This Problem statement includes **Context**, **Specifications**, **Description** and **Hints**

Context: One of the most common approaches for data extraction from text is through regular expressions(also known as 'regex').

While regular expressions are a practical and efficient approach, they come with some pitfalls. One of these comes in the form of a "bad regular expression". These cause the control to be lost in a backtracking limbo.

Suppose you were to develop a system that handles a large amount of string matching requests, which may include bad regexes as input. How would you exit the backtracking limbo?

Specifications:

1) POST api:

input JSON:

```
{
        "regex": “”,
        "textBody": ""
}
```

output json:

```
{
"match": “”,
“error”: false
}
```

Description: -

- "match" should contain the value picked up by the regex, for the sake of convenience just include the first match.
- If a match isn't found for that particular regex, the value should be null.
- If the regex is a bad regex then appropriately exit the matching process, the value should be null and error should be true.

Hints: -

- While stuck in backtracking, the concerned thread or process may become unresponsive, it is crucial to eliminate such fatal states.