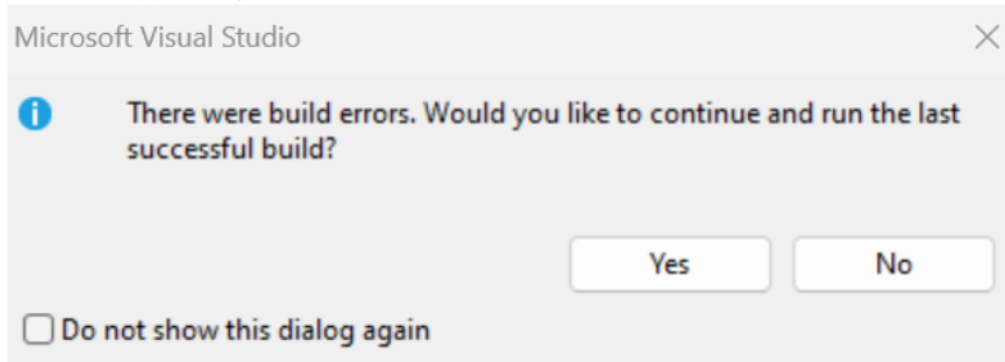
NO

Code cannot compile

Test: 5 2 5

(100 points)

Screenshot(s) of output:



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

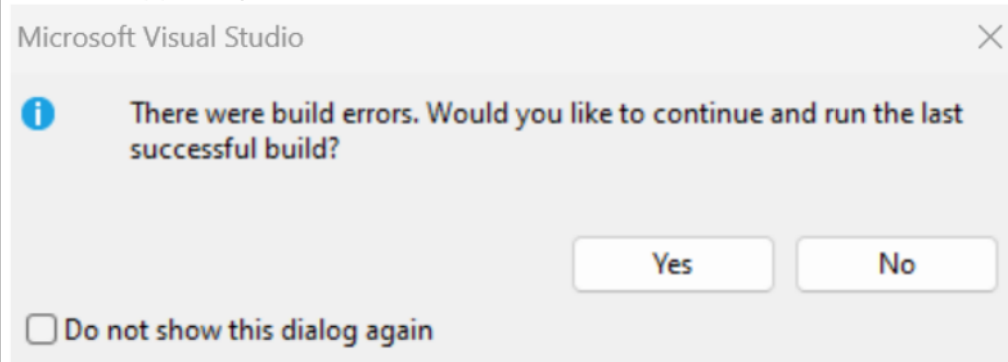
NO

Code cannot compile

Test: 1 5 5

(100 points)

Screenshot(s) of output:



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

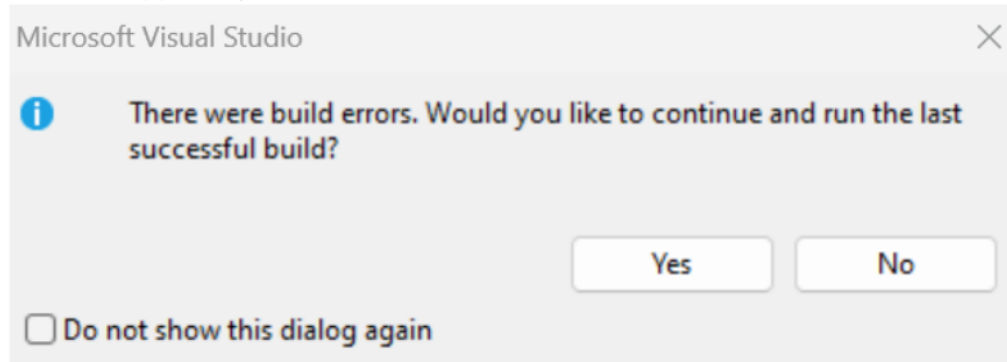
NO

Code cannot compile

Test: 5 5 5

(2000 points)

Screenshot(s) of output:



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

NO

Code cannot compile

1. 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:

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

```
#include <stdlib.h>
```

```
#include <time.h>
```

```
int spin() {
```

```
    return rand() % 5 + 1;
```

```
int calculatePoints(int reel1, int reel2, int reel3) {  
    int points = 0;  
  
    if (reel1 == reel2 && reel2 == reel3 && reel1 != 5) {  
        points += 20;  
    }  
    else if (reel1 == 5 && reel2 == 5 && reel3 != 5) {  
        points += 100;  
    }  
    else if (reel1 == 5 && reel2 == 5 && reel3 == 5) {  
        points += 2000;  
    }  
  
    return points;  
}
```

```
int main() {  
    srand((unsigned int)time(NULL));  
  
    int points = 0;  
    char playAgain = 'y';  
  
    printf("Welcome to the Fruit Machine Game!\n");
```

```
while (playAgain == 'y' || playAgain == 'Y') {  
    int reel1 = spin();  
    int reel2 = spin();  
    int reel3 = spin();  
  
    printf("\nSpinning... %d %d %d\n", reel1, reel2, reel3);  
  
    int spinPoints = calculatePoints(reel1, reel2, reel3);  
    points += spinPoints;  
  
    printf("Points scored in this spin: %d\n", spinPoints);  
    printf("Total points: %d\n", points);  
  
    printf("Do you want to spin again? (y/n): ");  
    scanf(" %c", &playAgain);  
}  
  
printf("Thank you for playing! Your final score is: %d\n", points);  
  
return 0;  
}
```

