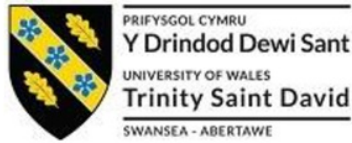


Programs in C

by David Isaac

FILE	13188_DAVID_ISAAC_PROGRAMS_IN_C_173935_557756758.DOCX (437.53K)		
TIME SUBMITTED	05-MAY-2019 01:52PM (UTC+0100)	WORD COUNT	2236
SUBMISSION ID	106340395	CHARACTER COUNT	11339



University of Wales Trinity Saint David

School of Applied Computing

Programs in C and documentation

Introduction to Computer Programming

David Isaac

1051176

18/03/2019

Table of Contents

Introduction	3
Tasks.....	4
Part A.....	4
Part B.....	5
Exercise 1: Variables	5
Exercise 2: Simple calculations	9
Exercise 3: Cost calculator	12
Exercise 4: Fruit Machine.....	14
Conclusions	17
References	18

Introduction

In this assignment a number of programmes are to be written demonstrating ability to program using variables effectively, the use of simple math to make calculations, use of arrays and to be able to generate random numbers and use conditions such as if statements.

1. Exercise 1 – Use of variables
2. Exercise 2 – Making simple calculations
3. Exercise 3 – Listing prices and more calculations
4. Exercise 4 – Make and meeting conditions (if statement), generating random numbers

Tasks

Part A

1. What is a variable and why are they important in writing programs?
 - a. [1] The oxford dictionary defines a variable as: Not consistent or having a fixed pattern; liable to change. In regards to computer programming a variable is a value given a name that can be changed as per the user's needs. Examples of various variable are:
 - i. Float – numeric value using decimal point.
 - ii. Int – numeric value whole numbers only.
 - iii. Char- single letter value
2. Explain why the for-loop can also be referred to as a counting-loop and give an example of when you would use it.

A for-loop can be referred to as a counting loop because the variable use is a whole number (int) and depending on the values assigned the for command this will create a countdown/up. [2] This integer known as a loop control variable. This can be use to count the number of times a value is input. E.g. the number of items a shopper buys as the price is input.

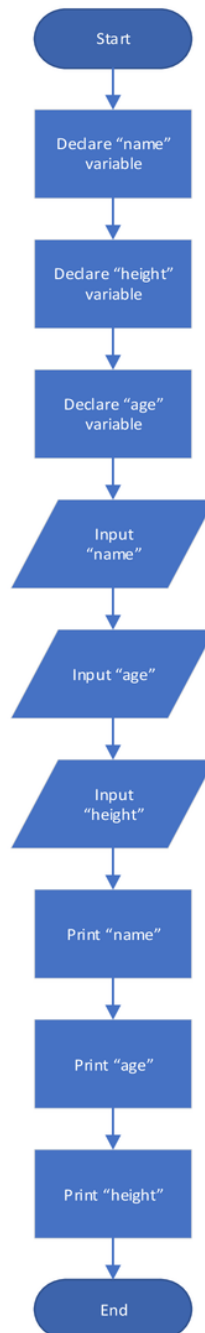
3. Explain what an array is, what their features are, and give an example of when you would use one.

An array is a list of variables attached to a single variable name. This can be used to link similar types of data together under one variable. E.g. an array could hold the 12 months of the year under a variable called 'months'.

Part B

Exercise 1: Variables

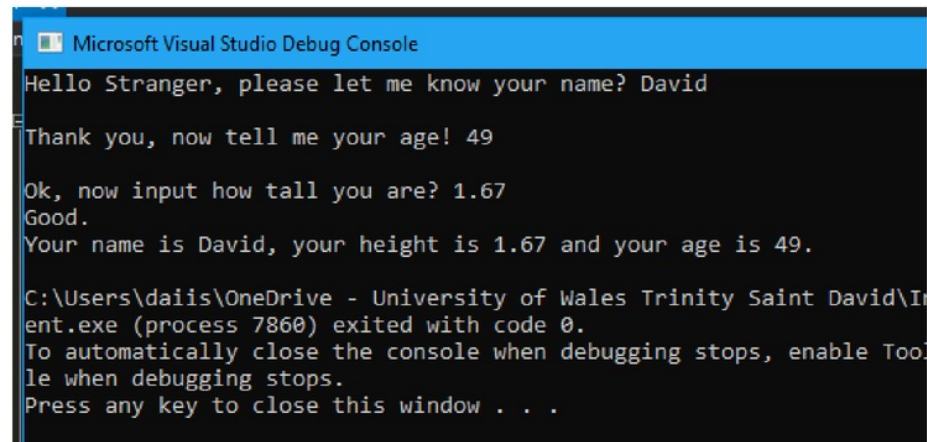
This exercise is designed to show how variables are used. Three variables have been chosen, these are numeric, numeric with floating point (decimal point) and characters.



