SCSE21081 - Web-based Programming Development Environment with Automatic Grading

Li Pingrui
School of Computer Science and Engineering

Assoc Prof. Hui Siu Cheung
School of Computer Science and Engineering

Abstract - In recent years, more and more people have learned programming and practiced programming skills with the help of various online programming platforms. Most of such web-based online programming platforms provide the functionalities of compiling and running code submitted by users in cloud servers. Execution results or error messages generated during execution are usually returned to the client for verification and debugging purposes.

However, such platforms usually have limited support for interaction between the user and programs in execution. Most of the systems are only able to execute user code with pre-defined test cases, which may or may not be transparent to users. Some platforms allow users to input their own test case in a specific format, and take in all input as a whole data before code execution.

Therefore, this project aims to investigate techniques that can be used to implement interactive input and output for an online programming platform. The system will prompt users to input and wait for acceptable input characters. Input is from the keyboard with the user during code execution in order to achieve better interaction.

This project is developed based on an existing online programming platform, Automated Programming Assessment System (APAS). APAS is a browser-based web application that is used for NTU students to do programming exercises, make submissions and perform automated judgement. APAS is developed with the Django framework.

Keywords – Software, System development, Python

1 INTRODUCTION

1.1 BACKGROUND

As programming is becoming a preferred, or even required skill for jobs of more and more disciplines, the needs of students to take programming courses arose drastically in the recent years. For students to learn programming efficiently, the best way is to provide them with more opportunities of hands-on programming experience. Hence, an online coding platform could play an important role in helping students to learn efficiently, as well as helping to reduce the burden of reading through and grading code submitted by students for lecturers and tutors.

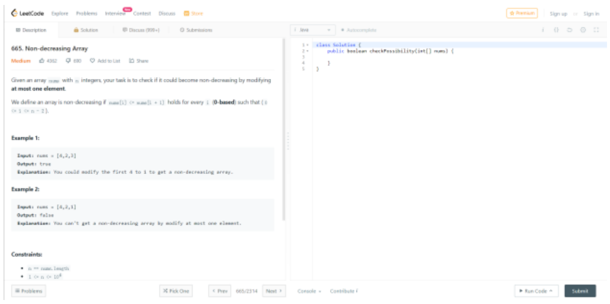
At the School of Computer Science and Engineering in Nanyang Technological University, there is currently a platform in use, which is called Automated Programming Assessment System (APAS). APAS is a web-based systems, that has the functionality of allowing students to write and run code, doing and submit coding assignment and quizzes.

2 INPUT STYLES

2.1 TRADITIONAL STYLE

The traditional style for online coding platform is that the system will take in all input as a whole data, and then perform code execution.

Some popular online coding platforms adopting this traditional style are listed below:



665. Non-decreasing Array

Medium 4362 690 Add to List Share

Given an array `nums` with `n` integers, your task is to check if it could become non-decreasing by modifying **at most one element**.

We define an array is non-decreasing if `nums[i] <= nums[i + 1]` holds for every `i` (**0-based**) such that (`0 <= i <= n - 2`).

Example 1:

```
Input: nums = [4,2,3]
Output: true
Explanation: You could modify the first 4 to 1 to get a non-decreasing array.
```

Example 2:

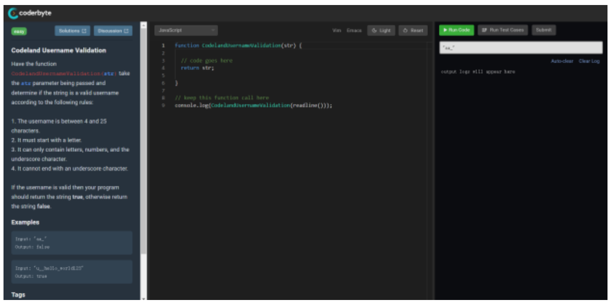
```
Input: nums = [4,2,1]
Output: false
Explanation: You can't get a non-decreasing array by modify at most one element.
```

Constraints:

- `n == nums.length`
- `1 <= n <= 10^4`

```
class Solution {
    public boolean checkPossibility(int[] nums) {
        
    }
}
```

Figure 1. Screenshot of leetcode.com



Codeland Username Validation

Have the function `CodelandUsernameValidation(str)` take the `str` parameter being passed and determine if the string is a valid username according to the following rules:

1. The username is between 4 and 25 characters.
2. It must start with a letter.
3. It can only contain letters, numbers, and the underscore character.
4. It cannot end with an underscore character.