## **WORKSHEET 5 – TESTING (Treasure Hunt)**

Original code :

```
#define _CRT_SECURE_NO_WARNINGS
```

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

```
#include <math.h>
```

```
#include <string.h>
```

```
#include <stdlib.h>
```

```
#include <time.h>
```

```
int numbers[4][5];
```

```
int rows = sizeof(numbers) / sizeof(numbers[0]);
```

```
int columns = sizeof(numbers[0]) / sizeof(numbers[0][0]);
```

```
int RevealBoard() {
```

```
    numbers[0][0] = 0;
```

```
    numbers[0][1] = 0;
```

```
    numbers[0][2] = 0;
```

```
    numbers[0][3] = 0;
```

```
    numbers[0][4] = 0;
```

```
    numbers[1][0] = 0;
```

```
numbers[1][1] = 1;  
numbers[1][2] = 0;  
numbers[1][3] = 0;  
numbers[1][4] = 0;
```

```
numbers[2][0] = 0;  
numbers[2][1] = 0;  
numbers[2][2] = 0;  
numbers[2][3] = 0;  
numbers[2][4] = 0;
```

```
numbers[3][0] = 0;  
numbers[3][1] = 0;  
numbers[3][2] = 0;  
numbers[3][3] = 0;  
numbers[3][4] = 0;
```

```
for (int i = 0; i < rows; i++)  
{  
    for (int j = 0; j < columns; j++)  
    {  
        printf("%d ", numbers[i][j]);
```

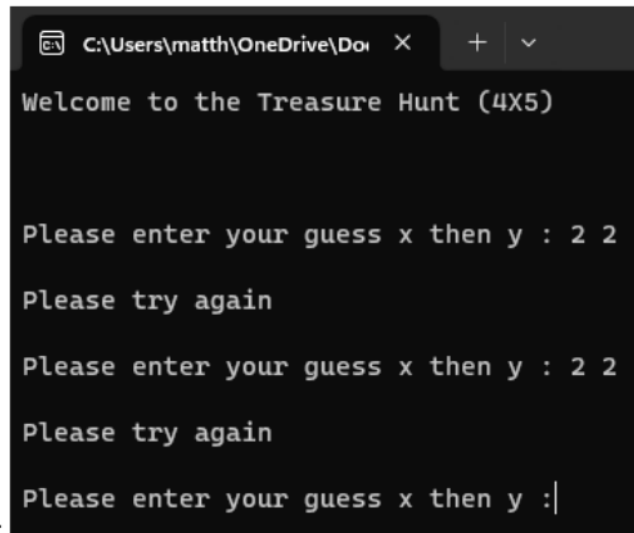
```
    }  
    printf("\n");  
}  
  
return 0;  
  
}  
  
int main(){  
  
    int Gx, Gy, win = -1;  
  
    printf("Welcome to the Treasure Hunt (4X5)\n\n");  
  
    while (win < 0) {  
  
        printf("\n\nPlease enter your guess x then y :");  
        scanf("%d %d", &Gx, &Gy);
```

```
    if (Gx == 1 && Gy == 1) {  
  
        RevealBoard();  
  
        win = 1;  
        printf("\nCongratulations you found the treasure");  
    }  
    else  
        printf("\nPlease try again");  
  
}  
  
}
```

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: Enter : 2 2



```
C:\Users\matth\OneDrive\Doi X + v
Welcome to the Treasure Hunt (4X5)

Please enter your guess x then y : 2 2
Please try again
Please enter your guess x then y : 2 2
Please try again
Please enter your guess x then y :|
```

Screenshot(s) of output:

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

YES

Test: Enter: 1 1

```
Welcome to the Treasure Hunt (4X5)

Please enter your guess x then y :1 1
0 0 0 0 0
0 1 0 0 0
0 0 0 0 0
0 0 0 0 0

Congratulations you found the treasure
```

Screenshot(s) of output:

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

YES

Test: Enter: -1 -1

```
Welcome to the Treasure Hunt (4X5)

Please enter your guess x then y :-1 -1
Please try again
Please enter your guess x then y :|
```

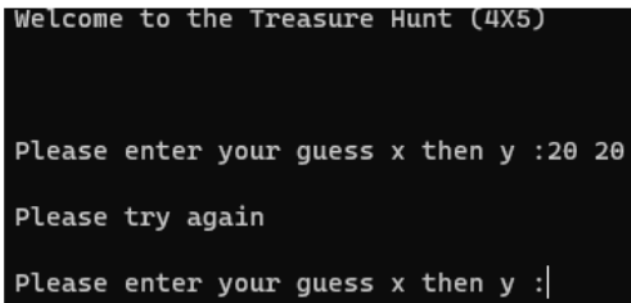
Screenshot(s) of output:

