Practice Arena

Practice problems aimed to improve your coding skills.

- PRACTICE-02_SCAN-PRINT
- PRACTICE-03_TYPES
- LAB-PRAC-02_SCAN-PRINT
- LAB-PRAC-01
- PRACTICE-04_COND
- BONUS-PRAC-02
- LAB-PRAC-03_TYPES
- PRACTICE-05_COND-LOOPS
- LAB-PRAC-04_COND
- LAB-PRAC-05_CONDLOOPS
- PRACTICE-07_LOOPS-ARR
- LAB-PRAC-06_LOOPS
- LAB-PRAC-07_LOOPS-ARR
  - Home Alone
  - Arrangements with Arrays
  - Overlapping Patterns
  - Lucky Draw
  - Diamond Array
  - Linear Leap
  - Candy Crush
  - Nested Safes
  - Heros Arc
  - Linear Loopy Maze
  - Histogram Heights
  - Changing Times
- LABEXAM-PRAC-01_MIDSEM
- PRACTICE-09_PTR-MAT
- LAB-PRAC-08_ARR-STR
- PRACTICE-10_MAT-FUN
- LAB-PRAC-09_PTR-MAT
- LAB-PRAC-10_MAT-FUN
- PRACTICE-11_FUN-PTR
- LAB-PRAC-11_FUN-PTR
- LAB-PRAC-12_FUN-STRUC
- LABEXAM-PRAC-02_ENDSEM
- LAB-PRAC-13_STRUC-NUM
- LAB-PRAC-14_SORT-MISC

Linear Loopy Maze

LAB-PRAC-07_LOOPS-ARR

Linear Loopy Maze [10 marks]

Problem Statement
Mr C is caught in a linear maze. Help him find out if he can get out of the maze or not. In the first line of the input you will be given a **strictly positive integer** n. We promise that n will be less than or equal to 20. In the next line, we will give you n **non-negative** integers, separated by a single space. Store these numbers in an array. Lets call this array maze.

Mr C will land on the first element of maze i.e. subscript 0 of the array. The value maze[0] at that subscript will tell Mr C which subscript to land next i.e. in the next step Mr C will land on the maze[0]-th subscript of the array maze. The value stored there will tell him which subscript to land next. The exit of this maze is the last location of the array i.e. the subscript (n-1).

1. If following the above procedure, Mr C ever lands on the exit subscript, print "EXIT" (without quotes) followed by a space followed by the number of steps it took to reach the exit location for the first time.
2. If following the above procedure Mr C ever lands on a subscript which tells him to go to an illegal subscript in the array, print "BAD" (without quotes) followed by a space followed by the bad subscript.
3. If following the above procedure Mr C is never asked to go to an illegal subscript, but he can never hope to land on the exit subscript either, print "TRAP" (without quotes).

Caution

1. Be careful about spelling errors, extra/missing lines and extra/missing spaces.
2. When calculating the number of steps taken to reach the exit subscript (if reachable at all), the act of landing on the subscript 0 of the maze at the very beginning is to be considered the first step.
3. If Mr C ever reaches the exit subscript, he simply exits, he does not have to read the value at the exit subscript and continue this process anymore.

EXAMPLE 1:
INPUT
4
1 0 3 2

OUTPUT:
TRAP

Explanation: Mr C will land on subscript 0 in the array which contains 1 and will tell him to go to subscript 1. But there he will be told to go to subscript 0 and so he will get into an infinite loop and never hope to reach the exit subscript 3.

EXAMPLE 2:
INPUT

6
1 2 3 4 5 0

OUTPUT:
EXIT 6

Explanation:
Step 1: land on subscript 0
Step 2: land on subscript 1
Step 3: land on subscript 2
Step 4: land on subscript 3
Step 5: land on subscript 4
Step 6: land on subscript 5
and subscript 5 is the exit subscript

EXAMPLE 2:
INPUT
5
1 22 4 2 0

OUTPUT:
BAD 22

Explanation: In the third step Mr C will be asked to go to the subscript 22 which is out of bounds of the array.

Grading Scheme:
Total marks: **[10 Points]**

There will be no partial grading in this question. An exact match will receive full marks whereas an incomplete match will receive 0 points. Please be careful of missing/extra spaces and missing/lines (take help of visible test cases). Each visible test case is worth 1 point and each hidden test case is worth 2 points. There are 2 visible and 4 hidden test cases.

Start Solving! (/editor/practice/6146)