Assignment: A Doctor Survey

A Doctor Survey results is stored in 2 arrays. First array represents all the doctors ids (working and non -working both). The second array represents only working doctor's id. Please find the doctor ids who are not working. Write a Pseudocode for this scenario.

Input and Output Format:

First line of input corresponds to n1, the size of the first array and next n1 lines correspond to the elements of the first array.

The next line corresponds to n2, the size of the second array and next n2 lines correspond to the elements of the second array

Output is the id's of doctor who are not working

Note:

Print "Invalid array size" when size of the array is a negative number or zero and terminate the program

Print "Invalid id" when there is any negative numbers or zero available in the input array and terminate the program.

Sample Input 1 :

Enter the size of first array

5

Enter the values

109

117

120

149

124

Enter the size of second array

3

Enter the values

120

109

149

Sample Output :

Not Working IDs

117

124

Sample Input 2 :

Enter the size of first array

0

Sample Output 2 :

Invalid array size

Explanation for Pseudocode :

Let the array names be id1, id2.

Let n1, n2 be the size of the 2 arrays.

Declare variables n1, n2, id1[20], id2[20].

Also declare variables id1status and id2status to track if any negative or zero in id1 or id2

```
DECLARE variables n1,n2,id1[20],id2[20],id1status,id2status
SET id1status = 0,id2status = 0
```

Next read size

```
READ n1, n2
```

IF n1<=0 && n2<=0 THEN display "Invalid array size"

```
IF n1<=0 && n2<=0
    PRINT  "Invalid array size"
ELSE
```

Else iterate and get the input for id1 and id2

For id1 loop from 0 to n1-1

```
FOR i in 0 to n1-1 DO
    READ id1[i];
```

Check if the input is invalid

```
        IF id1[i] <= 0 THEN
            id1status=1
            BREAK
        ENDIF
    ENDFOR
```

We break the loop because there is no meaning in getting further input, because id cannot be 0.

Next read input for id2. Get for id2 only if id1status=0. That is all values in id1 are valid.

```
    IF id1status ==0 THEN
        FOR i in 0 to n2-1 DO
            READ id2[i]
```

Check if the input is invalid

```
                IF id2[i] <= 0 THEN
                    id2status=1
                    BREAK
                ENDIF
            ENDFOR
        END IF
```

If id1status=0 and id2status=0, then all input are valid.
If the value in id1 array does not exist in id2 array then that id is not working.

To check this

```
    IF id1status==0 && id2status==0 THEN
        FOR i in 0 to n1-1                  ------> To iterate id1
            SET count=0                     -----> To track if the value in id1 is present
            FOR j in 0 to n2-1              -----> To iterate id2
```

```
            IF id1[i]==id2[j] THEN          ------> checks the value in id1 with each value in id2
                count = count+1;
                BREAK;                      ------> As value in id1 is present in id2 reset count and break the loop
            END IF
        END FOR                             -------> End loop for id2
        IF count==0 THEN                    -------> If count is 0, id not found, hence the id is not working
            PRINT id1[i]
        END IF
    END FOR                                 -------> End loop for id1
ENDIF                                       -------> End the check for id1status and id2status
```

Last terminate the pseudocode ***END***

Submission status

Submission status	Submitted for grading
Grading status	Not graded
Last modified	Sunday, 14 January 2024, 1:54 PM

Online text

```
BEGIN

DECLARE variables n1,n2,id1[20],id2[20],id1status,id2status
    SET id1status = 0,id2status = 0
   READ n1, n2
   IF n1<=0 && n2<=0
      PRINT  "Invalid array size"
   ELSE
    FOR i in 0 to n1-1 DO
      READ id1[i];
       IF id1[i] <= 0 THEN
          id1status=1
          BREAK
       ENDIF
    ENDFOR
   IF id1status ==0 THEN
      FOR i in 0 to n2-1 DO
         READ id2[i]
             IF id2[i] <= 0 THEN
                  id2status=1
                  BREAK
             ENDIF
        ENDFOR
     END IF
   IF id1status==0 && id2status==0 THEN
      FOR i in 0 to n1-1
         SET count=0
         FOR j in 0 to n2-1
              IF id1[i]==id2[j] THEN
```

```
                    count = count+1;
                    BREAK;
                END IF
            END FOR
            IF count==0 THEN
                PRINT id1[i]
            END IF
        END FOR
    ENDIF

END
```

Submission comments Show comments ▸ Comments (0)

Edit submission Remove submission

You can still make changes to your submission.

◄ Sweet Stall

Jump to...

Store the marks of students ►