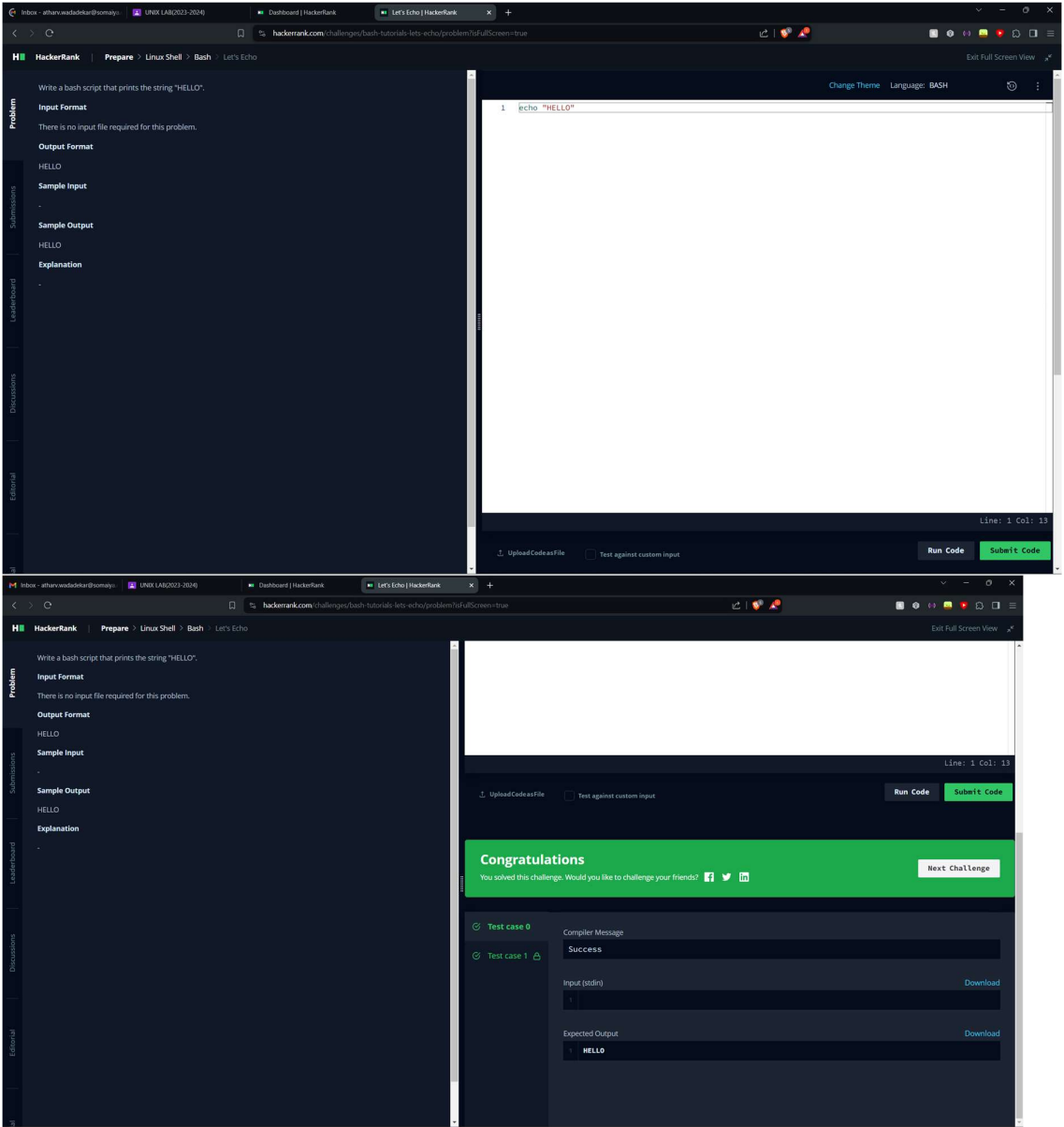


Name: Atharv Wadadekar

Roll No.:69

Objectives: To complete at least 5 challenges (2 Easy, 2 Medium, 1 Hard) on Unix Shell scripting via HackerRank.