The program below demonstrates the use of variables.

```
// Exercise 1
#include<stdio.h>
void main()
{
    int age; //whole number variable
    float height; //floating point variable
    char name[15]; //character variable using string to hold a name
    printf("Hello Stranger, please let me know your name? ");
    scanf("%s", &name); //user input the character string
    printf("\nThank you, now tell me your age! ");
    scanf("%d", &age); //user input the integer for age
    printf("\nOk, now input how tall you are? ");
    scanf("%f", &height); //user input the floating point variable for height
    printf("Good.");
    printf("\nYour name is %s, your height is %0.2f and your age is %d.\n", name, height, age);
    // %s calls the character string, %0.2f calls the float variable, %d calls the integer variable
}
```

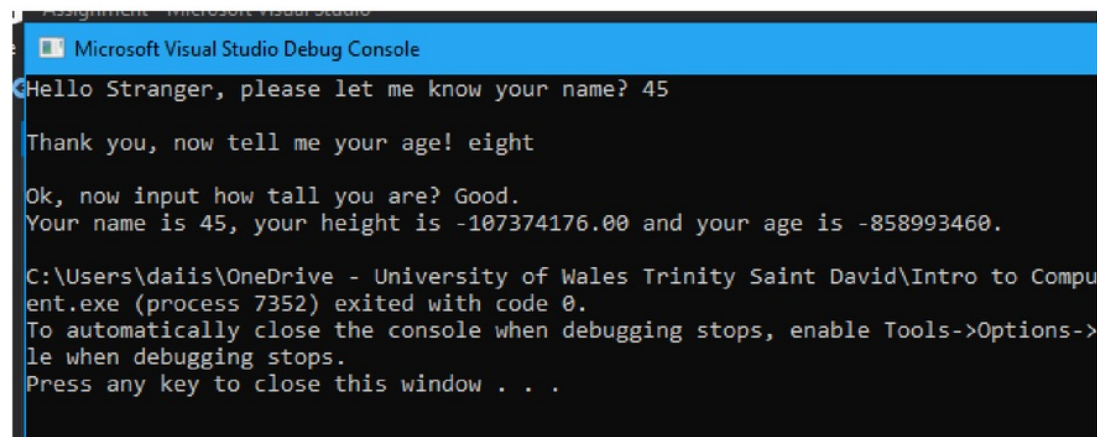
The following screen shot shows the program working with the correct type of data input. The programmer has no constraints on the data to be input.



```
Microsoft Visual Studio Debug Console
Hello Stranger, please let me know your name? David
Thank you, now tell me your age! 49
Ok, now input how tall you are? 1.67
Good.
Your name is David, your height is 1.67 and your age is 49.

C:\Users\daiis\OneDrive - University of Wales Trinity Saint David\Intro to Computing\ent.exe (process 7860) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Tool window when debugging stops.
Press any key to close this window . . .
```

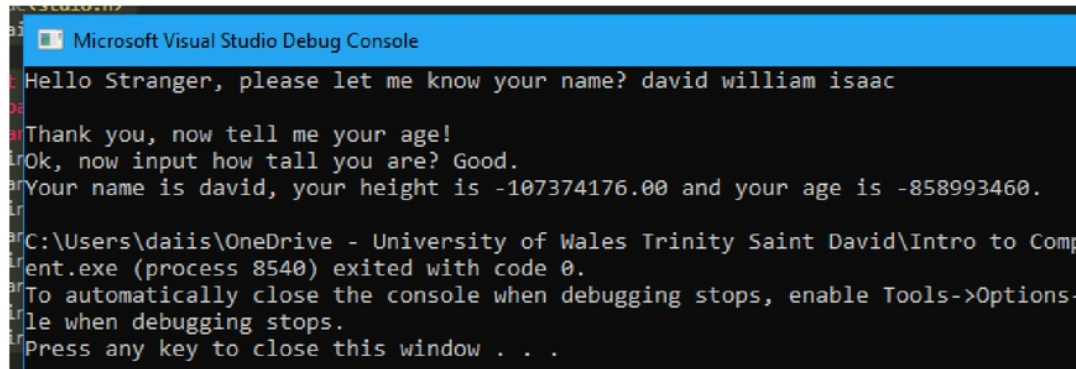
Exercise 1 program is a simple program that has no constraints which allows the user to input any data when the scanf's call for data input. This could cause issues as the program will not work correctly if the user inputs data that is not valid. This is demonstrated below in the screenshots.



```
Microsoft Visual Studio Debug Console
Hello Stranger, please let me know your name? 45
Thank you, now tell me your age! eight
Ok, now input how tall you are? Good.
Your name is 45, your height is -107374176.00 and your age is -858993460.

C:\Users\daiis\OneDrive - University of Wales Trinity Saint David\Intro to Computing\ent.exe (process 7352) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Tool window when debugging stops.
Press any key to close this window . . .
```

In the screenshot above the user has input a numeric value for the first variable, this should be constrained to letters only. The second variable the programmer wanted the user to input a numeric value, but the user has input the number as characters. This caused the program to end prematurely.



```
Microsoft Visual Studio Debug Console
Hello Stranger, please let me know your name? david william isaac
Thank you, now tell me your age!
Ok, now input how tall you are? Good.
Your name is david, your height is -107374176.00 and your age is -858993460.
C:\Users\daiis\OneDrive - University of Wales Trinity Saint David\Intro to Comp
ent.exe (process 8540) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options-
le when debugging stops.
Press any key to close this window . . .
```

