Problem 168: Excel Sheet Column Title

Problem Information

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

Problem Description

Given an integer

columnNumber

, return

its corresponding column title as it appears in an Excel sheet

.

For example:

A -> 1 B -> 2 C -> 3 ... Z -> 26 AA -> 27 AB -> 28 ...

Example 1:

Input:

columnNumber = 1

Output:

"A"

Example 2:

Input:

columnNumber = 28

Output:

"AB"

Example 3:

Input:

columnNumber = 701

Output:

"ZY"

Constraints:

1 <= columnNumber <= 2

31

- 1

Code Snippets

C++:

```
class Solution {
public:
string convertToTitle(int columnNumber) {

}
};
```

Java:

```
class Solution {
public String convertToTitle(int columnNumber) {

}
}
```

Python3:

```
class Solution:
def convertToTitle(self, columnNumber: int) -> str:
```

Python:

```
class Solution(object):
def convertToTitle(self, columnNumber):
"""
:type columnNumber: int
:rtype: str
"""
```

JavaScript:

```
/**
* @param {number} columnNumber
* @return {string}
*/
var convertToTitle = function(columnNumber) {

};
```

TypeScript:

```
function convertToTitle(columnNumber: number): string {

};
```

C#:

```
public class Solution {
public string ConvertToTitle(int columnNumber) {

}
}
```

C:

```
char* convertToTitle(int columnNumber) {

}
```

Go:

```
func convertToTitle(columnNumber int) string {

}
```

Kotlin:

```
class Solution {
fun convertToTitle(columnNumber: Int): String {

}
}
```

Swift:

```
class Solution {
func convertToTitle(_ columnNumber: Int) -> String {

}
}
```

Rust:

```
impl Solution {
pub fn convert_to_title(column_number: i32) -> String {

}
}
```

Ruby:

```
# @param {Integer} column_number
# @return {String}
def convert_to_title(column_number)

end
```

PHP:

```
class Solution {

/**
* @param Integer $columnNumber
* @return String
*/
function convertToTitle($columnNumber) {

}
}
```

Dart:

```
class Solution {
String convertToTitle(int columnNumber) {

}
}
```

Scala:

```
object Solution {
def convertToTitle(columnNumber: Int): String = {

}
}
```

Elixir:

```
defmodule Solution do
@spec convert_to_title(column_number :: integer) :: String.t
def convert_to_title(column_number) do

end
end
```

Erlang:

```
-spec convert_to_title(ColumnNumber :: integer()) ->
unicode:unicode_binary().
convert_to_title(ColumnNumber) ->
```

```
.
```

Racket:

```
(define/contract (convert-to-title columnNumber)
(-> exact-integer? string?)
)
```

Solutions

C++ Solution:

```
/*
* Problem: Excel Sheet Column Title
* Difficulty: Easy
* Tags: string, math
*
* 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 convertToTitle(int columnNumber) {

}
};
```

Java Solution:

```
/**
* Problem: Excel Sheet Column Title
* Difficulty: Easy
* Tags: string, math
*
* 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 convertToTitle(int columnNumber) {

}
}
```

Python3 Solution:

```
"""
Problem: Excel Sheet Column Title
Difficulty: Easy
Tags: string, math

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 convertToTitle(self, columnNumber: int) -> str:
# TODO: Implement optimized solution
pass
```

Python Solution:

```
class Solution(object):
def convertToTitle(self, columnNumber):
"""
:type columnNumber: int
:rtype: str
"""
```

JavaScript Solution:

```
/**
* Problem: Excel Sheet Column Title
* Difficulty: Easy
* Tags: string, math
*
* 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 {number} columnNumber
* @return {string}
*/
var convertToTitle = function(columnNumber) {

};
```

TypeScript Solution:

```
/**
* Problem: Excel Sheet Column Title
* Difficulty: Easy
* Tags: string, math
*
* 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 convertToTitle(columnNumber: number): string {

};
```

C# Solution:

```
/*
* Problem: Excel Sheet Column Title
* Difficulty: Easy
* Tags: string, math
*
* 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 ConvertToTitle(int columnNumber) {

}
}
```

C Solution:

```
/*
* Problem: Excel Sheet Column Title
* Difficulty: Easy
* Tags: string, math
*
* 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* convertToTitle(int columnNumber) {

}
```

Go Solution:

```
// Problem: Excel Sheet Column Title
// Difficulty: Easy
// Tags: string, math
//
// 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 convertToTitle(columnNumber int) string {

}
```

Kotlin Solution:

```
class Solution {
fun convertToTitle(columnNumber: Int): String {

}
```

```
}
```

Swift Solution:

```
class Solution {
func convertToTitle(_ columnNumber: Int) -> String {

}
}
```

Rust Solution:

```
// Problem: Excel Sheet Column Title
// Difficulty: Easy
// Tags: string, math
//
// 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 convert_to_title(column_number: i32) -> String {

}
}
```

Ruby Solution:

```
# @param {Integer} column_number
# @return {String}
def convert_to_title(column_number)

end
```

PHP Solution:

```
class Solution {

/**
* @param Integer $columnNumber
* @return String
```

```
*/
function convertToTitle($columnNumber) {

}
}
```

Dart Solution:

```
class Solution {
String convertToTitle(int columnNumber) {

}
}
```

Scala Solution:

```
object Solution {
def convertToTitle(columnNumber: Int): String = {

}
}
```

Elixir Solution:

```
defmodule Solution do
@spec convert_to_title(column_number :: integer) :: String.t
def convert_to_title(column_number) do

end
end
```

Erlang Solution:

```
-spec convert_to_title(ColumnNumber :: integer()) ->
unicode:unicode_binary().
convert_to_title(ColumnNumber) ->
.
```

Racket Solution:

```
(define/contract (convert-to-title columnNumber)
(-> exact-integer? string?)
)
```