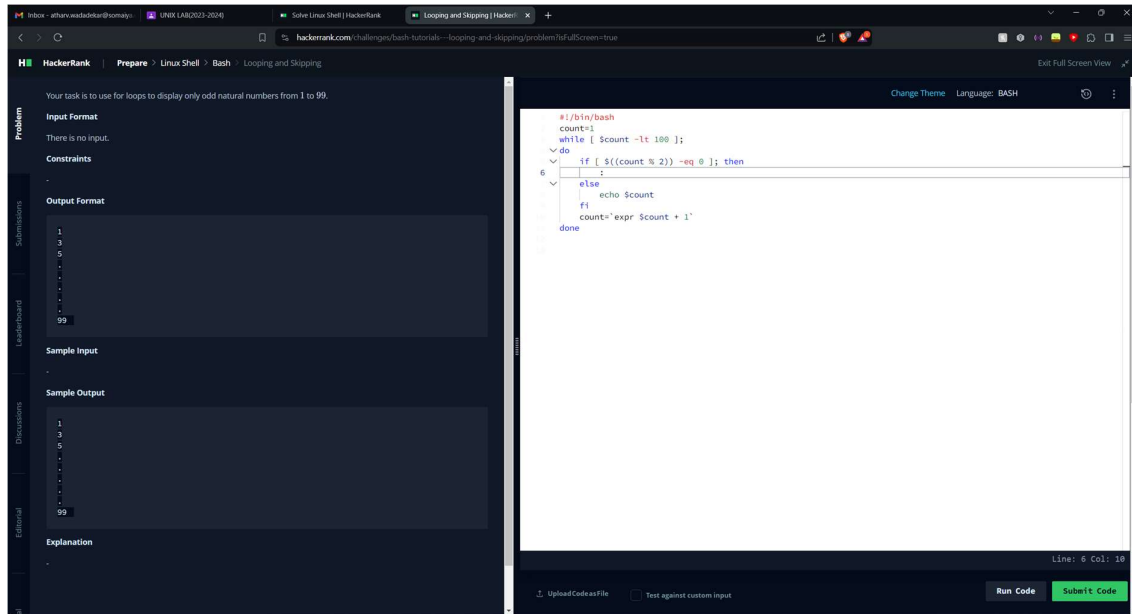
1. Easy Challenge 1



Prepare > Linux Shell > Bash > Let's Echo > Submissions

Let's Echo ★					
Problem Submissions Leaderboard Discussions Editorial Tutorial					
RESULT	SCORE	LANGUAGE	TIME		
Accepted	1.0	BASH	an hour ago	View Results	

2. Easy Challenge 2



Problem

Your task is to use for loops to display only odd natural numbers from 1 to 99.

Input Format

There is no input.

Constraints

-

Output Format

1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
51
53
55
57
59
61
63
65
67
69
71
73
75
77
79
81
83
85
87
89
91
93
95
97
99

Sample Input

-

Sample Output

1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
51
53
55
57
59
61
63
65
67
69
71
73
75
77
79
81
83
85
87
89
91
93
95
97
99

Explanation

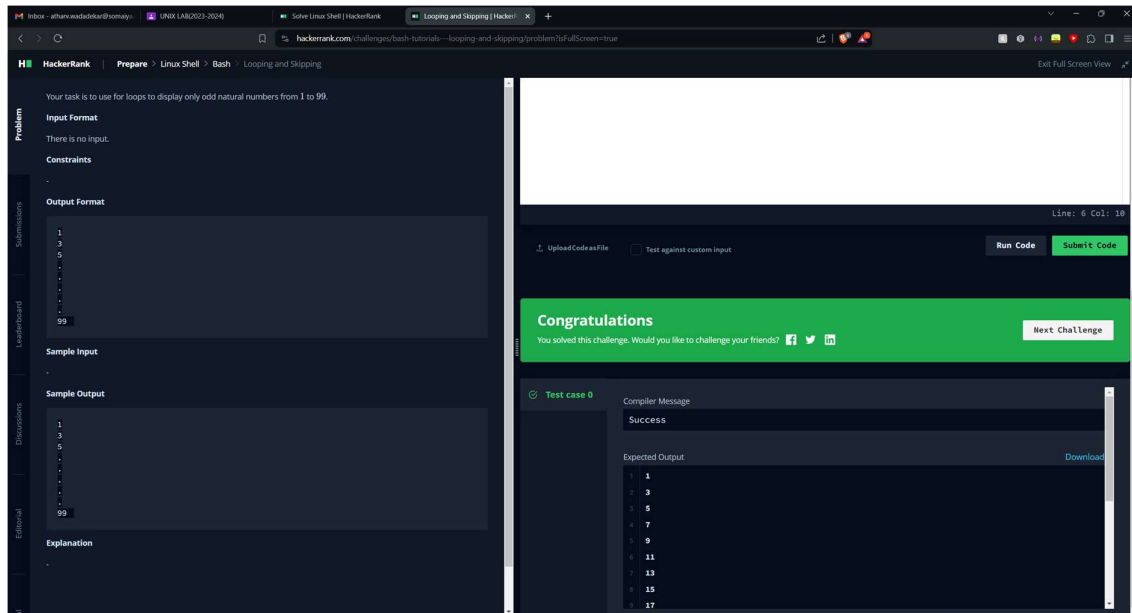
-

```
#!/bin/bash
count=1
while [ $count -lt 100 ];
do
    if [ $((count % 2)) -eq 0 ]; then
        continue
    else
        echo $count
        count=$((count + 1))
    fi
done
```

