Labsheet 8 - Programs using different Data Types and Returning from Functions

You should be getting **3 programs done per lab session at a minimum**. If you don't you should work on them outside of lab time and have them done before the next lab session.

Remember to design your programs with pseudo-code and/or flow diagrams first. Name the programs appropriately.

It is absolutely prohibited to copy anyone else's design or code. You can ask for help with a particular problem from friends, colleagues, the lecturers in the lab etc but you must write your own design and code the fix to the problem yourself.

If you get help from someone or take design/code from the web or elsewhere, you have to comment in your code to state what help you got and from where.

Saving Your Projects

You can create a new project for each program or create them as separate files within the one project.
You should save the programs to a folder called **labsheet 8** within **Programming\Labsheets** folder on your **M drive**.

Every program should have a **comment** at the top stating your name, student ID number, date created, approx number of hours worked on, overall brief description of the program and any known bugs in it.

Important: Global variables should be kept to a minimum. They should only be used when absolutely necessary.

Q1.

Write a C++ program that asks the user to input the number of a month and outputs the name of that month. **Important:** Write this with a **switch** statement. Appropriate error message should be displayed for invalid input.

Q2.

a. Write the pseudocode algorithm for this program. The **main()** function should read the centimetres from the user. The centimetres should be passed as an argument to the function inchesEquivalent( ). Write the function **`inchesEquivalent()`** which accepts an argument in centimetres and converts it to inches. The function inchesEquivalent( ) should return the inches back to the main() function. The main( ) function should display the inches on the screen.

   An example function call is

   ```
   float inches = inchEquivalent(10.5);
   ```

   Hint: Multiply centimetres by 0.394 to calculate inches.

   An appropriate error message should be displayed if the user enters a number less than or equal to 0.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q3.

a. Write the pseudocode algorithm for this program. The **main()** function should read the time in hours, minutes and seconds from the user. These 3 arguments should be passed to the function **secondsCal( )**. Write the function **secondsCal( )**, which accepts three integers, representing a time in hours, minutes and seconds. It should calculate the total time in seconds and **return** it to the main() function. The main() function should display the total time in seconds to the screen.

   An example function call is:

   ```
   int totalSecs = secsIn( 1, 1, 2 );   // returns 3662
   ```

   An appropriate error message should be displayed if the user enters a number less than 0.
   An appropriate error message should be displayed if the user enters all 0s when inputting the hours, minutes and seconds.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q4. Exam Grades

a. Write the pseudocode algorithm for this program which asks for a student's overall exam mark. The exam mark is then checked and depending on the exam mark one of the following messages are displayed:

   Fail – if the mark is less than 40
   Pass – if the mark is greater than or equal to 40 and less than 55
   Merit – if the mark is greater than or equal to 55 and less than 70
   Distinction – if the mark is greater than or equal to 70 and less than or equal to 100
   Appropriate error message – If the mark is less than 0 or greater than 100

   This program should use two functions main() and getExamGrade( ) as follows:

   The main( ) function should ask the user to enter an exam mark. It should then call a
   function getExamGrade( ) which accepts a single argument which is the exam mark. The function getExamGrade( ) should work out what the grade is and return the grade as a string back to main( ). Main( ) should then display the appropriate grade on the screen.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q5.

a. Write the pseudocode algorithm for this program which asks the user for 2 numbers, it then indicates whether the first number is exactly divisible by the second number.

   Structure your program to use functions as follows. Main( ) should read the two numbers from the user, it should then call another function called **isDivisible()** which should accept the 2 numbers that have been inputted as arguments. This function should then return a boolean of true or false depending whether the first number is divisible by the second number. An appropriate message is then output to the screen in **main( )** function stating whether the numbers are divisible or not.

   An appropriate error message should be displayed if the user enters a number less than 0.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q6. Deal a card

a. Write the pseudocode algorithm for this program which will deal a card. This program has a main( ) and two other functions as follows:

   Write a function called **getCardSuit()**. This function should randomly generate a number between 1 – 4 which represents the suit of the card. The program should then convert the number to a suit (hearts, diamonds, clubs and spades). The function should return this string.

