P2:Pseudocode

19: The manufacturing cost of producing an item depends on its complexity. A company manufactures three different types of item, with costs based on the following calculations:

Item type 1: item cost = parts cost * 1.5
Item type 2: item cost = parts cost * 2.5
Item type 3: item cost = parts cost * 5.0

The company makes 1000 items per day.

Write an algorithm, using pseudocode, flowchart or otherwise, which
- **inputs the item type and parts cost of each item**
- **outputs the item cost for each item**
- **calculates and outputs the average (mean) item cost per day (based on 1000 items being made). [5]**

ANSWER:

```
TotalCost <-- 0
FOR Count <-- 1 TO 1000
INPUT ItemType, PartsCost
IF ItemType=1 THEN ItemCost=PartsCost*1.5
ELSE IF ItemType=2 THEN ItemCost=PartsCost*2.5
ELSE IF ItemType=3 THEN ItemCost=PartsCost*5.0
OUTPUT Item Cost
TotalCost <-- TotalCost + ItemCost
NEXT
Average=TotalCost/1000
OUTPUT Average
```

16 (a) Write an algorithm, using pseudocode or a flowchart, which:
- **inputs 50 numbers**
- **outputs how many of the numbers were > 100 [3]**

(b) Write an algorithm, using pseudocode or a flowchart, which:
- **inputs 100 numbers**
- **finds the average of the input numbers**
- **outputs the average [3]**

ANSWER:

a)
```
ValidNumber <-- 1
Count <-- 1
WHILE Count<50 DO
INPUT Number
IF Number>100
THEN ValidNumber <-- ValidNumber+1
ENDWHILE
OUTPUT ValidNumber
```

b)
```
Count <-- 1
Total <-- 0
FOR Count <-- 1 TO 100
INPUT Number
Total <-- Total+Number
NEXT
Average=Total/100
OUTPUT Average
```

18: A group of students were monitoring the temperature every day over a one-year period. Readings were taken ten times every day (you may assume a year contains 365 days).
Write an algorithm, using pseudocode or flowchart, which
- **inputs all the temperatures (ten per day)**
- **outputs the highest temperature taken over the year**
- **outputs the lowest temperature taken over the year**
- **outputs the average temperature per day**
- **outputs the average temperature for the whole year [7]**

ANSWER:

```
Highest <-- -200
Lowest <-- 200
Total_Year_Temp <-- 0
FOR Days <-- 1 TO 365
 Total_Day_Temp <-- 0
  FOR Count <-- 1 TO 10
     INPUT Temp
     Total_Day_Temp = Total_Day_Temp + Temp
     Total_Year_Temp = Total_Year_Temp + Temp
     IF Temp>Highest THEN Highest=Temp
     IF Temp<Lowest THEN Lowest=Temp
  NEXT Count
Average_Day_Temp=Total_Day_Temp/10
OUTPUT Average_Day_Temp
NEXT
Average_Year_Temp=Total_Year_Temp/3650
OUTPUT Highest, Lowest, Average_Year_Temp
```

17 (a) A car's speed is measured between points A and B, which are 200 km apart.

The final speed of the car is calculated using the formula:

$$\text{Final Speed} = \frac{200}{\text{Time (hours)}}$$

What is the final speed of a car if it takes 2 hours to get from A to B? [1]

(b) Write an algorithm, using pseudocode or otherwise, which inputs the times for 500 cars, calculates the final speed of each car using the formula in part (a), and then outputs:

- **the final speed for ALL 500 cars**
- **the slowest (lowest) final speed**
- **the fastest (highest) final speed**
- **the average final speed for all the cars. [6]**

ANSWER:

a) 100 km/h

b)
```
Lowest <-- 1000
Highest <-- 0
Total_Speed <-- 0
FOR Cars <-- 1 TO 500
INPUT Time
Final_Speed=200/Time
Total_Speed <-- Total_Speed+Final_Speed
IF Final_Speed>Highest THEN Highest=Final_Speed
IF Final_Speed<Lowest THEN Lowest=Final_Speed
OUTPUT Final_Speed
NEXT
Average_Final_Speed=Total_Speed/500
OUTPUT Highest, Lowest, Average_Final_Speed
```