Line: 6 Col: 38

Upload Code as File ☐ Test against custom input

Run Code Submit Code



Problem

Your task is to use for loops to display only odd natural numbers from 1 to 99.

Input Format

There is no input.

Constraints

-

Output Format

1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
51
53
55
57
59
61
63
65
67
69
71
73
75
77
79
81
83
85
87
89
91
93
95
97
99

Sample Input

-

Sample Output

1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
51
53
55
57
59
61
63
65
67
69
71
73
75
77
79
81
83
85
87
89
91
93
95
97
99

Explanation

-

Line: 6 Col: 38

Upload Code as File ☐ Test against custom input

Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends? [Facebook](#) [Twitter](#) [LinkedIn](#) [Next Challenge](#)

Test case 0

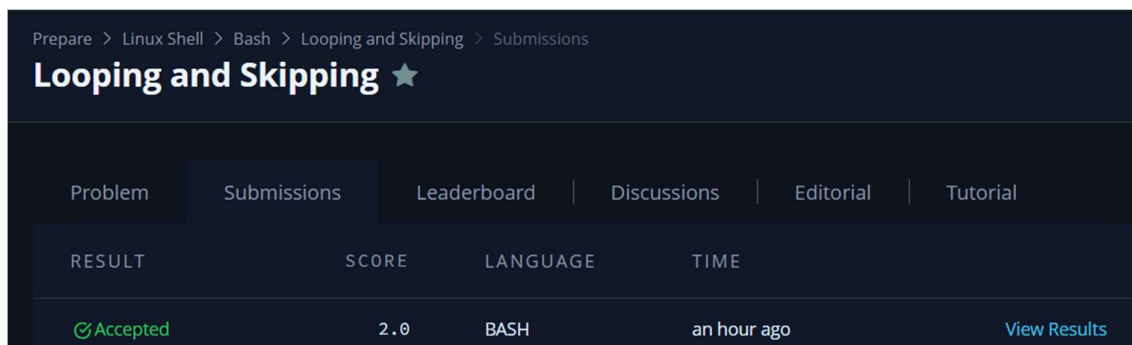
Compiler Message

Success

Expected Output

Download

1
3
5
7
9
11
13
15
17



Prepare > Linux Shell > Bash > Looping and Skipping > Submissions

Looping and Skipping ★

Problem Submissions Leaderboard Discussions Editorial Tutorial

RESULT	SCORE	LANGUAGE	TIME
Accepted	2.0	BASH	an hour ago

[View Results](#)

3. Medium Challenge 1

Problem

Submissions

Leaderboard

Discussions

Editorial

Given N integers, compute their average, rounded to three decimal places.

Input Format
The first line contains an integer, N .
Each of the following N lines contains a single integer.

Output Format
Display the average of the N integers, rounded off to three decimal places.

Input Constraints
 $1 \leq N \leq 500$
 $-10000 \leq x \leq 10000$ (x refers to elements of the list of integers for which the average is to be computed)

Sample Input

```
4
1
2
9
8
```

Sample Output

```
5.000
```

Explanation
The '4' in the first line indicates that there are four integers whose average is to be computed.
The average = $(1 + 2 + 9 + 8) / 4 = 20 / 4 = 5.000$ (correct to three decimal places).
Please include the zeroes even if they are redundant (e.g. 0.000 instead of 0).

