# Laboratory Report

## LABORATORY 4/5 – REPORT

**Student name:** Michael Lenehan

**Student ID:** 15410402

**Programme:** CE

I hereby declare that the attached submission is all my own work, that it has not previously been submitted for assessment, and that I have not knowingly allowed it to be used by another student. I understand that deceiving or attempting to deceive examiners by passing off the work of another as one's own is not permitted. I also understand that using another's student’s work or knowingly allowing another student to use my work is against the University regulations and that doing so will result in loss of marks and possible disciplinary proceedings.

**Signed:** Michael Lenehan **Date:**  10 March 2016

Note: Coursework examiners are entitled to reject any coursework which does not have a signed copy of this form attached or are submitted late.

Problem 1Determine whether the statements are true or false, and explain the answers given.

### Answers

1. **True:** A float occupies four bytes (a double occupies 8), an integer occupies 4 bytes and a char type occupies 1 byte.
2. **False:** The new variable replaces the value of the old variable unless otherwise defined.
3. **False:** Assignment of a float to an integer type variable will cause the loss of any number after the decimal.
4. **False:** The variable is correctly defined, the scan function however scans for a string and not for a character, and the print function prints for an integer and not a character.
5. **True:** All print functions will print a value of 4.

### Problem 2 (Problem Set No.1, Problem 1.1)

The task for this problem was to write a code to print the length of the hypotenuse of a right angled triangle of side’s length 3 and 4. Additionally the code must be adjusted to take inputs for the values of the length of the sides.

### Plan

Declare all necessary variables; adjacent, opposite, hypotenuse.

Print a welcome message.

Scan the length of the sides, ie. the adjacent and the opposite sides.

Compute the length of the hypotenuse.

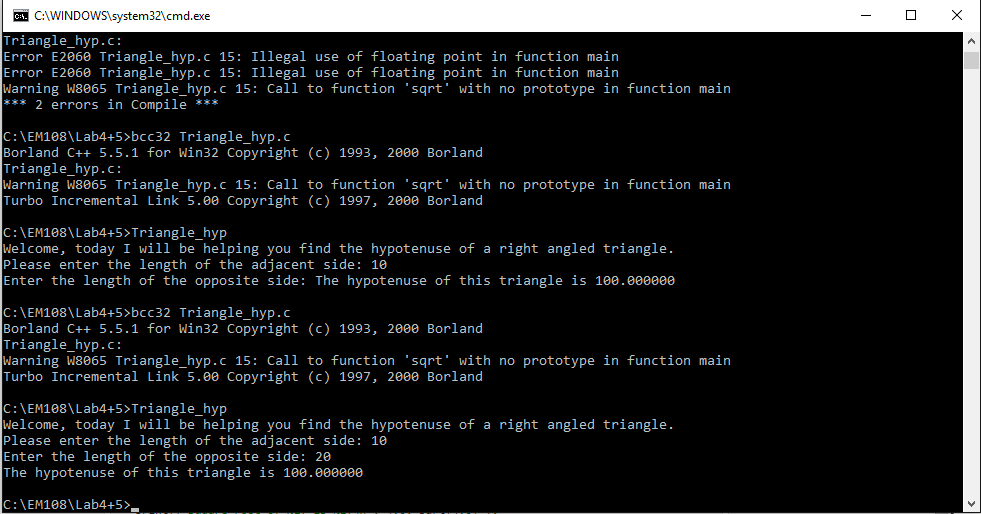
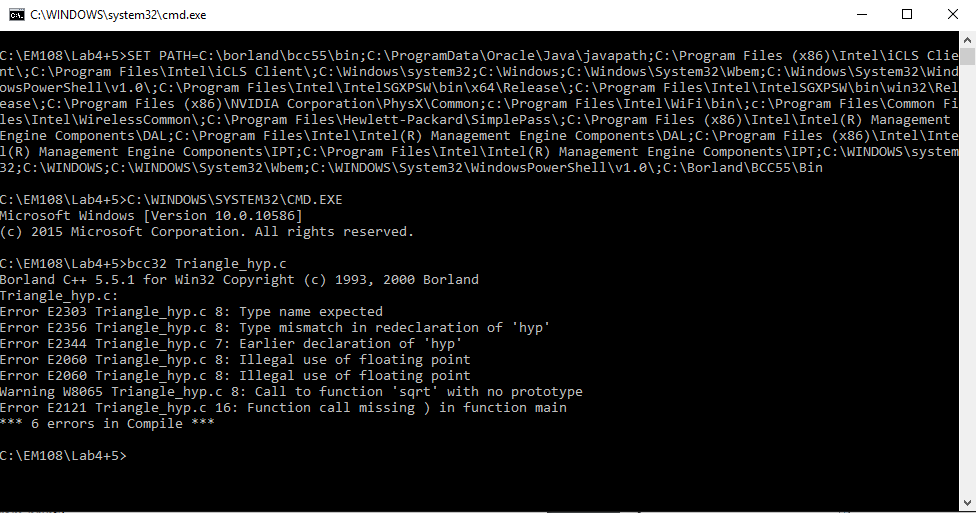
Print the value of the hypotenuse.