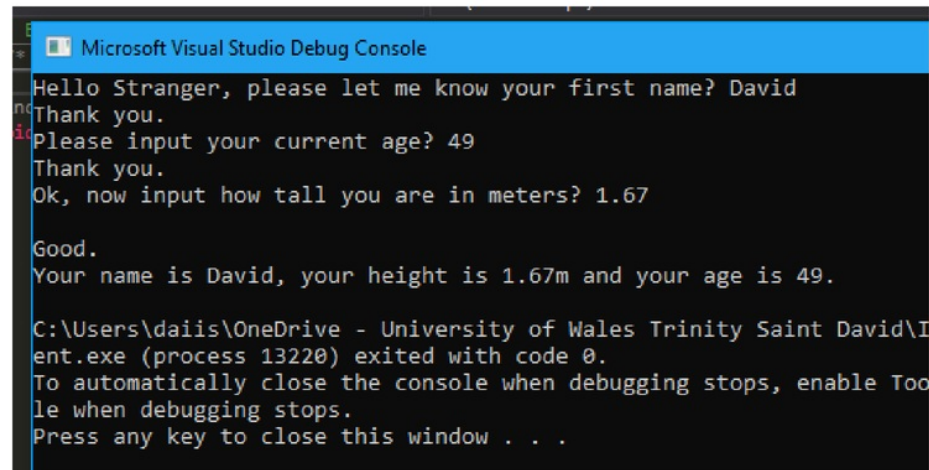
The above screenshot shows how the user has input their full name when the programmer had intended for first name only. This has caused the program to finish prematurely as well.

The program below is a modification on the first program designed to help prevent incorrect data being input.

```
#include<stdio.h>
void main()
{
    int age = 0; //The while statement below needs the variables to be initialised before it will
work
    float height = 0; //By initialising the variables outside the ranges given in the while
statements the program works
    char name[30]; //character variable using string to hold a name
    printf("Hello Stranger, please let me know your first name? ");
    scanf("%s", &name); //user input the character string

    printf("Thank you.");
    while (age < 1 || age>120)
    {
        printf("\nPlease input your current age? ");
        scanf("%d", &age); //user input the integer for age
    }
    printf("Thank you.");
    while (height<0.5 || height>2.4)
    {
        printf("\nOk, now input how tall you are in meters? ");
        scanf("%f", &height); //user input the floating point variable for height
    }
    printf("\nGood.");
    printf("\nYour name is %s, your height is %0.2fm and your age is %d.\n", name, height, age);
    //%s calls the character string, %0.2f calls the float variable, %d calls the integer variable
}
```

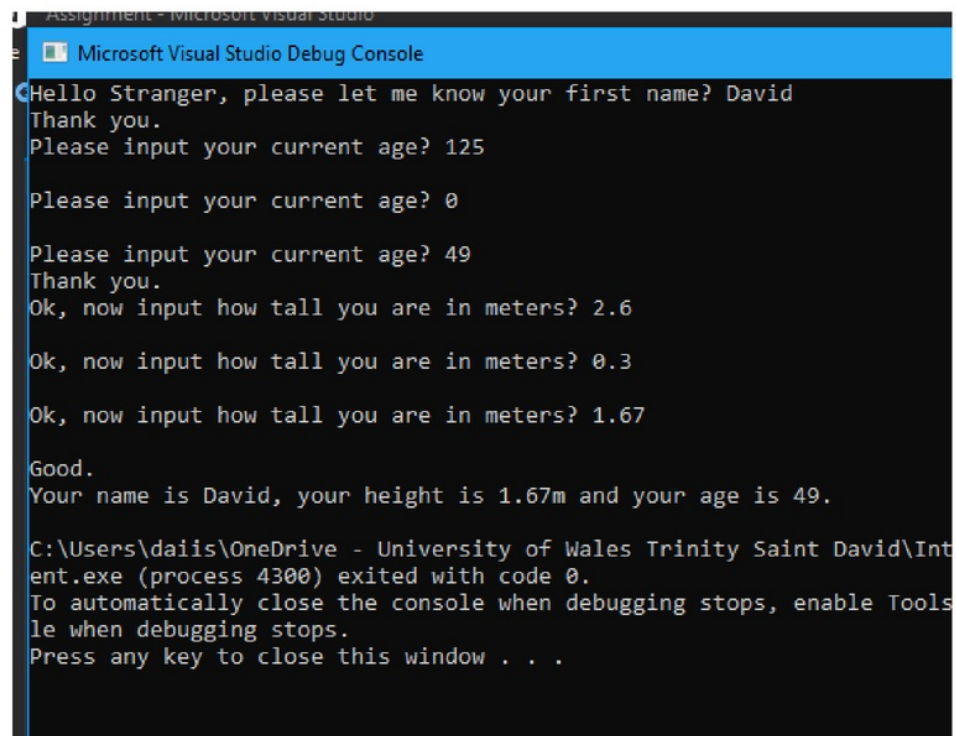
The following screenshots show the program working as intended and what happens when incorrect data gets input.



```
Microsoft Visual Studio Debug Console
Hello Stranger, please let me know your first name? David
Thank you.
Please input your current age? 49
Thank you.
Ok, now input how tall you are in meters? 1.67

Good.
Your name is David, your height is 1.67m and your age is 49.

C:\Users\daiis\OneDrive - University of Wales Trinity Saint David\Int
ent.exe (process 13220) exited with code 0.
To automatically close the console when debugging stops, enable Too
le when debugging stops.
Press any key to close this window . . .
```



```
Assignment - Microsoft Visual Studio
Microsoft Visual Studio Debug Console
Hello Stranger, please let me know your first name? David
Thank you.
Please input your current age? 125

Please input your current age? 0

Please input your current age? 49
Thank you.
Ok, now input how tall you are in meters? 2.6

Ok, now input how tall you are in meters? 0.3

Ok, now input how tall you are in meters? 1.67

Good.
Your name is David, your height is 1.67m and your age is 49.

C:\Users\daiis\OneDrive - University of Wales Trinity Saint David\Int
ent.exe (process 4300) exited with code 0.
To automatically close the console when debugging stops, enable Tools
le when debugging stops.
Press any key to close this window . . .
```

