Problem 2710: Remove Trailing Zeros From a String

Problem Information

Difficulty: Easy
Acceptance Rate: 0.00%
Paid Only: No

Problem Description

Given a

positive

integer

num

represented as a string, return

the integer

num

without trailing zeros as a string

.

Example 1:

Input:

num = "51230100"

Output:

"512301"

Explanation:

Integer "51230100" has 2 trailing zeros, we remove them and return integer "512301".

Example 2:

Input:

num = "123"

Output:

"123"

Explanation:

Integer "123" has no trailing zeros, we return integer "123".

Constraints:

1 <= num.length <= 1000

num

consists of only digits.

num

doesn't have any leading zeros.

Code Snippets

C++:

```
class Solution {
public:
```

```
string removeTrailingZeros(string num) {

}
};
```

Java:

```
class Solution {
public String removeTrailingZeros(String num) {

}
}
```

Python3:

```
class Solution:
def removeTrailingZeros(self, num: str) -> str:
```

Python:

```
class Solution(object):
def removeTrailingZeros(self, num):
"""
:type num: str
:rtype: str
"""
```

JavaScript:

```
/**
* @param {string} num
* @return {string}
*/
var removeTrailingZeros = function(num) {

};
```

TypeScript:

```
function removeTrailingZeros(num: string): string {

};
```

C#:

```
public class Solution {
public string RemoveTrailingZeros(string num) {

}
}
```

C:

```
char* removeTrailingZeros(char* num) {

}
```

Go:

```
func removeTrailingZeros(num string) string {

}
```

Kotlin:

```
class Solution {
fun removeTrailingZeros(num: String): String {

}
}
```

Swift:

```
class Solution {
func removeTrailingZeros(_ num: String) -> String {

}
}
```

Rust:

```
impl Solution {
pub fn remove_trailing_zeros(num: String) -> String {

}
}
```

Ruby:

```
# @param {String} num
# @return {String}
def remove_trailing_zeros(num)

end
```

PHP:

```
class Solution {

/**
* @param String $num
* @return String
*/
function removeTrailingZeros($num) {

}
}
```

Dart:

```
class Solution {
String removeTrailingZeros(String num) {

}
}
```

Scala:

```
object Solution {
def removeTrailingZeros(num: String): String = {

}
}
```

Elixir:

```
defmodule Solution do
@spec remove_trailing_zeros(num :: String.t) :: String.t
def remove_trailing_zeros(num) do
```

```
end
end
```

Erlang:

```
-spec remove_trailing_zeros(Num :: unicode:unicode_binary()) ->
unicode:unicode_binary().
remove_trailing_zeros(Num) ->
.
```

Racket:

```
(define/contract (remove-trailing-zeros num)
(-> string? string?)
)
```

Solutions

C++ Solution:

```
/*
* Problem: Remove Trailing Zeros From a String
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

class Solution {
public:
string removeTrailingZeros(string num) {

}
};
```

Java Solution:

```
/**
* Problem: Remove Trailing Zeros From a String
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

class Solution {
public String removeTrailingZeros(String num) {

}
}
```

Python3 Solution:

```
"""
Problem: Remove Trailing Zeros From a String
Difficulty: Easy
Tags: string

Approach: String manipulation with hash map or two pointers
Time Complexity: O(n) or O(n log n)
Space Complexity: O(1) to O(n) depending on approach
"""

class Solution:
def removeTrailingZeros(self, num: str) -> str:
# TODO: Implement optimized solution
pass
```

Python Solution:

```
class Solution(object):
def removeTrailingZeros(self, num):
"""
:type num: str
:rtype: str
"""
```

JavaScript Solution:

```
/**
* Problem: Remove Trailing Zeros From a String
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

/**
* @param {string} num
* @return {string}
*/
var removeTrailingZeros = function(num) {

};
```

TypeScript Solution:

```
/**
* Problem: Remove Trailing Zeros From a String
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

function removeTrailingZeros(num: string): string {

};
```

C# Solution:

```
/*
* Problem: Remove Trailing Zeros From a String
* Difficulty: Easy
* Tags: string
*
```

```
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

public class Solution {
public string RemoveTrailingZeros(string num) {

}
}
```

C Solution:

```
/*
* Problem: Remove Trailing Zeros From a String
* Difficulty: Easy
* Tags: string
*
* Approach: String manipulation with hash map or two pointers
* Time Complexity: O(n) or O(n log n)
* Space Complexity: O(1) to O(n) depending on approach
*/

char* removeTrailingZeros(char* num) {

}
```

Go Solution:

```
// Problem: Remove Trailing Zeros From a String
// Difficulty: Easy
// Tags: string
//
// Approach: String manipulation with hash map or two pointers
// Time Complexity: O(n) or O(n log n)
// Space Complexity: O(1) to O(n) depending on approach

func removeTrailingZeros(num string) string {

}
```

Kotlin Solution:

```
class Solution {
fun removeTrailingZeros(num: String): String {

}
}
```

Swift Solution:

```
class Solution {
func removeTrailingZeros(_ num: String) -> String {

}
}
```

Rust Solution:

```
// Problem: Remove Trailing Zeros From a String
// Difficulty: Easy
// Tags: string
//
// Approach: String manipulation with hash map or two pointers
// Time Complexity: O(n) or O(n log n)
// Space Complexity: O(1) to O(n) depending on approach

impl Solution {
pub fn remove_trailing_zeros(num: String) -> String {

}
}
```

Ruby Solution:

```
# @param {String} num
# @return {String}
def remove_trailing_zeros(num)

end
```

PHP Solution:

```
class Solution {

/**
* @param String $num
* @return String
*/
function removeTrailingZeros($num) {

}
}
```

Dart Solution:

```
class Solution {
String removeTrailingZeros(String num) {

}
}
```

Scala Solution:

```
object Solution {
def removeTrailingZeros(num: String): String = {

}
}
```

Elixir Solution:

```
defmodule Solution do
@spec remove_trailing_zeros(num :: String.t) :: String.t
def remove_trailing_zeros(num) do

end
end
```

Erlang Solution:

```
-spec remove_trailing_zeros(Num :: unicode:unicode_binary()) ->
unicode:unicode_binary().
remove_trailing_zeros(Num) ->
.
```

Racket Solution:

```
(define/contract (remove-trailing-zeros num)
(-> string? string?)
)
```