### Development

Development began by writing the code to compute the hypotenuse of a triangle of opposite and adjacent of lengths 3 and 4. The standard C libraries of ‘stdlib.h’, ‘stdio.h’, and ‘math.h’ were included. The float variables of the opposite, adjacent and hypotenuse were declared and the opposite and adjacent defined as 3 and 4 respectively. The expression to compute the length of the hypotenuse of the triangle was written. The print function to print the value of the length was then added and the programme terminated here. Once this had been deemed as correctly working, development moved to the second section of the problem, to write a code to scan in user inputs for the lengths of both the opposite and adjacent sides of the right angled triangle. Changes made included removing the definitions of the values of 3 and 4 for the opposite and adjacent sides, messages were printed to ask for user inputs for adj and opp variables, and scan functions were added to read these inputs.

### Testing

Syntax errors were encountered in the code.



The square root function was incorrectly used. Adjustments were made and the programme deemed as working correctly.

### Conclusion

This programme taught the use of the square root function and tested the ability to correctly write mathematical expressions in the code. It also tested the use of the basic print and scan functions.

### Code

//Include C libraries

#include <stdlib.h>

#include <stdio.h>

#include <math.h>

//Declare Variables

float adj;

float opp;

float hyp;

//Main function scans for inputs and outputs hyp value

int main()

{

printf("Welcome, today I will be helping you find the hypotenuse of a right angled triangle.\nPlease enter the length of the adjacent side: ");

scanf("f\n", &adj);

printf("Enter the length of the opposite side: ");

scanf("%f\n", &opp);

//compute hyp value

hyp = sqrt(adj\*adj) + (opp\*opp);

printf("The hypotenuse of this triangle is %.01f\n" ,hyp);

return(EXIT\_SUCCESS);

}

### Problem 3 (Problem Set No.1, Problem 1.3)

The task for this problem was to create a programme which could calculate the height of a building using the expression h=1/2(g.t^2). The programme must take three values for t from the user, find the average and output a value for h.

### Plan

Declare all necessary variables; h, g, t1, t2, t3, tav.

Print a suitable welcoming message.

Take input values for t1, t2, and t3 using the scan function.

Compute the value of tav (the average time value) and h (the height of the building).

Print the value of h.

Terminate the programme.