Exercise 2: Simple calculations

This exercise is designed to show how simple math is used within a program.

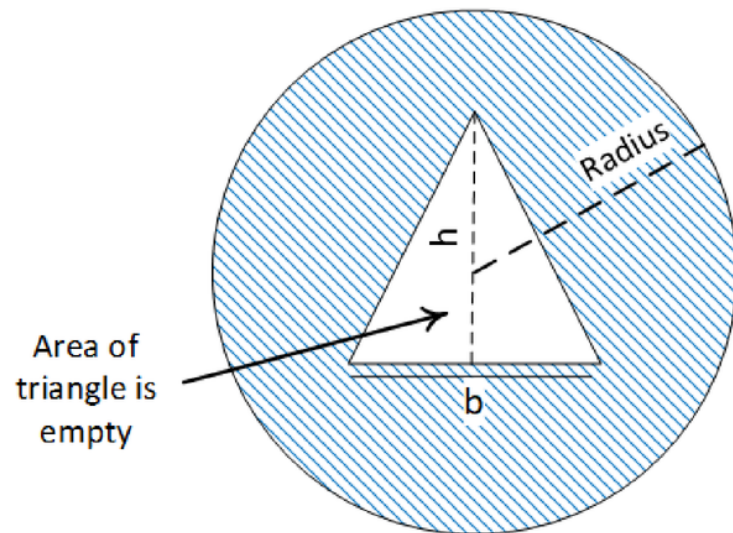
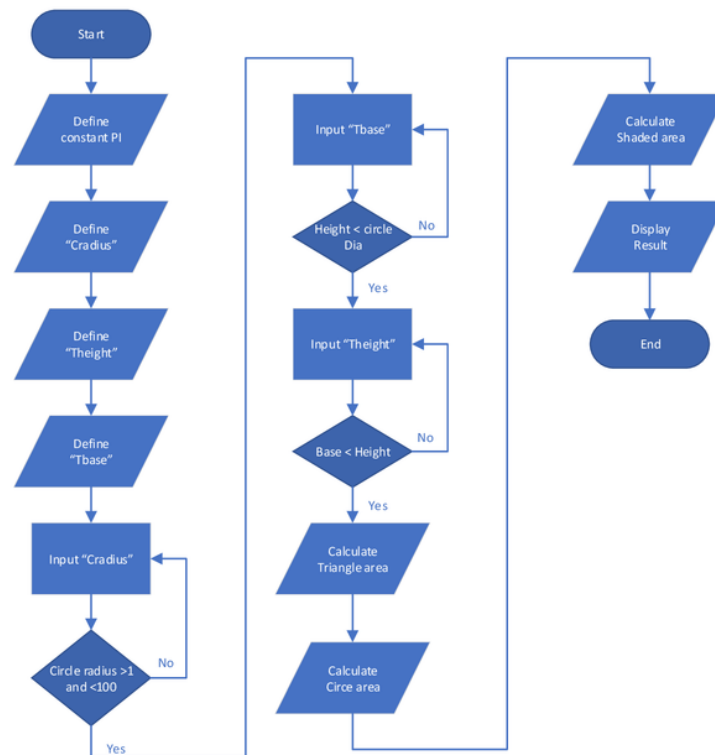


Figure 1 Calculate Shaded Area [3]

This program allows the user to input the dimensions of the triangle and circle and will then calculate the area of the shaded area.

