**Task 1 - 7 Step Algorithm**

Use your jotter to follow the steps for this algorithm exactly as they are written.

You cannot ask questions – just do exactly what the instruction says. DO NOT let anyone else see your results. .

1. Draw a diagonal line.
2. Draw another diagonal line connected to the top of the first one.
3. Draw a straight line from the point where the diagonal lines meet.
4. Draw a horizontal line over the straight line.
5. At the bottom of the straight line, draw a curvy line.
6. Draw a diagonal line from the bottom of the first diagonal to the straight line.
7. Draw a diagonal line from the bottom of the second diagonal to the straight line.
* Show and Tell – What shape did you get?
* What was wrong with this algorithm?
* How could you improve it?

**Task 2 – Create your own**

Write your own algorithm for a simple shape that can be drawn on paper.

Hand the algorithm to your partner for them to complete the shape. You cannot tell them any additional instructions.

Take it in turns to complete each other’s algorithm.

Discuss the results with what you expected your partner to achieve. What was difficult about the task? Could your algorithm have been improved?

The following tasks will enable you topractice your analytical and problem solving skills through the development of Algorithms. For each task a scenario is presented. You should analyse the scenario and create an Algorithm for your proposed solution. The initial focus will be on writing the Algorithm using Pseudocode, progressing to Structured Diagrams, Flowcharts and finally to writing the actual programs using Visual Basic.

**EXAMPLE 1 -** Calculate the sum and average of two numbers. A sample Algorithm using Pseudocode and a Structured Diagram is shown below:

|  |  |  |
| --- | --- | --- |
| **PSEUDOCODE** | **STRUCTURED DIAGRAM** | **FLOWCHART** |
|

|  |  |
| --- | --- |
| 1 | **Get Numbers** |
| 2 | **Calculate sum and average** |
| 3 | **Display sum and average** |
|  |  |
| 1 | **Get Numbers** |
| 1.1 | Get 1st number 1 |
| 1.3 | Get 2nd Number |
|  |  |
| 2 | **Calculate sum & average** |
| 2.1 | Total = 1st Number + 2nd Number  |
| 2.2 | Average = Total / 2  |
|  |  |
| 3 | **Display Sum & Average** |
| 3.1 | Send Total to display  |
| 3.2 | Send Average to display  |

 |  |  |

**NOTE**: - The structure chart shows the relationship among the various components of the program. The top level is a general description of the problem to be solved where the lower levels show all the detail or specifics of the solution. The second level of a structure chart attempts to break the solution into three basic parts - an **input module**, a **processing module** and an **output module**. The sequence is interpreted by reading from left to right.

**Task 3 -** Design the algorithm for a program that calculates the area of a triangle.

|  |  |  |
| --- | --- | --- |
| **PSEUDOCODE** | **STRUCTURED DIAGRAM** | **FLOWCHART** |
|

|  |  |
| --- | --- |
| 1 |  |
| 2 |  |
| 3 |  |
|  |  |
|  |  |
|  |  |
|  |  |
| 1 |  |
| 1.1 |  |
| 1.2 |  |
|  |  |
|  |  |
| 2 |  |
| 2.1 |  |
| 2.2 |  |
|  |  |
|  |  |
| 3 |  |
| 3.1 |  |
| 3.2 |  |
|  |  |
|  |  |

 |  |  |

**Task 4 -** Design the algorithm for a program that calculates the total of a retail sale. The program should ask the user for the following: the retail price of the item being purchased and the sales tax rate. Once the information has been entered the program should calculate and display the following: the sales tax for the purchase and the total sale.

|  |  |  |
| --- | --- | --- |
| **PSEUDOCODE** | **STRUCTURED DIAGRAM** | **FLOWCHART** |
|

|  |  |
| --- | --- |
| 1 |  |
| 2 |  |
| 3 |  |
|  |  |
|  |  |
|  |  |
|  |  |
| 1 |  |
| 1.1 |  |
| 1.2 |  |
|  |  |
|  |  |
| 2 |  |
| 2.1 |  |
| 2.2 |  |
|  |  |
|  |  |
| 3 |  |
| 3.1 |  |
| 3.2 |  |
|  |  |
|  |  |

 |  |  |

**Task 5 - -** Design the algorithm for a program that calculates the current balance in a savings account. The program should obtain from the user the following information: the starting balance, the total amount of deposits made, the total amount of withdrawals made, and the monthly interest rate. After the program has calculated the current balance, it should be displayed on the screen. Assume one input for deposits and one input for withdrawals.