#!/bin/bash

read n

sum=0

for ((i=0; i<n; i++)); do

read num

sum=\$((sum + num))

done

average=\$((awk "BEGIN {printf \"%.3f\", \$sum / \$n}"))

echo \$average

Upload Code as File

Test against custom input

Run Code

Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

Test case 0

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Compiler Message

Success

Input (stdin)

Download

1 100

2 9829

3 8093

4 8839

5 4794

6 7444

7 7599

8 1684

9 9283

Prepare > Linux Shell > Bash > Compute the Average > Submissions

Compute the Average ★

Problem

Submissions

Leaderboard

Discussions

Editorial

RESULT

SCORE

LANGUAGE

TIME

Accepted

4.0

BASH

6 minutes ago

View Results

4. Medium Challenge 2

The screenshot shows the HackerRank interface for the 'Arithmetic Operations' challenge. On the left, the problem description states: 'A mathematical expression containing +, *, ^, / and parenthesis will be provided. Read in the expression, then evaluate it. Display the result rounded to 3 decimal places.' Constraints specify that all numeric values are ≤ 999 . Sample inputs and outputs are listed: Sample Input 1 is $5+50*3/20 + (19*2)/7$ with Sample Output 1 being 17.929; Sample Input 2 is $-185+50*3/20 + (19*2)/7$ with Sample Output 2 being -45.929; Sample Input 3 is $(-185.5*7+50*3)/20 + (19*2)/7$ with Sample Output 3 being 22.146. On the right, a Bash script is shown in a code editor, using `bc` to evaluate the expression and `printf` to format the result to three decimal places. The script is as follows:

```
#!/bin/bash
read expression
5 result=$(echo "scale=4; $expression" | bc)
if [ -z "$result" ]; then
    echo "Error: Invalid expression"
elif [ "$result" = "0" ]; then
    echo "Error: Division by zero"
else
    printf "%.3f" $result
fi
```

The screenshot shows the HackerRank interface after successfully solving the challenge. A green banner displays 'Congratulations' and 'You solved this challenge. Would you like to challenge your friends?'. Below this, a list of test cases is shown, all marked as successful. The details for Test case 0 are expanded, showing the input `5+50*3/20 + (19*2)/7` and the expected output `17.929`. The interface also includes buttons for 'Run Code', 'Submit Code', and 'Next Challenge'.

The screenshot shows the HackerRank interface displaying the submission results for the 'Arithmetic Operations' challenge. The submission is listed as 'Accepted' with a score of 3.0, using the BASH language, and was submitted 2 minutes ago. The interface also includes a navigation bar with links to 'Problem', 'Submissions', 'Leaderboard', 'Discussions', 'Editorial', and 'Tutorial'.

RESULT	SCORE	LANGUAGE	TIME	
Accepted	3.0	BASH	2 minutes ago	View Results

5. Hard Challenge 1

The screenshot displays the HackerRank interface for the 'Lonely Integer - Bash!' challenge. The left sidebar contains the problem description, which states: 'There are N integers in an array A . All but one integer occur in pairs. Your task is to find the number that occurs only once.' It also provides the input format, constraints ($1 \leq N < 100$, $N \% 2 = 1$, $0 \leq A[i] \leq 100$), and sample inputs/outputs. The main area shows a Bash script solution that reads the input, iterates through the array, and prints the element that occurs only once. Below the script, a green banner reads 'Congratulations' and 'You solved this challenge. Would you like to challenge your friends?'. The bottom section shows the submission details, including the problem name, score, language, and time.

```
#!/bin/bash
read count
read -a array
result=0
for element in "${array[@]"; do
    result=$((result ^ element))
done
echo "$result"
```

Problem | **Submissions** | **Leaderboard** | **Discussions**

Lonely Integer - Bash! ★

Prepare > Linux Shell > Arrays in Bash > Lonely Integer - Bash! > Submissions

RESULT	SCORE	LANGUAGE	TIME
Accepted	10.0	BASH	21 minutes ago

[View Results](#)

Score:

The screenshot shows the HackerRank interface for the 'Linux Shell' challenge. The user's score is 20 points and their rank is 64068. The page lists five challenges, all of which have been solved:

- Let's Echo** (Easy, Bash (Basic), Max Score: 1, Success Rate: 97.67%) - Solved
- Looping and Skipping** (Easy, Bash (Basic), Max Score: 3, Success Rate: 95.72%) - Solved
- Arithmetic Operations** (Medium, Bash (Basic), Max Score: 3, Success Rate: 92.07%) - Solved
- Compute the Average** (Medium, Bash (Basic), Max Score: 4, Success Rate: 90.26%) - Solved
- Lonely Integer - Bash!** (Hard, Max Score: 10, Success Rate: 97.83%) - Solved

On the right side, there are filters for STATUS (Solved, Unsolved), SKILLS (Bash (Basic)), DIFFICULTY (Easy, Medium, Hard), and SUBDOMAINS (Bash, Text Processing, Arrays in Bash, Grep Sed Awk).

Outcomes: Able to proficiently solve diverse Unix Shell scripting challenges, showcasing problem-solving skills.