If the username is valid then your program should return the string **true**, otherwise return the string **false**.

Examples

```
Input: "aa_"
Output: false
```

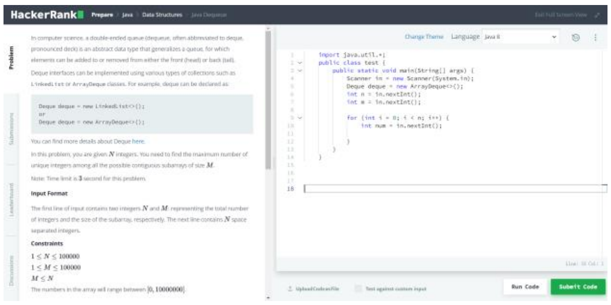
```
Input: "u__hello_world123"
Output: true
```

```
function CodelandUsernameValidation(str) {

  // code goes here
  return str;

}

// keep this function call here
console.log(CodelandUsernameValidation(readline()));
```

```
"aa_"
```

```
output logs will appear here
```

Figure 2. Screenshot of coderbyte.com



In computer science, a double-ended queue (dequeue, often abbreviated to deque, pronounced deck) is an abstract data type that generalizes a queue, for which elements can be added to or removed from either the front (head) or back (tail).

Deque interfaces can be implemented using various types of collections such as `LinkedList` or `ArrayDeque` classes. For example, deque can be declared as:

```
Deque deque = new LinkedList<>();
or
Deque deque = new ArrayDeque<>();
```

You can find more details about Deque here.

In this problem, you are given N integers. You need to find the maximum number of unique integers among all the possible contiguous subarrays of size M.

Note: Time limit is 3 second for this problem.

Input Format

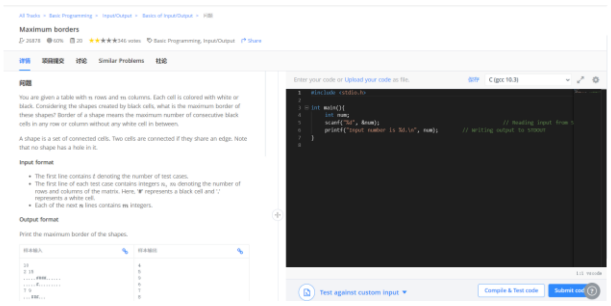
The first line of input contains two integers N and M: representing the total number of integers and the size of the subarray, respectively. The next line contains N space separated integers.

Constraints

$1 \leq N \leq 100000$
$1 \leq M \leq 100000$
$M \leq N$
The numbers in the array will range between $[0, 10000000]$.

```java
import java.util.*;
public class test {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        Deque deque = new ArrayDeque<>();
        int n = in.nextInt();
        int m = in.nextInt();

        for (int i = 0; i < n; i++) {
            int num = in.nextInt();

        }
    }
}
```

Figure 3. Screenshot of hackerrank.com


All Tracks > Basic Programming > Input/Output > Basics of Input/Output > 问题
26878
60%
20
346 votes
Basic Programming, Input/Output
Share
详情
等级提交
讨论
Similar Problems
社论
Enter your code or Upload your code as file.
保存
C (gcc 10.3)
1:1 vscode
Test against custom input
Compile & Test code
Submit code

Maximum borders

问题

You are given a table with n rows and m columns. Each cell is colored with white or black. Considering the shapes created by black cells, what is the maximum border of these shapes? Border of a shape means the maximum number of consecutive black cells in any row or column without any white cell in between.

A shape is a set of connected cells. Two cells are connected if they share an edge. Note that no shape has a hole in it.

Input format

- The first line contains t denoting the number of test cases.
- The first line of each test case contains integers n, m denoting the number of rows and columns of the matrix. Here, '#' represents a black cell and '.' represents a white cell.
- Each of the next n lines contains m integers.

Output format

Print the maximum border of the shapes.