### Development

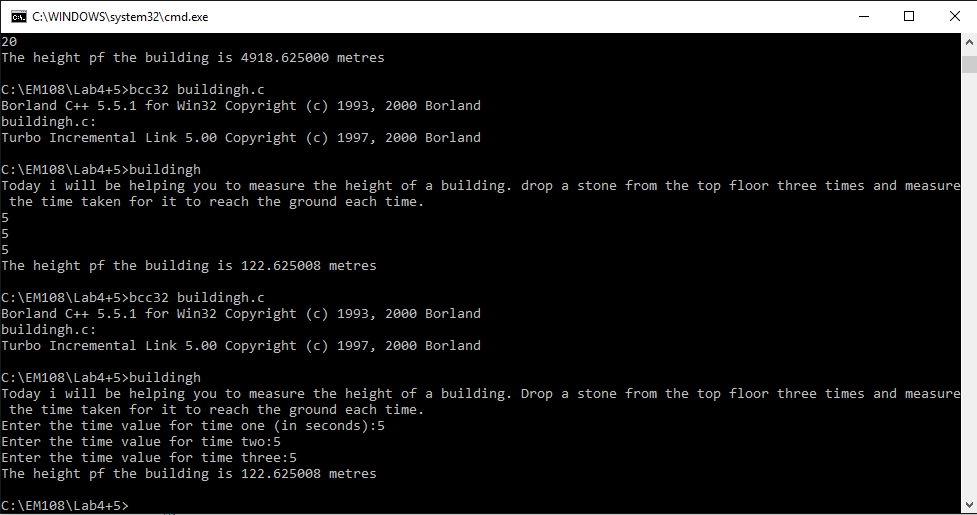
Development began by including the standard C libraries of ‘stdlib.h’, and ‘stdio.h’. The float variables were then all declared. The main function then prints an opening message, and requests values for t1, t2, and t3. The function scans these values and an expression was written to compute the values of tav, and the height of the building, h. The value of h is then output to the screen and the programme terminates.

### Testing

Scan function was incorrectly scanning twice for the value of t1.



The mistake in the code (an extra scan function) was removed and the end print text was added.



A spelling mistake in the final line was corrected, a suitable number of decimal places was added, and the programme was again tested and deemed working.

### Conclusion

This programme tested the use of scan and print functions, along with the correct declaration of variables. It also tested the ability to correctly compute the average of a set of values and the use of expressions to compute the values of variables using a given formula.

### Code

//C libraries included

#include <stdlib.h>

#include <stdio.h>

//Variables declared

float h, g, t1, t2, t3, tav;

//main function scans inputs and computes value for h

int main()

{

printf("Today i will be helping you to measure the height of a building. Drop a stone from the top floor three times and measure the time taken (in seconds) for it to reach the ground each time.\n");

printf("Enter the time value for time one:");

scanf("%f", &t1);

printf("Enter the time value for time two:");

scanf("%f", &t2);

printf("Enter the time value for time three:");

scanf("%f", &t3);

//compute average time using inputs

tav = (t1+t2+t3)/3;

//define value for g and compute value for h

g = 9.81;

h = (g\*tav\*tav)/2;

printf("The height of the building is %.02f metres.\n", h);

return(EXIT\_SUCCESS);

}

### Problem 4 (Problem Set No.2, Problem 2.5)

The task of this problem was to write a programme to calculate the cost of a circular steel plate, cut from a square steel sheet, and the amount of wastage from the cutting, and the return value of this wastage.

### Plan

Declare any necessary variables; r, pi, area\_s, cost, refund, area\_c, cost\_c, cost\_s.

Print an opening message.

Request input value for the radius of the circular plate and scan the input value.

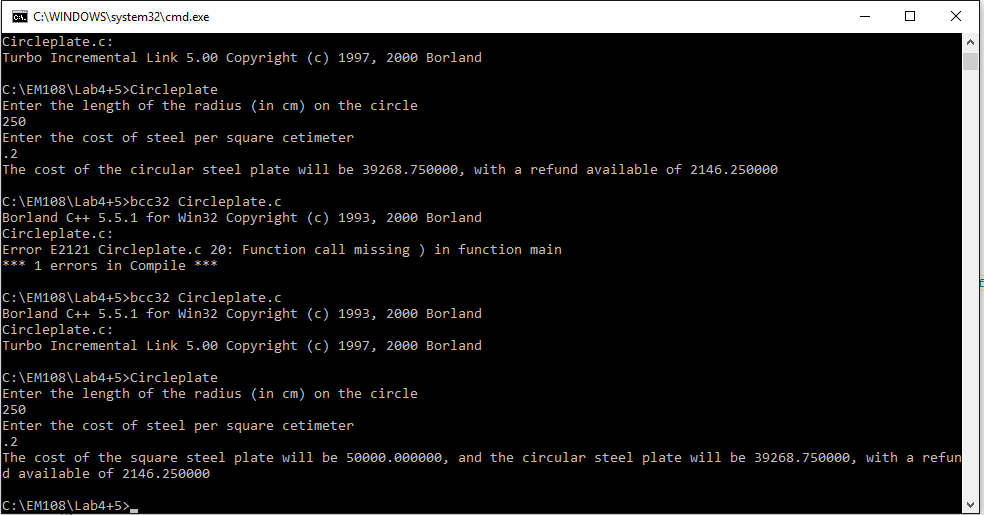
Request input value for the cost of steel per cm^2 and scan the input value.