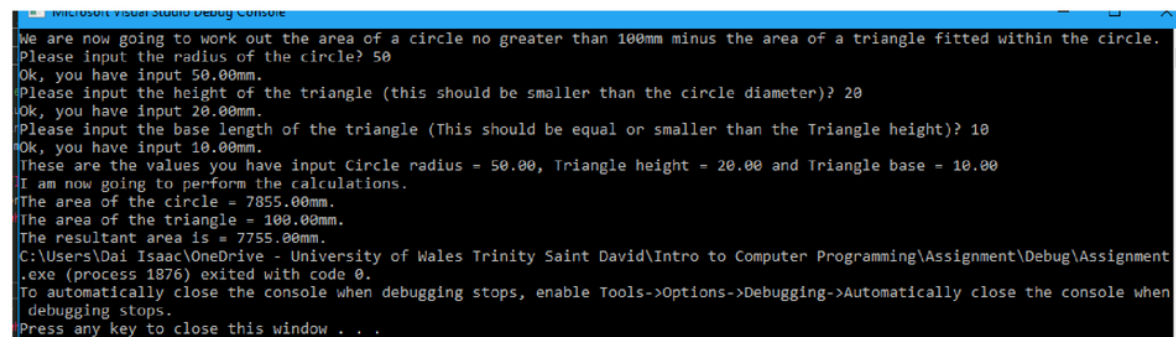


```

#include<stdio.h>
#define PI 3.142 //define a constant (this is a value that cannot be altered while the program runs)
void main()
{
    float Cradius = 0, Theight = 0, Tbase = 0; //define and initialise the required variables
    printf("We are now going to work out the area of a circle no greater than 100mm minus the area of a triangle fitted within the circle.");
    while (Cradius < 1 || Cradius>100) //Sets the circle dimensions to be between 1-100
    {
        printf("\nPlease input the radius of the circle? ");
        scanf("%f", &Cradius); //input the circle radius
        printf("Ok, you have input %0.2fmm.", Cradius);
    }
    while (Theight<1 || Theight>=(Cradius * 2)+1) //sets the limits for the triangle to be smaller than the circle
    {
        printf("\nPlease input the height of the triangle (this should be smaller than the circle diameter)? ");
        scanf("%f", &Theight); //input the triangle height
        printf("Ok, you have input %0.2fmm.", Theight);
    }
    while (Tbase<1 || Tbase>Theight) //sets the triangles base limit to be equal or smaller than the triangle height
    {
        printf("\nPlease input the base length of the triangle (This should be equal or smaller than the Triangle height)? ");
        scanf("%f", &Tbase); //input the triangle base length
        printf("Ok, you have input %0.2fmm.", Tbase);
    }
    printf("\nThese are the values you have input Circle radius = %0.2f, Triangle height = %0.2f and Triangle base = %0.2f", Cradius, Theight, Tbase);
    printf("\nI am now going to perform the calculations.");
    float Carea, Tarea, result; //adds the variables that are required to hold the calculations results
    Carea = PI * (Cradius*Cradius); //calculation to work out the circle area
    printf("\nThe area of the circle = %0.2fmm.", Carea);
    Tarea = (Tbase*Theight) / 2; //calculation to work out the triangle area
    printf("\nThe area of the triangle = %0.2fmm.", Tarea);
    result = Carea - Tarea; //calculation to work out the required area
    printf("\nThe resultant area is = %0.2fmm.", result); //displays the final required answer
}

```

The following screenshot displays the program working when all the data is correctly input



```

Microsoft Visual Studio Debug Console
We are now going to work out the area of a circle no greater than 100mm minus the area of a triangle fitted within the circle.
Please input the radius of the circle? 50
Ok, you have input 50.00mm.
Please input the height of the triangle (this should be smaller than the circle diameter)? 20
Ok, you have input 20.00mm.
Please input the base length of the triangle (This should be equal or smaller than the Triangle height)? 10
Ok, you have input 10.00mm.
These are the values you have input Circle radius = 50.00, Triangle height = 20.00 and Triangle base = 10.00
I am now going to perform the calculations.
The area of the circle = 7855.00mm.
The area of the triangle = 100.00mm.
The resultant area is = 7755.00mm.
C:\Users\Dai Isaac\OneDrive - University of Wales Trinity Saint David\Intro to Computer Programming\Assignment\Debug\Assignment.exe (process 1876) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .

```

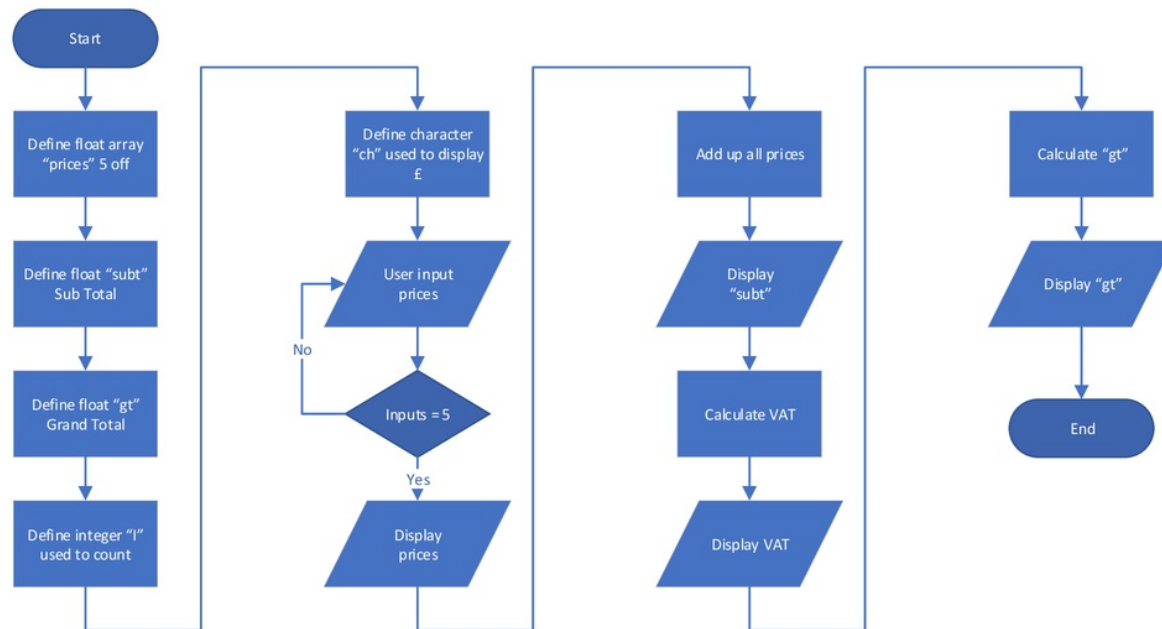
The screenshot below shows what happens when the data is input and outside the ranges that need to be input.

```
We are now going to work out the area of a circle no greater than 100mm minus the area of a triangle fitted within the circle.
Please input the radius of the circle? 0
Ok, you have input 0.00mm.
Please input the radius of the circle? 101
Ok, you have input 101.00mm.
Please input the radius of the circle? 50
Ok, you have input 50.00mm.
Please input the height of the triangle (this should be smaller than the circle diameter)? 101
Ok, you have input 101.00mm.
Please input the height of the triangle (this should be smaller than the circle diameter)? -1
Ok, you have input -1.00mm.
Please input the height of the triangle (this should be smaller than the circle diameter)? 40
Ok, you have input 40.00mm.
Please input the base length of the triangle (This should be equal or smaller than the Triangle height)? 41
Ok, you have input 41.00mm.
Please input the base length of the triangle (This should be equal or smaller than the Triangle height)? 30
Ok, you have input 30.00mm.
These are the values you have input Circle radius = 50.00, Triangle height = 40.00 and Triangle base = 30.00
I am now going to perform the calculations.
The area of the circle = 7855.00mm.
The area of the triangle = 600.00mm.
The resultant area is = 7255.00mm.
C:\Users\Dai Isaac\OneDrive - University of Wales Trinity Saint David\Intro to Computer Programming\Assignment\Debug\Assignment
```

It can be seen that if the data is too small or too big then the program continues to ask for data that fits within the constraints.

Exercise 3: Cost calculator

This exercise demonstrates the ability to input values, calculate the sum of these values and then work out a percentage (VAT) and add them up.



The following screenshot displays the program working when the correct type of data is input.

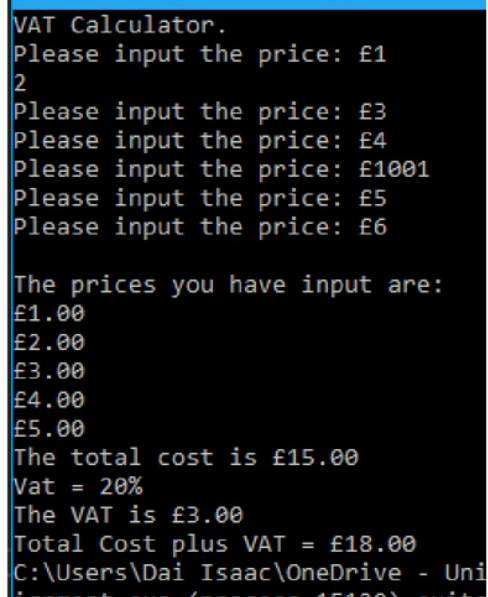
```
Microsoft Visual Studio Debug Console
VAT Calculator.
Please input the price: £1.23
Please input the price: £2.34
> Please input the price: £3.45
Please input the price: £4.56
Please input the price: £5.67
[
The prices you have input are:
£1.23
£2.34
£3.45
£4.56
£5.67
The total cost is £17.25
Vat = 20%
The VAT is £3.45
Total Cost plus VAT = £20.70
C:\Users\Dai Isaac\OneDrive - University of Wales
ignment.exe (process 10328) exited with code 0.
To automatically close the console when debugging
le when debugging stops.
Press any key to close this window . . .
```