样本输入	样本输出
10 2 15 .....####...... ......#........ 7 9 ...###...	4 5 9 6 7 8

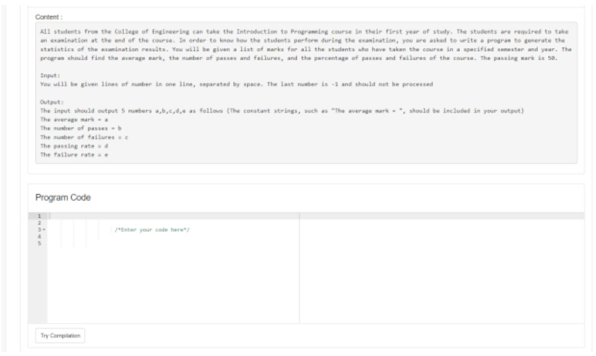
```c
#include <stdio.h>

int main(){
    int num;
    scanf("%d", &num);                              // Reading input from S
    printf("Input number is %d.\n", num);       // Writing output to STDOUT
}
```

Figure 4. Screenshot of hackerearth.com

Platforms adopting this input style usually take in all code written by the user as a whole, then compile and run the code with their pre-defined test cases, and finally display the execution result. This input style is simple and convenient for experienced programmers, because they are freed from repeatedly writing similar code to handle input and output.

However, for students who are new to programming, it is always trivial to make use of the standard input and output, so that they can add input or output statements anywhere in the program for debugging purpose.

APAS adopts this traditional input style so far. Students need to write code in the editor, the code will be sent to backend server as text. The server then concatenate the code snippet with the code template, compile and execute the complete concatenated code. Any compilation or runtime error will be returned to users for debugging purpose. If nothing goes wrong, all the output generated during execution are send as a whole to users for display.



Content :

```
All students from the College of Engineering can take the Introduction to Programming course in their first year of study. The students are required to take
an examination at the end of the course. In order to know how the students perform during the examination, you are asked to write a program to generate the
statistics of the examination results. You will be given a list of marks for all the students who have taken the course in a specified semester and year. The
program should find the average mark, the number of passes and failures, and the percentage of passes and failures of the course. The passing mark is 50.

Input:
You will be given lines of number in one line, separated by space. The last number is -1 and should not be processed

Output:
The input should output 5 numbers a,b,c,d,e as follows (The constant strings, such as "The average mark = ", should be included in your output)
The average mark = a
The number of passes = b
The number of failures = c
The passing rate = d
The failure rate = e
```

Program Code

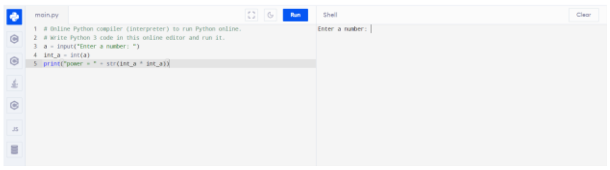
```
/*Enter your code here*/
```

Figure 5. Screenshot of APAS

2.2 INTERACTIVE STYLE

Some platforms adopt the interactive input style, which simulates running code in console. The program prompts user to input and waits for acceptable input characters. Input is from the keyboard with the user in order to achieve better interaction.

One of such platforms is programiz as shown below. The demonstration uses a simple code snippet which prompts user to enter a number, calculate the power of the number, format the result and print out.

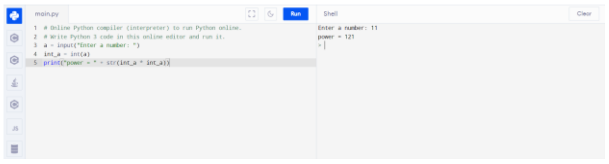


```python
# Online Python compiler (interpreter) to run Python online.
# Write Python 3 code in this online editor and run it.
a = input("Enter a number: ")
int_a = int(a)
print("power = " + str(int_a * int_a))
```

```
Enter a number: 
```

Figure 6. Code snippet on programiz.com

As shown in Figure 6, the program hangs until user provide input. After user provide input, the program continues to execute and produce output accordingly. In such case, the user could locate bugs more easily.



```python
# Online Python compiler (interpreter) to run Python online.
# Write Python 3 code in this online editor and run it.
a = input("Enter a number: ")
int_a = int(a)
print("power = " + str(int_a * int_a))
```

```
Enter a number: 11
power = 121
> 
```

Figure 7. Execution result

3 TECHNIQUES

In this section, some techniques that could probably help to implement interactive input style are investigated.

3.1 WEB-SOCKET

In the case of interactive input style, there is frequent 2-way communication between client and backend server. Any user input should be sent to the server, so that the program can continue to execute. Then all outputs are sent back to the client for display.

In such scenario, web socket could perform better than HTTP. In the use case of normal HTTP, each HTTP connection requires an underlying TCP connection between the server and the client. If the client polls the server, in this specific case, for execution result, there will be a new TCP connection being set up for each incoming message. With web socket, traffic in both direction, from the server to the client and from the client to the server, shares a single TCP connection.