   Write a second function called **getCardFace()**. This function should randomly generate a number between 1 – 13 which represents the card number. The program should then convert this number to a string which represents the number of the card. For example a 1 represents an Ace, 2 the number 2 … 11 is a Jack, 12 is a Queen and so on.

   Note: use a **Switch** statement as appropriate.

   For example your program could output the following:
   10 Hearts
   Queen Diamonds
   Ace Clubs
   Etc.

   You should structure your program as follows:

   The **main( )** function should call the two other functions for the suit and number of the card. It should then display the card on the screen.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q7. Cinema Example

a. Write the pseudocode algorithm for this program making use of functions to work out how much a person pays to go to the cinema. The program should input an age and then output the cost of your cinema ticket based on the following:

- Under 5 is free
- Aged 5 to 12 is half price
- Aged 13 to 54 is full price
- Aged 55 or over is free

The ticket price is 6 euro.

Your solution should contain a function called **calculatePrice(age)** which accepts someone's age and returns the price of the ticket for that age.

Your program should display an error message if an age below 0 is entered.

Note: It should be easy to change the price of the ticket and your program should still work correctly.

Question: Does your program still work if the ticket price is 7 euro?

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q8.

a. Write the pseudocode algorithm for this program which asks the user to enter the length and width of a rectangle and calculates the area of the rectangle. Use a separate function in your program which calculates the area of the rectangle as follows:

```
float area = calAreaRectangle(length, width);
```

The area value calculated is returned from the function `calAreaRectangle();` Also calculate the perimeter of the rectangle as a separate function.

Write the **main()** function which asks the user to enter the length and width of a rectangle and calls the function `calAreaRectangle()` that calculates the area. It should also call a function to calculate the perimeter of the rectangle.

An appropriate error message should be displayed if the user enters a number less than or equal to 0.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q9.

a. Write the pseudocode algorithm for this program which does the following: There is a function named **secsToHMS()** which takes in a number of seconds and converts them to hours, minutes and seconds. **Hint:** Make use of the % and / operators.

   For example 3662 seconds is 1 hour, 1 minute and 2 seconds.

   The **main()** function asks the user to enter the total number of seconds and calls the function **secsToHMS()** that converts it to hours, minutes and seconds.

   An appropriate error message should be displayed if the user enters a number less than or equal to 0.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q10. Student Exams

a. Write the pseudocode algorithm for the following program which:
Asks a student to enter their student number followed by their assessment mark and then their final exam mark. Verify the user input i.e. the assessment and exam marks are between 0 –100. Display an error message if they are not.

Student Number:
Student Assessment Mark:
Student Final Exam Mark:

Write a function called **calFinalResult()** which accepts the assessment and exam marks as arguments. This function should calculate the final result and return it to the main( ) function. The final result is calculated as follows:
Assessment – 30%
Final Exam – 70%

Display the final result on the screen.

For example if the following data is entered:

Student Number: XXXX
Student Assessment Mark: 60
Student Final Exam Mark: 80

The student's final result is: 74%

Student Number: YYYY
Student Assessment Mark: 75
Student Final Exam Mark: 30

The student's final result is: 43.5%

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q11.

a. Write the pseudocode algorithm for the following program which:

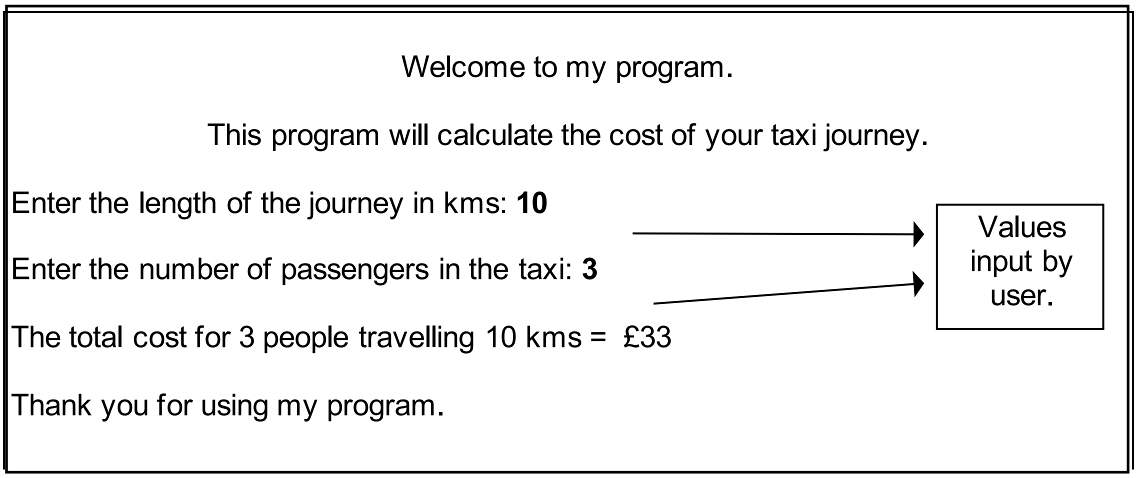
A taxi service charges a minimum of 4.00 euro per person. Then an extra 1 euro per km travelled for every km over 3 km is charged, again per person in the taxi.