The code for the previous screenshot:

```
#include<stdio.h>
void main()
{
    float prices[5], subt, gt; //sets up the float variables
    int i; //sets up the counter for inputting the prices
    char ch = 156; //sets up the variable to allow the program to display the £ symbol
    printf("VAT Calculator.\n");
    for (i = 0; i < 5; i++) //the counter to input the prices
    {
        while (prices[i] < 0 || prices[i] > 1000) //constrains the input values to between 0 -
1000
        {
            printf("Please input the price: %c", ch);
            scanf("%f", &prices[i]);
        }
        printf("\nThe prices you have input are: "); //confirming the prices that have been input
        for (i = 0; i < 5; i++)
        {
            printf("\n%c%0.2f", ch, prices[i]);
        }
        subt = prices[0] + prices[1] + prices[2] + prices[3] + prices[4]; //adds all the prices
together
        printf("\nThe total cost is %c%0.2f", ch, subt); //display the total price
        gt = (subt / 100) * 20; //calculates the vat
        printf("\nVat = 20%");
        printf("\nThe VAT is %c%0.2f", ch, gt); //displays the vat
        gt = gt + subt; //adds the vat to the subtotal
        printf("\nTotal Cost plus VAT = %c%0.2f", ch, gt); //displays the grand total
    }
}
```

[4] Accessed this web site to learn how to display the £ symbol

The below screenshot shows that the values input has been constrained, so if the user inputs any values outside this range then it will continue to ask for a value



```
VAT Calculator.
Please input the price: £1
2
Please input the price: £3
Please input the price: £4
Please input the price: £1001
Please input the price: £5
Please input the price: £6

The prices you have input are:
£1.00
£2.00
£3.00
£4.00
£5.00
The total cost is £15.00
Vat = 20%
The VAT is £3.00
Total Cost plus VAT = £18.00
C:\Users\Dai Isaac\OneDrive - Uni
```

Exercise 4: Fruit Machine

In this exercise the program to be written should generate 3 random numbers then assign a score based on the numbers generated as follows:

1. Any two 5's – 20 points
2. Any three numbers the same but not 5's – 100 points
3. Three 5's – 2000 points

The program should then display any winnings and also a running total of multiple spins.

The program below is a Jackpot game, it uses 3 variables to hold the current value of the 3 "wheels". With 2 variables holding the winning score and the running total. While command is used to continue play if y selected. It then generates three random numbers, then a number of if statements are checked to see if the random numbers meet the criterion for any of the 3 wins.

```
#include<stdio.h>
#include<stdlib.h> //calls the library in to be able to use rand command
#include<time.h> //this is needed to produce random number every time
void main()
{
    srand(time(NULL)); //resets the random number
    int Lnumber, Mnumber, Rnumber, pTotal = 0, pRtotal = 0; //sets up the variables
    char play = 'n', spin = 'n'; //sets up and declares variables used to ask for another go
    printf("Do you want to play JACKPOT? (y/n)? ");
    scanf(" %c", &play); //ask the play if they want to play

    if (play == 'y') //when player selects y this part of the program works
    {
        printf("\nOk spin the wheels! (y/n)? ");
        scanf(" %c", &spin); //ask the play to spin the wheel

        while (spin == 'y') //this part of the program works when the player selects y
        {
            Lnumber = rand() % (9); //generates a random number and assigns it the this
            Mnumber = rand() % (9); //generates a random number and assigns it the this
            Rnumber = rand() % (9); //generates a random number and assigns it the this

            printf("Ok.....here we go.....the numbers are....\n");
            printf(" %d %d %d\n", Lnumber, Mnumber, Rnumber); //Displays the 3 random
            numbers

            if (Lnumber == Mnumber && Lnumber == Rnumber && Lnumber != 5) //what happens
            when three of the same numbers are rolled that do not = 5
            {
                pTotal = 100; //awards a score
                printf("\nYay! You win 100 points"); // displays win
                pRtotal = pRtotal + pTotal; //adds score to running total
                printf("\nYour current score is now %d.", pRtotal); //displays running
                total
            }
            else if (Lnumber == Mnumber && Lnumber == Rnumber && Lnumber == 5) //what
            happens when three 5's generated
            {
                pTotal = 2000; //awards score
                printf("\nWhoo Hoo You scored the JACKPOT of 2000 points"); //displays
                score

                pRtotal = pRtotal + pTotal; //adds score to running total
                printf("\nYour current score is now %d.", pRtotal); //displays total
                score
            }
        }
    }
}
```

```

Don't need Lnumber!=5, 3 5s have been tested for
else if (Lnumber != 5 && Mnumber == 5 && Rnumber == 5) //2 5's routine Middle
Right
{
    pTotal = 20;
    printf("Oh look you won 20 points");
    pRtotal = pRtotal + pTotal;
    printf("\nYour current score is now %d.", pRtotal);
}
else if (Lnumber == 5 && Mnumber != 5 && Rnumber == 5) //2 5's routine Left
Right
{
    pTotal = 20;
    printf("Oh look you won 20 points");
    pRtotal = pRtotal + pTotal;
    printf("\nYour current score is now %d.", pRtotal);
}
else if (Lnumber == 5 && Mnumber == 5 && Rnumber != 5) //2 5's routine Left
middle
{
    pTotal = 20;
    printf("Oh look you won 20 points");
    pRtotal = pRtotal + pTotal;
    printf("\nYour current score is now %d.", pRtotal);
}

printf("\nSpin again? (y/n)? ");
scanf(" %c", &spin); //ask if want to spin again
}

printf("\nOk! Your final score is now %d", pRtotal); //End game display final score
}

printf("\n\nTHE END!!");
}