Another benefit of web socket is that after the connection is set up, the overhead of HTTP head attached to each transmitted message can be saved [1].

The following image demonstrates the communication process using web socket. The connection is set up with 2 steps. Stage 1 step up a normal TCP connection though 3-way handshake. Stage 2 is initiated by sending a web socket upgrade request over an HTTP GET request. If the other side replies with a web socket upgrade request, then the connection is set up successfully and is kept alive until terminated by either side of communication.

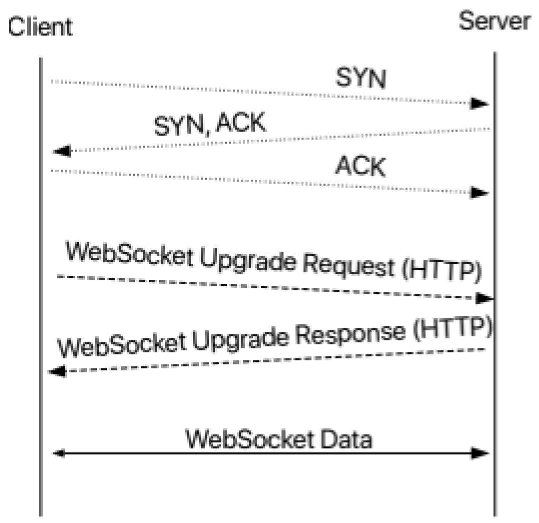

Client
Server
SYN
SYN, ACK
ACK
WebSocket Upgrade Request (HTTP)
WebSocket Upgrade Response (HTTP)
WebSocket Data

Figure 8. Communication process of web socket [2]

A performance test carried out at University of Zagreb has proven that although the latency and amount of network traffic generated by web socket communication is a bit higher than the most underlying TCP connection, web socket still out-performs HTTP connection [2], which is used in current version of APAS. Thus, due to the stated reasons, to implement the feature of interactive input for APAS, web socket is a technique that is worth discussing.

4 IMPLEMENTATION DIFFICULTIES

4.1 TRANSMIT CODE

The data transmitted between client and server are in JSON format, which is indeed plaintext and have considerable overhead. Moreover, parsing the plaintext into executable code and data to input to the program could incur more parsing overhead if there is more user interaction involved in the program.

4.1 HANG PROGRAM EXECUTION

To implement interactive input, we must come up with a method to make the program hangs for user input to be sent from the browser. In current implementation, the code submitted by user is inserted to code template and formed a whole executable code snippet with pre-defined or user input test cases. The code and then be submitted to thread pool for execution. However, to simulate the console, the code is not fed with test data and should hang up for user input, which is not easy to implement with current codebase.

One method to achieve this feature is to use a file as an intermediate. However, read from a file is not like console input. If the file doesn’t have any content, the reading stream will be terminated immediately and return a null value to the caller. Hence, if the code is not tailored carefully for executing in such an interactive manner, the result may be unexpected. However, this deviated from our original intention of helping students who are new to programming.

5 FUTURE WORKS

Section 4.1 discussed the difficulty of message exchanging and overhead of parsing, which is more about performance consideration. There are existing packages to format code from plaintext that might be helpful.

Section 4.2 discussed more about the difficulties to realize functional requirement of interactive input, which is more important. Some methods were tested out and explained for others to reference.

6 CONCLUSION

In this project, the feature of interactive styled input and feasibility of implementation for current ASAP was investigated. Two techniques that could be used in the implementation was studied. The difficulties of implementation was discussed and some relevant work that might be carried out in the future was discussed.

ACKNOWLEDGMENT.

I would like to acknowledge the funding support from Nanyang Technological University – URECA Undergraduate Research Programme for this research project.

I would like to acknowledge Assoc Prof. Hui Siu Cheung for his generous support, advice and guidance throughout the process of this project.

REFERENCES

[1] Fette, I. and A. Melnikov, "The WebSocket Protocol", RFC 6455, DOI 10.17487/RFC6455, December 2011. https://www.rfc-editor.org/info/rfc6455.

[2] D. Skvorc, M. Horvat and S. Srbljic, "Performance evaluation of Websocket protocol for implementation of full-duplex web streams," 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014, pp. 1003-1008, doi: 10.1109/MIPRO.2014.6859715.