

Kieran Perrin

ACADEMIC &  
EMPLOYABILITY SKILLS

In Class Test under Exam Conditions



Question 1.

- A. The name of the function "InputCostsProcessAndOutputResult" suggests that the programmer has not given sufficient thought into the structure of the code. Some reasons for this are that the programmer has put everything into a single function and not split the program into sections of the different problems. This to me says that they rushed straight into writing the program without prior thought. Because of this if the code were needed to be changed it would be difficult for anyone other than the original programmer to implement any changes.

B.

1. Set username and password ✓
2. Ask user for username and password ✓
3. Validate the username and password ✓
4. If user attempts username and password 3 times and fails set timer for 5 minutes until they can try again ✓
5. Ask user for amount of products bought ✓
6. Validate the user entered a positive number ✓
7. Ask user for price of items for amount they entered ✓
8. Validate that the user enters a positive number to 2 decimal points ✓
9. Calculate 20% of all the items ✓
10. Add all the VAT of the items together ✓
11. Output total of VAT to user ✓

C.

1. Test that not entering correct username and password locks user out ✓
2. Entering correct username and password allows user access ✓
3. Entering a non-positive number ask user to enter a positive number ✓
4. Entering a positive number ✓
5. Entering a negative number or letters asks the user to enter a correct price ✓
6. Entering item price ✓
7. Check the calculation of the program is correct ✓

Question 2

A.

1. Display Menu ✓
2. Ask user for their option for the menu ✓
3. Validate user input an option for menu, if not ask user again ✓
4. If they chose area of rectangle ask user for height ✓
5. Validate height is a positive number, if not ask user again ✓
6. Ask user for width ✓
7. Validate width is a positive number, if not ask again ✓
8. Calculate area and display ✓
9. If user chose area of circle ask user for radius ✓
10. Validate radius is a positive number, if not ask again ✓
11. Calculate area of circle and display ✓
12. If user chose exit, exit the program ✓

### Question 3

A.

First you would ask the user for the x position of the square, then ask for the y position of the square. This almost plots a point on an imaginary graph. Then the user is asked for the width of the square. This allows us to point the top right, bottom left and bottom right points of the square as squares have the same length sides all around. To work out top right, the math would be  $x + \text{width}$ . Bottom left is  $y - \text{width}$ . Bottom right is bottom left + width. If you give each point on the square a letter it can simplify the equation

Top left = a, b

Top right = c, b

Bottom left = a, d

Bottom right = c, d

$c = a + \text{width}$

$d = b - \text{width}$

To work out if the point is in the square or not you could use the equation

If E,F is the new point

If  $e < a$  or  $e > c$  and  $F < d$  or  $F > b$  then it wouldn't be within the square

If the top right position of the square is 5,5 and its width is 5

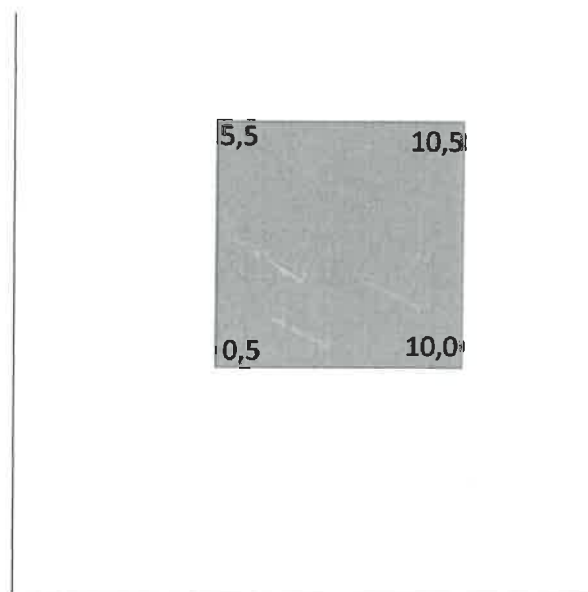
$a = 5$

$b = 5$

$c = 10$

$d = 0$

So it would work out like:



If the user then asks if 6,5 is in the square 6 is greater than a and less than c it would be in the square.

