**Question05:**

**Algorithm :**1. Start  
2. Initialize total = 0 and an empty array 'numbers'.  
3. Loop:  
 a. Prompt user to enter an integer (0–9).  
 b. If input is invalid (<0, >9, or non-integer), break loop.  
 c. Otherwise, store input in numbers[total] and increment total.  
4. If total == 0, print "No number is inputted" and stop.  
5. Initialize an array 'occurrences[10]' with all zeros.  
6. For each number in 'numbers':  
 a. Increment occurrences[number].  
7. Print table header ("Number Occurrences").  
8. For i = 0 to 9:  
 a. If occurrences[i] > 0, print i and its count.  
9. Stop.  
  
**Pseudocode:**  
  
BEGIN  
 SET total = 0  
 CREATE array numbers[]  
 WHILE true DO  
 PROMPT "Enter any integer number (0-9):"  
 READ user\_input  
 IF user\_input < 0 OR user\_input > 9 OR invalid THEN  
 EXIT loop  
 ENDIF  
 numbers[total] = user\_input  
 total = total + 1  
 ENDWHILE  
  
 IF total == 0 THEN  
 PRINT "No number is inputted"  
 STOP  
 ENDIF  
  
 CREATE occurrences[10] = {0,0,0,0,0,0,0,0,0,0}  
 FOR i = 0 TO total-1 DO  
 occurrences[numbers[i]] = occurrences[numbers[i]] + 1  
 ENDFOR  
  
 PRINT "Number Occurrences"  
 FOR i = 0 TO 9 DO  
 IF occurrences[i] > 0 THEN  
 PRINT i, occurrences[i]  
 ENDIF  
 ENDFOR  
END  
 **IPO Chart**

|  |  |  |
| --- | --- | --- |
| **Input** | **Process** | **Output** |
| 1.Integer numbers between 0 and 9 entered by the user   2.Input ends when the user enters an invalid number (<0, >9, or non-integer) | Store valid inputs in an array.  Count occurrences of each number (0–9) using a frequency array.  Display results showing numbers and their respective counts. | A table showing each entered number and how many times it occurred |

**PAC chart**

|  |  |
| --- | --- |
| **Given Data** | **Required Result** |
| User enters integer numbers (0–9). Input ends when user enters invalid data (<0, >9, or non-integer) | Display how many times each number (0–9) occurred. If no valid input is given, print "No number is inputted |
| **Processing** | **Solution Alternatives** |
| 1. Initialize total = 0.  2. Read user input in a loop.  3. If input valid → store in array and increment total.  4. If input invalid → stop input.  5. If total == 0 → print "No number is inputted".  6. Else, initialize occurrences[10] = {0}.  7. Traverse numbers and increment occurrences.  8. Print table of number and occurrences | 1. Use fixed-size array (current solution).  2. Use hash map/dictionary to handle dynamic ranges.  3. Sort numbers and count consecutive duplicates |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

**C code**#include <stdio.h>

int print\_result(int \*numbers\_0\_pt,int total);

int main(){

    short int user\_input\_num;

    int total = 0 ;

    int numbers [total];

    int valid\_input ;

    while(1){

        printf("Enter any integer number (0-9) :");

        valid\_input = scanf("%d",&user\_input\_num);

        if(valid\_input == 0 || user\_input\_num > 9 || user\_input\_num < 0){

            break;

        }

        numbers[total] = user\_input\_num;

        total++;

    }

    print\_result(&numbers[0],total);

    return 0 ;

}

int print\_result(int \*numbers\_0\_pt,int total){

    if (total ==0){

        printf("No number is inputted");

        return 0;

    }

    int occurences[10]= {0,0,0,0,0,0,0,0,0,0}; //? Ordered/Sorted list of occurences of numbers from 0 - 9

    for (int i=0;i<total;i++){

        occurences[\*numbers\_0\_pt] += 1;

        numbers\_0\_pt++;

    }

    int \*occ\_pt = &occurences[0];

    printf("Number \t Number of Occurrences\n");

    for (int i=0;i<10;i++){

        int value = \*occ\_pt;

        if (value == 0){

            occ\_pt++;

            continue;

        }

        printf("%d \t %d\n",i,value);

        occ\_pt++;

    }

    return 0;

}