17 A school is doing a check on the heights and weights of all its students. The school has 1000 students. Write an algorithm, using pseudocode or a flowchart, which

- **inputs the height and weight of all 1000 students**
- **outputs the average (mean) height and weight**
- **includes any necessary error traps for the input of height and weight [5]**

ANSWER:
```
Total_Height <-- 0
Total_Weight <-- 0
FOR Count <-- 1 TO 1000
  INPUT Height, Weight
  IF Height>10 OR Height<0
  THEN OUTPUT 'Error'
  IF Weight>300 OR Weight<0
  THEN OUTPUT 'Error'
  ELSE Total_Height<-- Total_Height+Height
  AND Total_Weight<-- Total_Weight+Weight
NEXT
Average_Height=Total_Height/1000
Average_Weight=Total_Weight/1000
OUTPUT Average_Height, Average_Weight
```

5000 numbers are being input which should have either 1 digit (e.g. 5), 2 digits (e.g. 36), 3digits (e.g. 149) or 4 digits (e.g. 8567). Write an algorithm, using pseudocode or flowchart only, which

- **inputs 5000 numbers**
- **outputs how many numbers had 1 digit, 2 digits, 3 digits and 4 digits**
- **outputs the % of numbers input which were outside the range [6]**

ANSWER:
```
1digit<--0
2digit<--0
3digit<--0
4digit<--0
OutsideRange<--0
FOR Count<--1 TO 5000
   INPUT Number
   IF Number>999 THEN 4digit<--4digit+1
   ELSE IF Number>99 THEN 3digit<--3digit+1
   ELSE IF Number>9 THEN 2digit<--2digit+1
   ELSE IF Number>=0 OR Number<=9 THEN 1digit<--1digit+1
   ELSE OutsideRange<--OutsideRange+1
NEXT
Outside_percent=(OutsideRange/5000)*100
OUTPUT 1digit, 2digit, 3digit, 4digit, Outside_percent
```

Write an algorithm using either pseudocode or a flowchart, to:

- **input a positive integer**
- **use this value to set up how many other numbers are to be input**
- **input these numbers**
- **calculate and output the total and the average of these numbers. [6]**

ANSWER:

```
INPUT NumberCount
Total<--0
FOR Count<--1 TO NumberCount
INPUT Number
Total<--Total+Number
NEXT
Avg=Total/NumberCount
OUTPUT Total, Avg
```

(a) Write an algorithm to input 1000 numbers. Count how many numbers are positive and how many numbers are zero. Then output the results. Use either pseudocode or a flowchart. [6]

ANSWER:

```
Count<--0
Positive<--0
Zero<--0
WHILE Count<1000 DO
  INPUT Number
  IF Number>0
  THEN Positive<--Positive+1
  ELSE IF Number=0
  THEN Zero<--Zero+1
  Count<--Count+1
ENDWHILE
OUTPUT Positive, "Positive Numbers"
OUTPUT Zero, "Zeros"
```

2 (a) Write an algorithm, using pseudocode, to input three different numbers, multiply the two larger numbers together and output the result. Use the variables: Number1, Number2 and Number3 for your numbers and Answer for your result. [5]

(b)Give two sets of test data to use with your algorithm in part (a) and explain why you chose each set. [4]

ANSWER:

```
REPEAT
 INPUT Number1, Number2, Number3
UNTIL Number1<>Number2 AND Number2<>Number3 AND Number1<>Number3
IF Number1<Number2 AND Number1<Number3
 THEN Answer<--Number2*Number3
ENDIF
IF Number2<Number1 AND Number2<Number3
 THEN Answer<--Number1*Number3
ENDIF
IF Number3<Number2 AND Number3<Number1
 THEN Answer<--Number2*Number1
ENDIF
OUTPUT "Answer=", Answer
```

A town contains 5000 houses. Each house owner must pay tax based on the value of the house. Houses over $200 000 pay 2% of their value in tax, houses over $100 000 pay 1.5% of their value in tax and houses over $50 000 pay 1% of their value in tax. All others pay no tax. Write an algorithm to solve the problem using pseudocode.

