DiagramsTemplate

11 August 2025 16:25

Visualization (for n=5):

```
text

Row 0:        1
             / \
Row 1:      1   1
           / \ / \
Row 2:    1   2   1
         / \ / \ / \
Row 3: 1   3   3   1
      / \ / \ / \ / \
Row 4:1   4   6   4   1
```

n=5

Pascal's Triangle Index Structure

```
text

Row 0:      [0]
           / \
Row 1:   [0] [1]
        / \ / \
Row 2: [0] [1] [2]
      / \ / \ / \
Row 3:[0][1][2][3]
     / \/ \/ \/ \
Row4:[0][1][2][3][4]
```

```
oldarr = [1]
triangle = [oldarr]

for (rowindex = 1; rowindex < 5; rowindex++):
  newarr = new array of size (rowindex + 1)

  for (col = 0; col <= rowindex; col++):
    if (col == 0 or col == rowindex):
      newarr[col] = 1
    else:
      newarr[col] = oldarr[col] + oldarr[col-1]

  triangle.append(newarr)
  oldarr = newarr
```

Fibonacci Series

```
Step 0: [a=0]—fib(0)
        [b=1]—fib(1)
        |
Step 1: ├─[a=1, b=1]—fib(2)
        |   |
Step 2: |   ├─[a=1, b=2]—fib(3)
        |   |   |
Step 3: |   |   ├─[a=2, b=3]—fib(4)
        |   |   |   |
Step 4: |   |   |   └─[a=3, b=5]—fib(5)
```

0 1 1 2 3

```
FUNCTION fibonacci(n):
  IF n <= 0:
    RETURN "Invalid input"  // Handle invalid
cases
  ELSE IF n == 1:
    RETURN [0]          // First term is 0
  ELSE IF n == 2:
    RETURN [0, 1]       // First two terms

  sequence = [0, 1]       // Initialize with first
two terms

  FOR i FROM 2 TO n-1:
    next_term = sequence[i-1] + sequence[i-2]
    sequence.APPEND(next_term)

  RETURN sequence
```

Write

1
2
3
4
6
9
12
18