Compute the cost of the steel needed, the refund available for the wastage from the process and print these results.

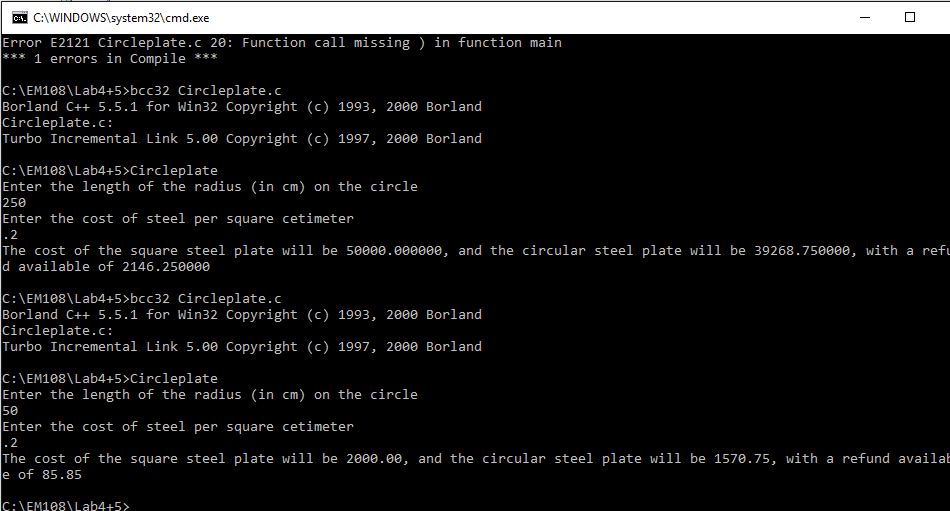
### Development

Development again began with the inclusion of the C libraries ‘stdlib.h’, ‘stdlib.h’, and ‘math.h’. All float variables were then declared. An opening message is then printed. A message to request the input values for the radius of the circle and the cost per cm^2 of the circle is then printed, and the input values are scanned. The value for pi was defined, and expressions to calculate the area of the circle and the square, the cost of the steel for the circle and the square sheet were written. An expression was then added which would calculate the value of refund available for returning the left over scrap. A print function then prints the values of the cost of the circular and square sheets of steel and the refund value.

### Testing



Steel plate was scanning incorrectly due to absence of ampersand (&) symbol in the scan function. This was corrected along with errors in the included expressions which caused incorrect calculations.



Decimal points were added to neaten up the output values and give a more finished look to the programme. The code was then deemed to be correctly working.

### Conclusion

This problem tested the ability to use scan and print functions correctly. It also tested the use of expressions, and the defining of variables (such as defining a value for pi).

### Code

//Include C libraries

#include<stdio.h>

#include<stdlib.h>

#include<math.h>

//Declaration of variables

float r, pi, area\_s, cost, refund, area\_c, cost\_c, cost\_s;

//Main function scans values and outputs cost of the square and circular sheets along with refund value

int main()

{

printf("Welcome! Today I will be calculating the cost of the steel you will need to make a circular plate of radius r!\n");

printf("Enter the length of the radius (in cm) on the circle\n");

scanf("%f", &r);

printf("Enter the cost of steel per square cetimeter\n");

scanf("%f", &cost);

//Value of pi is defined

pi = 3.1415;

//expressions calculate the area of the square and circular sheets and their costs

area\_c = pi\*(r)\*(r);

cost\_c = area\_c\*cost;

area\_s = (2\*r)\*(2\*r);

cost\_s = area\_s\*cost;

refund = (area\_s-area\_c)\*(cost\*0.20);

printf("The cost of the square steel plate will be %.2f, and the circular steel plate will be %.2f, with a refund available of %.2f\n", cost\_s, cost\_c, refund);

return(EXIT\_SUCCESS);

}