So if 2 people travel 2 kms they are charged: 2×€4 = €8

If 3 people travel 10 kms they are charged: 3×€4 + 3×€1×(10 - 3) = €12 + €21 = €33

Write a program which will allow the user to enter the values for the number of passengers in the taxi and the distance travelled (in kms) on the journey. The program will then calculate the total cost of the journey and print out the number of passengers, the kms travelled and the total cost of the journey.

The program, when run, might output the following (bold print is input by user):



```
                    Welcome to my program.

         This program will calculate the cost of your taxi journey.

Enter the length of the journey in kms: 10

Enter the number of passengers in the taxi: 3

The total cost for 3 people travelling 10 kms =  £33

Thank you for using my program.
```

You should write a function called **calCost( )** which will take the length of the journey and the number of passengers as arguments and return the total cost of that taxi journey.

Appropriate error messages should be displayed for invalid inputs.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q12. Worm Journey

a. Write the pseudocode algorithm for the following program which:

A worm is at the bottom of a wall. Every day the worm crawls up 3 feet, but at night it slips down 2 feet.

Write a program that asks the user to enter the height of the wall and it should then work out how many days it would take the worm to get to the top of the wall.

You should write a function called **calNoDays( )** which accepts the height of the wall as an argument and it should then work out and return how many days it would take the worm to get to the top of the wall.

Appropriate error messages should be displayed for invalid inputs.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q13. Elevator

a. Write the pseudocode algorithm for the following program which:

The capacity of an elevator is either 20 children or 15 adults.

Write a program which will ask the user to enter the number of children getting into the elevator. The program should then calculate and output to the screen the number of adults that can still get in.

Write a function called **calNoAdults( )** which will accept the number of children as an argument. It will then calculate and return how many adults can still get into the elevator.

For example if 12 children are currently in the elevator, how many adults can still get in?

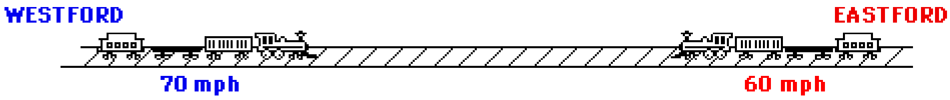
Appropriate error messages should be displayed for invalid inputs.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q14. Trains

Two trains leave different cities heading toward each other at different speeds.

Train A, traveling 70 km per hour (kmph), leaves Westford heading toward Eastford, 260 miles away. At the same time Train B, traveling 60 kmph, leaves Eastford heading toward Westford. When do the two trains meet? How far from each city do they meet?



a. Write the pseudocode algorithm for the above program which will ask the user to enter the speed of train A (kmph) and the speed of train B (kmph). It should then calculate when and where the two trains will meet.

   Make use of **functions** as appropriate.

   Appropriate error messages should be displayed for invalid inputs.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.

Q15. Theater Group

There are eight more women than men in a theater group.

a. Write the pseudocode algorithm for the above program which will ask the user to enter the total number of members. Your program should then work out how many men and how many women are in the group.

   Make use of **functions** as appropriate.

   For example if the group has a total of 44 members. How many men and how many women are in the group?

   Appropriate error messages should be displayed for invalid inputs.

b. Next write test data for this program.
c. Next code the program.
d. Next test your code against the test data.