Draw the Structure Diagram for this algorithm.

|  |  |  |
| --- | --- | --- |
| **PSEUDOCODE** | **STRUCTURED DIAGRAM** | **FLOWCHART** |
|

|  |  |
| --- | --- |
| 1 |  |
| 2 |  |
| 3 |  |
|  |  |
|  |  |
|  |  |
|  |  |
| 1 |  |
| 1.1 |  |
| 1.2 |  |
|  |  |
|  |  |
| 2 |  |
| 2.1 |  |
| 2.2 |  |
|  |  |
|  |  |
| 3 |  |
| 3.1 |  |
| 3.2 |  |
|  |  |
|  |  |

 |  |  |

**Task 6 –** Design an algorithm for Mrs Bird’s scenario.Mrs Bird is taking 15 students on a school trip. The cost of the trip is £50 but the students can pay by instalment. Mrs Bird has called a meeting to take the first instalment and she wants a piece of software (a program) to let her enter the amount paid by each student. At the end of the meeting she would like to be able to display how much each student has left to pay. She also wants to know how many students still have the full £50 left to pay.

|  |  |  |
| --- | --- | --- |
| **PSEUDOCODE** | **STRUCTURED DIAGRAM** | **FLOWCHART** |
|

|  |  |
| --- | --- |
| 1 |  |
| 2 |  |
| 3 |  |
|  |  |
|  |  |
|  |  |
|  |  |
| 1 |  |
| 1.1 |  |
| 1.2 |  |
|  |  |
|  |  |
| 2 |  |
| 2.1 |  |
| 2.2 |  |
|  |  |
|  |  |
| 3 |  |
| 3.1 |  |
| 3.2 |  |
|  |  |
|  |  |

 |  |  |

**Task 7 –** Design the algorithm for a program that will compute the average of 10 exam scores entered by the user. List the variables needed for this program. Perform a dry run using the following values: 75, 25, 51, 49, 88, 12, 22, 23, 52, and 78.

**Algorithms for Program**

|  |  |  |
| --- | --- | --- |
| **PSEUDOCODE** | **STRUCTURED DIAGRAM** | **FLOWCHART** |
|

|  |  |
| --- | --- |
| 1 | Get Details |
| 2 | Calculate Average |
| 3 | Display Average |
|  |  |
| 1 | **Get Details** |
| 1.1 | Repeat 10 times |
| 1.2 | Get Number |
| 1.3 | Add Number to total |
|  |  |
| 2 | **Calculate Average** |
| 2.1 | Divide total by 10 for average |
|  |  |
| 3 | **Display average** |
| 3.1 | Send average to display |

 |  |  |

**Task 8-** Design and Algorithm for a program that will dispense money from an ATM machine. Continue to request a withdrawal amount from the user until they enter -999. If the amount entered is not evenly divisible by twenty, output the message "You must enter multiples of twenty only" otherwise calculate and output the number of twenty pound notes you will be dispensing. List the necessary variables for this program. Perform a dry run with the following input values: 35, 60, 61, 360, and -999. (Page 27, + 31)

|  |  |  |
| --- | --- | --- |
| **PSEUDOCODE** | **STRUCTURED DIAGRAM** | **FLOWCHART** |
| 1. SET amount = 0
2. WHILE amount NOT(-999) DO
3. RECEIVE amount FROM KEYBOARD AS INTEGER
4. IF (amount/20) - int(amount/20) NOT = 0 THEN
5. SEND “You must enter multiples of 20 only”
6. ELSE
7. notes = amount/20
8. SEND notes TO DISPLAY
9. END IF
10. END WHILE
 |  |  |