ANSWER:

```
FOR Count<--1 TO 5000
  INPUT House
  IF House>200000 THEN Tax=House*0.02
  ELSE IF House>100000 THEN Tax=House*0.015
  ELSE IF House>50000 THEN Tax=House*0.01
  ELSE Tax=0
  PRINT Tax
NEXT
```

Write an algorithm using pseudocode which takes temperatures input over a 100 day period (once per day) and output the number of days when the temperature was below 20C and the number of days when the temperature was 20C or above.

ANSWER:

```
Days_below<--0
Days_above<--0
FOR Count<--1 TO 100
  INPUT Temperature
  IF Temperature<20 THEN Days_below<--Days_below+1
  ELSE IF Temperature>20 THEN Days_above<--Days_above+1
NEXT
  OUTPUT Days_below, Days_above
```

Write down **two** types of conditional statement and for each one show how you would select someone whose age was either 10 or 20. [*6 marks*]

ANSWER:

1. IF…THEN…ELSE…ENDIF
2. CASE…OF…OTHERWISE…ENDCASE

```
For first one:
INPUT Age
IF Age=20
  THEN PRINT "Selected"
ENDIF
IF Age=10
  THEN PRINT "Selected"
ENDIF
```

```
For second one:
CASE Age OF
10:PRINT "Selected"
20:PRINT "Selected"
 OTHERWISE PRINT "Error"
ENDCASE
```

a Write down **three** different types of loop structure. [*3 marks*]

b Here are **three** problems. For each one, choose a different loop structure for your pseudocode solution.

i Input 10 numbers and print out their total. [*6 marks*]

ii Input some numbers and calculate their total, an input of –1 stops the process. [*6 marks*]

iii Input some numbers and calculate their total, stop when the total is greater than 20. [*6 marks*]

ANSWER:

a)
1. FOR…TO…NEXT
2. WHILE…DO…ENDWHILE
3. REPEAT…UNTIL

b)

i)
```
Total<--0
FOR Count<--1 TO 10
INPUT Number
Total<--Total+Number
NEXT Count
PRINT Total
```

ii)
```
Total<--0
INPUT Number
WHILE Number<>-1 DO
Total<--Total+Number
INPUT Number
ENDWHILE
PRINT Total
```

iii)
```
Total<--0
REPEAT
INPUT Number
Total<--Total+Number
UNTIL Total>20
PRINT Total
```

a Write a pseudocode algorithm to input 20 numbers and find the average of the positive numbers. [*7 marks*]

ANSWER:

```
Total<--0
PositiveCount<--0
FOR Count<--1 TO 20
  INPUT Number
  IF Number>0
  THEN Total<--Total+Number
       PositiveCount<--PositiveCount+1
NEXT Count
  Average=Total/PositiveCount
OUTPUT "Average of the positive numbers is", Average
```

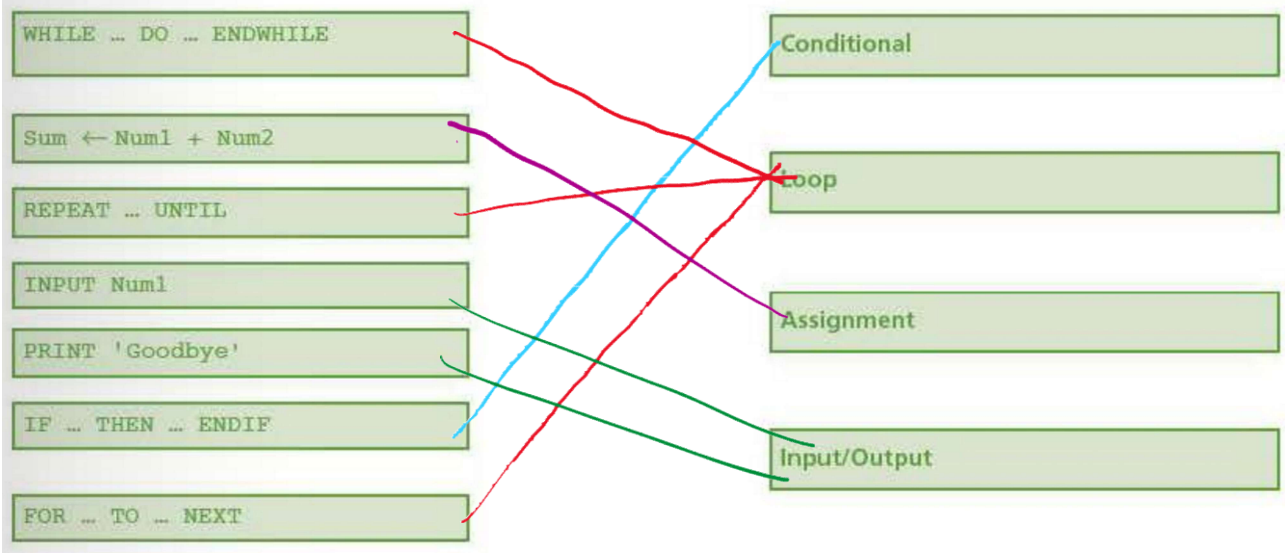
b Explain how you would change your algorithm to reject any negative numbers. [*2 marks*]