```

The following screenshots display the program working:

```

C:\Users\Dai Isaac\OneDrive - University of Wales Trinity Saint Davi
Do you want to play JACKPOT? (y/n)? y
Ok spin the wheels! (y/n)? y
Ok.....here we go.....the numbers are....
 6 5 3

Spin again? (y/n)? y
Ok.....here we go.....the numbers are....
 8 1 7

Spin again? (y/n)? y
Ok.....here we go.....the numbers are....
 4 4 2

Spin again? (y/n)? y
Ok.....here we go.....the numbers are....
 4 4 4

Yay! You win 100 points
Your current score is now 100.
Spin again? (y/n)?

```

```
Spin again? (y/n)? y
Ok.....here we go.....the numbers are....
 1 5 5
Oh look you won 20 points
Your current score is now 120.
Spin again? (y/n)?
```

```
Spin again? (y/n)? y
Ok.....here we go.....the numbers are....
 5 5 2
Oh look you won 20 points
Your current score is now 140.
Spin again? (y/n)?
```

```
Spin again? (y/n)? y
Ok.....here we go.....the numbers are....
 5 0 5
Oh look you won 20 points
Your current score is now 160.
Spin again? (y/n)?
```

```
Spin again? (y/n)? y
Ok.....here we go.....the numbers are....
 5 5 5

Whoo Hoo You scored the JACKPOT of 2000 points
Your current score is now 2960.
Spin again? (y/n)? y
Ok.....here we go.....the numbers are
```

```
Spin again? (y/n)? n

Ok! Your final score is now 2960

THE END!!
C:\Users\Dai Isaac\OneDrive - Univers
ignment.exe (process 920) exited with
```

Conclusions

References

- [1] Oxford Dictionaries “English Oxford Living Dictionaries”, [Online]
<https://en.oxforddictionaries.com/definition/variable>. [Accessed Mar. 18, 2019].
- [2] Chortle, “assembly tutorial”, [Online] Available: http://chortle.ccsu.edu/assemblytutorial/Chapter-18/ass18_10.html. [Accessed Apr. 5, 2019].
- [3] University of Wales Trinity Saint David, “Faculty of Applied Computing” UWTSd 2019. [Online]. Available: <https://moodle.uwtsd.ac.uk/course/view.php?id=6804>. [Accessed Apr. 5, 2019].
- [4] Go4expert “Forums”, 2008. [Online] <https://www.go4expert.com/forums/i-display-sign-printf-t16928/>. [Accessed Apr. 5, 2019].
- [5] M. McGrath, *C Programming*, 4th Edition, Leamington Spa: In Easy Steps Limited, 2014.

Programs in C

GRADEMARK REPORT

FINAL GRADE

73/100

GENERAL COMMENTS

Instructor

Part A

Variables - A bit short, but never the less covers the main points. You could also have talked about the naming conventions for variables 8/10

Counting Loops - the point with a for loop is that the number of iterations is known before the loops starts, as opposed to a while loop where it isn't. You need to include a coded example 5/10

Arrays - a bit short although you've mentioned the main points. You need a coded example and you could also mention the index and how they start from 0 in C 5/10

Part B

Exercise 1

Code-Annotation - Generally good, but with suitable validation 5/5

Documentation - Breakdown is OK, remember that both input and output should use the parallelogram shape, you've used a rectangular box for your prints. For the testing, you've tested the happy path and some impossible - be careful to test the boundaries as well ie 0 and 120 and 0.5 and 2.4 The boundaries are where logic error creep in, so need careful testing.3.5/5

Exercise 2

Code-Annotation - Good implementation with suitable validation. Well done! Don't forget to declare all variables at the top of the program rather than halfway through for best practice. 9.5/10

Documentation- Breakdown is good - Testing is good, you've got most of the boundaries this time 4.5/5

Exercise 3

Code-Annotation You don't need to have an upper limit on the price really, just a lower limit to prevent negative entries. You started off with the array well, but then reverted to manual use further down - it would be more efficient to loop when adding the prices. Nevertheless, you have a working program. 8/10

Documentation Breakdown is good - testing is a bit short - you need to test the lower boundary (0) and negative numbers too 3.5/5

Exercise 4

Code-Annotation-Some of the statements on your else if are a bit redundant - once you've established that the numbers are not all 5s, you don't need to test again below. So, not entirely efficient, but nevertheless, a working game with an accumulating total. 15/20

Documentation Breakdown is a bit confused - you don't necessarily need a flowchart, but you should number the steps so the logic can be clearly shown. You need a bit more depth on the testing - every if statement needs to be tested 5.5/10

Total: 72.5%

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

Text Comment. Don't need Lnumber!=5, 3 5s have been tested for above

PAGE 16

PAGE 17

PAGE 18