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Exampractice //Kieran Perrin - PER09098703
{
    class Program
    {
        static void Main(string[] args)
        {

            bool validate = false;

            while (!validate)
            {
                Program.Menu(); //draws the method menu to show the menu for the user

                validate = int.TryParse(Console.ReadLine(), out int option); //if the
                number is a valid nu,ber the code will continue if not it will ask user
                again

                if (option == 1)
                {
                    Program.Rectangle();
                }

                if (option == 2)
                {
                    Program.Circle();
                }

                if (option == 3)
                {
                    Environment.Exit(0);
                }

                if (option != 1 && option != 2 && option != 3) //if the menu number used
                is not valid will show menu and ask user agian
                {
                    Console.WriteLine("Please Enter an option from the menu");
                    validate = false;
                }
            }
            Console.ReadLine(); //pause

        }

        public static void Menu() //method for menu
        {
            Console.WriteLine("----Menu----");
            Console.WriteLine("1. Calculate area of rectangle");
            Console.WriteLine("2. Calculate area of circle");
            Console.WriteLine("3. Exit");
            Console.WriteLine("");
            Console.Write("What option do you wish to choose ");
        }

        public static void Rectangle() //method for rectangle calculation
        {
            double Width = 0;

```

```

double Height = 0;
bool validate = false;

Console.Clear(); //clear the console so the user experience is easier
Console.WriteLine("You chose option 1 area of rectangle");

while (!validate) //performs the code unless the user inputs a valid number
{
    Console.Write("Please enter the Width in cm: ");
    validate = double.TryParse(Console.ReadLine(), out Width);
    if (validate == false)
    {
        Console.WriteLine("Please enter a valid number");
    }
}
validate = false;

while (!validate)
{
    Console.Write("Please enter the Height in cm: ");
    validate = double.TryParse(Console.ReadLine(), out Height);
    if (validate == false)
    {
        Console.WriteLine("Please enter a valid number");
    }
}
validate = false;

Console.WriteLine("");
double rectangle = (Width * Height);
Console.WriteLine("The area of the rectangle is " + rectangle + "cm\xB2");
Console.ReadLine(); //pause
Environment.Exit(0);
}

public static void Circle() //method for circle calculation
{
    double radius = 0;
    bool validate = false;
    Console.Clear();
    Console.WriteLine("You chose option 2 area of a circle");

    while (!validate)
    {
        Console.Write("Please enter the Radius in cm: ");
        validate = double.TryParse(Console.ReadLine(), out radius);
        if (validate == false)
        {
            Console.WriteLine("Please enter a valid number");
        }
    }

    double volume = (3.14 * (radius * radius));
    Console.WriteLine("The area of the circle is " + volume + "cm\xB2");
    Console.ReadLine(); //pause
    Environment.Exit(0);
}
}
}
}

```

Exam Marking ACCA4016 – Academic and Employability Skills 12<sup>th</sup> March 2019

Student: **Kieram Perrin**

|          | Exam Date: 12 <sup>th</sup> March 2019 | Marking Date: 15 <sup>th</sup> March 2019 |
|----------|----------------------------------------|-------------------------------------------|
| Question | Max Possible Marks                     | Marks Achieved                            |
| 1a       | 4                                      | 4 ✓                                       |
| 1b       | 8                                      | 7 ✓                                       |
| 1c       | 8                                      | 7 ✓                                       |
|          |                                        |                                           |
| 2a       | 20                                     | 18 ✓                                      |
| 2b       | 30                                     | 30 ✓                                      |
|          |                                        |                                           |
| 3a       | 20                                     | 20 ✓                                      |
| 3b       | 10                                     | 0                                         |
|          |                                        |                                           |
|          |                                        |                                           |
| Total    | 100                                    | 86 ✓                                      |

Lecturer: G.P. Dawkins