ANSWER:

Put the input statement in a `REPEAT UNTIL` loop that rejects any number less than zero.

Seven examples are shown on the left and four types of statement are shown on the right.

Draw lines to connect each type of statement to the example(s) it applies to. [*7 marks*]



(a)Write an algorithm, using pseudocode or otherwise which;
- **inputs 50 numbers.**
- **checks whether each number is in the range 1000 to 9999.**
- **outputs how many of the input numbers were out of range.**
- **outputs the percentage of input numbers which were out of range. [6]**

ANSWER:
```
OutRange<--0
FOR Count<--1 TO 50
 INPUT Number
 IF Number<1000 OR Number>9999
 THEN OutRange<--OutRange+1
NEXT Count
 Percent=(OutRange/50)*100
OUTPUT OutRange, Percent
```

9 A computer program is required which inputs 10 numbers, multiplies them together and finally outputs the answer (the product). The following algorithm has been written to do this.

```
1   count = 0
2   product = 0
3   while count <= 10 do
4       input number
5       product = product * number
6       count = count + 1
7       print product
8   endwhile
```

(a) There are three errors in the algorithm. Locate and describe these errors. [3]

(b) A while … do loop has been used in the algorithm. State another type of loop that could have been used. [1]

ANSWER:
```
a) Error 1: product=0 on line 2
   Description: Should have use product=1
   Error 2: count<=10 on line 3
   Description: count<10 should have been used or in line 1 count=1 should have been used.
   Error 3: print product on line 7 before endwhile
   Description: print product should be outside the loop after endwhile statement.
b) REPEAT              FOR count=1 TO 10
   UNTIL count=10      NEXT Count
```

Read 10 numbers and find sum of even numbers

ANSWER:
```
Sum<--0
FOR Count<--1 TO 10
 INPUT Number
 IF
 Number%2=0
 THEN Sum<--Sum+Number
NEXT Count
 OUTPUT Sum
```

A shop sells books, maps and magazines. Each item is identified by a unique 4 – digit code. All books have a code starting with 1, all maps have a code starting with 2 and all magazines have a code starting with 3. The code 9999 is used to end the algorithm.

Write an algorithm in the form of a pseudocode which inputs the codes for all items in stock and outputs the number of books, number of maps and the number of magazines in stock. Include any validation checks needed.

ANSWER:
```
Books<--0
Maps<--0
Magazines<--0
REPEAT
INPUT code
  IF code>999 AND code<2000
  THEN Books<--Books+1
  ELSE IF code>1999 AND code<3000
  THEN Maps<--Maps+1
  ELSE IF code>2999 AND code<4000
  THEN Magazines<--Magazines+1
  ELSE PRINT "Input Error"
 ENDIF
 ENDIF
 ENDIF
UNTIL code=9999
PRINT Books, Maps, Magazines
```

(8) A company are carrying out a survey by observing traffic at a road junction.
Each time a car, bus, lorry or other vehicle passed by the road junction it was noted down.
10 000 vehicles were counted during the survey.