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

YES

Test: Enter: 20 20

Screenshot(s) of output:



```
Welcome to the Treasure Hunt (4X5)

Please enter your guess x then y :20 20

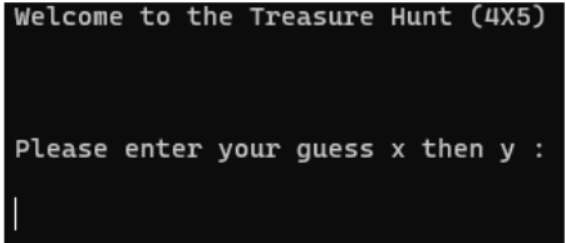
Please try again

Please enter your guess x then y :|
```

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

YES

Test: Enter: (blank input)

|                          |                                                                                                                                                                                                                                                              |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                          |                                                                                                                                                                                                                                                              |
| Screenshot(s) of output: |                                                                                                                                                                            |
|                          | <p>Result: Did the programme behave as expected ie syntactically and logically correct?</p> <p>NO</p> <p>Unable to detect blank input but will still read if one is eventually put in, however does not prompt the user to do so after the blank entries</p> |

# Matthew Coleman(2306158) Assignment testing 1-5

---

## GRADEMARK REPORT

---

FINAL GRADE

/10

GENERAL COMMENTS

Why have you uploaded everything in one tab? I set up 5 tabs for 5 individual worksheets.

### Worksheet 1:

**Coding:** A reasonable attempt, but you could have made more progress to preventing logic errors. For example, you could use a while until an age in a suitable range is entered

2.5/5

### Testing:

See my inline comments. Sometimes you have suggested that tests are passed when they are not. Failures can be logical as well as syntax errors

3/5

**Total:** 5.5/10

-----

### Worksheet 2:

**Coding:** Good progress made on the relative validation of square and circle size. Your formula could have been a bit more refined.

4 /5

**Testing:** Generally good 4/5

**Total:** 8/10

-----

**Worksheet 3:**

**Coding:**

Good progress on the negative number validation, see my comment below about letters/numbers. You could revise for efficiency. You need to loop total = price[0]+price[1]+price[2]+price[3]+price[4] rather than hard code.

7/10

**Testing:**

There were a few deficiencies you found in the code. You could also have considered what happens when letters/numbers are entered in the wrong place, eg letters for a price.

7/10

**Total:** 14/20

-----

**Worksheet 4:**

**Coding:**

Function work is puzzling. There's really no advantage in pushing your 3 values to another function, it doesn't save you anything.

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, a decent attempt overall. 6/10

/10

### **Testing:**

Next time, ask me to help you to the nearest working version of your code - there's no real point testing code that does not run.

2/10

**Total: 8/20**

-----

### **Worksheet 5:**

#### **Coding:**

A good attempt. A bit inefficient where you set up the numbers, you could have looped as you have to display the board - I'm a bit confused because surely you are resetting the board each time? You could have the part where you print the board in another function, so you can reuse it. It would be nice to see the board between turns. Also, watch your validation, you allow the user to enter number higher than the size of the board. It would also be nice if it was randomised so it isn't the same game each time!

8/15

**Testing:**

Some testing you've said is correct but isn't eg -1 and -1 are not positions in your array. The program doesn't break, but you can make it unstable by allowing out of range values. Could be a little more extensive.

9/15

**Total:** 17/30

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

Could be improved - for example, there is a lot of whitespace where your code is pasted which makes it difficult to read, always try to tidy things up to make things more presentable. In this case you could adjust the line spacing options within Word.

Be careful not to use personal pronouns like "I" or "you", keep the writing style more formal in academic reports - for example "Because you can't have a negative area" -> "The area of a triangle cannot be negative"

Also, you could number the pages, put your name at the top etc.

5/10

**Overall total for assignment:** 57.5%

**Test mark:** 63%

**Overall mark for module:** 59%

Keep practising between now and Year 2 to maintain/improve your skills.

---

PAGE 1

---

PAGE 2

---

PAGE 3

---

PAGE 4

---

PAGE 5

---

PAGE 6

---



### Comment 1

Not really - a minus number for a student number isn't logically suitable

PAGE 7

---

PAGE 8

---

PAGE 9

---



### Comment 2

Not really wrong - a blank house number would not be desirable

PAGE 10

---

PAGE 11

---

PAGE 12

---



### Comment 3

Is an age of 150 logically suitable? You should validate - just saying "restart" is not really sufficient, you should keep them at that point in the program until a suitable number is entered



### Comment 4

Is an age of -4 logically suitable?



### Comment 5

This is a C++ library, not C



### Comment 6

Use a while here



### Comment 7

C++ not C



## Comment 8

No, -5 is not logically suitable

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41

PAGE 42

PAGE 43

PAGE 44

PAGE 45

PAGE 46

PAGE 47

PAGE 48

PAGE 49

PAGE 50

PAGE 51

PAGE 52

PAGE 53

PAGE 54

PAGE 55

PAGE 56

PAGE 57

PAGE 58

PAGE 59

PAGE 60

PAGE 61

PAGE 62