Write an algorithm, using pseudocode, which:
- inputs all 10000 responses
- outputs the number of cars, buses and lorries that passed by the junction during the survey
- outputs the number of vehicles that weren't cars, buses or lorries during the survey **(10 points)**

ANSWER:

```
Car<--0
Lorry<--0
Bus<--0
Other<--0
FOR Count<--1 TO 10000
 INPUT vehicle
 IF vehicle=Car THEN Car<--Car+1
 ELSE IF vehicle=Lorry THEN Lorry<--Lorry+1
 ELSE IF vehicle=Bus THEN Bus<--Bus+1
 ELSE Other<--Other+1
NEXT Count
OUTPUT "The number of cars is", Car
OUTPUT "The number of lorries is", Lorry
OUTPUT "The number of buses is", Bus
OUTPUT "The number of vehicles that weren't cars,buses,lorries is", Other
```

(10) There are ten stations on a railway line:

1----2----3----4----5----6----7----8----9-----10

The train travels in both directions (i.e. from 1 to 10 and then from 10 to 1).
The fare between each station is $2.
A passenger inputs the number of the station at the start of his journey and the number of the destination station and the fare is calculated (e.g if a passenger gets on a station 3 and his destination is station 9 his fare will be $12). The calculation must take into account the direction of the train (e.g. a passenger getting on at station 7 and getting off at station 1 will also pay $12 and not a negative value!!). A discount of 10% is given if 3 or more passengers are travelling together.

Write an algorithm, using pseudocode, which:
- inputs the number of passengers travelling
- inputs the station number of the starting point and the station number of the destination
- calculates the total fare taking into account the direction of travel
- calculates any discount due
- outputs the cost of the tickets and prints the tickets **(10 points)**

ANSWER:

```
Total<--0
Ticket<--0
INPUT StartNum, DestinationNum, PasNum
CASE Direction OF
A: Ascending
B: Descending
OTHERWISE
ENDCASE
IF Direction=A AND PasNum>=3
THEN FareDiscount=(2*(DestinationNum-StartNum))*0.9
     Total<--Total+FareDiscount
     Ticket<--Ticket+PasNum
ENDIF
IF Direction=A AND PasNum<3
THEN Fare=2*(DestinationNum-StartNum)
     Total<--Total+Fare
     Ticket<--Ticket+PasNum
ENDIF
IF Direction=B AND PasNum>=3
THEN FareDiscount=(2*(StartNum-DestinationNum))*0.9
     Total<--Total+FareDiscount
     Ticket<--Ticket+PasNum
ENDIF
IF Direction=A AND PasNum<3
THEN Fare=2*(StartNum-DestinationNum)
     Total<--Total+Fare
     Ticket<--Ticket+PasNum
ENDIF
OUTPUT Total
PRINT Ticket, "Tickets"
```

Write an algorithm using pseudocode which:

inputs the top speeds of 5000 cars

outputs the fastest speed and the slowest speed

outputs the average speed of all the 5000 cars

```
Fastest<--0
Slowest<--1000
Total<--0
FOR Count<--1 TO 5000
 INPUT Speed
 IF Speed>Fastest
  THEN Fastest=Speed
 IF Speed<Slowest
  THEN Slowest=Speed
 Total<--Total+Speed
NEXT Count
Average=Total/5000
OUTPUT Fastest, Slowest, Average
```

A geography class decided to measure daily temperature and hours of sunshine per day over a 12 months period (365 days). Write pseudocode that inputs the temperature and hours of sunshine for all 365 days and give output as average temperature for the year and avg. number of hours per day over the year.

```
Total_temp<--0
Total_Shrs<--0
FOR Count<--1 to 365
INPUT temp
Total_temp<--Total_temp + temp
INPUT Shours
Total_Shrs<--Total_Shrs + Shours
NEXT Count
Avgtemp<--Total_temp/365
AvgShrs<--Total_Shrs/365
PRINT “Average temperature for whole year is:”, Avgtemp
PRINT “Average Sunshine hours for whole year is:”, AvgShrs
```