Competitive Programming

Character Series Patterns

B.Bhuvaneswaran, AP (SG) / CSE

9791519152

bhuvaneswaran@rajalakshmi.edu.in

RAJALAKSHMI ENGINEERING COLLEGE

An AUTONOMOUS Institution

Affiliated to ANNA UNIVERSITY, Chennai

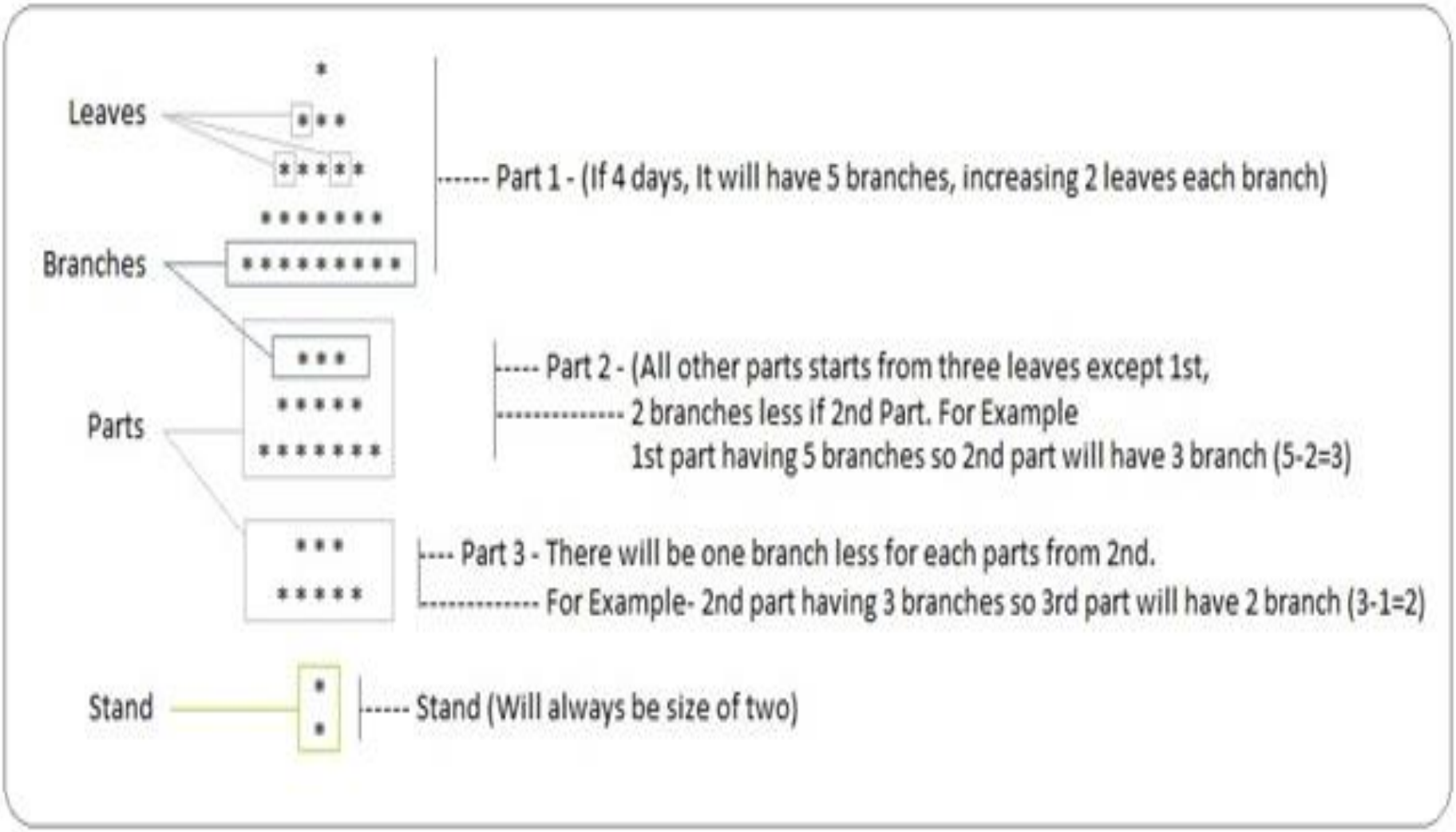
Christmas Tree

- The Christmas tree is comprised of the following:
  - Parts
  - Stand
- Each Part is further comprised of Branches. Branches are comprised of Leaves.
- How the tree appears as a function of days should be understood.
- Basis that print the tree as it appears on the given day.
- Below are the rules that govern how the tree appears on a given day.
- Write a program to generate such a Christmas tree whose input is number of days.

Christmas Tree

- If tree is one day old you cannot grow. Print a message “You cannot generate Christmas tree”.
- Tree will die after 20 days; it should give a message “Tree is no more”.
- Tree will have one part less than the number of days. E.g.
  - On 2nd day tree will have 1 part and one stand.
  - On 3rd day tree will have 2 parts and one stand
  - On 4th day tree will have 3 parts and one stand and so on.
- Top-most part will be the widest and bottom-most part will be the narrowest.
- Difference in number of branches between top-most and second from top will be 2
- Difference in number of branches between second from top and bottom-most part will be 1

Christmas Tree



Sample input and output

Input:

2

Output:

```
    *
  ***
*****
    *
    *
```

Sample input and output

Input:

3

Output:

```
    *
   ***
  *****
*******
   ***
  *****
    *
    *
```

Sample input and output

Input:

5

Output:

```
      *
     ***
    *****
   *******
  *********
***********
     ***
    *****
   *******
  *********
     ***
    *****
   *******
     ***
    *****
      *
      *
```

Constraints

- Edge cases: <=1, >20
- Stand is of height 2
- What is the structure?

Divide into small parts

- How many full triangle?
- How many number of rows for full triangle?
- How many spaces for full triangle?
  - Print the full triangle
- How many partial triangles?
- How many number of rows for partial triangle?
- How many spaces for partial triangle?
  - Print the partial triangle
- What is the height of the stand?
  - Print the stand

How many full triangle?

- Input 2 → 1 full triangle, no partial triangle
- Input 3 → 1 full triangle + 1 partial triangle

Input n

- 1 full triangle of n + 1 rows
- Partial triangles of decreasing rows

Example: Input 3

Output:

```
    *
   ***
  *****
*******
   ***
  *****
    *
    *
```

Example: Input 3

- Full triangle of 4 rows
- Partial triangle of 2 rows
- Stand

Example: Input 7

Output:

```
        *
       ***
      *****
     *******
    *********
   ***********
  *************
***************
       ***
      *****
     *******
    *********
   ***********
  *************
       ***
      *****
     *******
    *********
   ***********
       ***
      *****
     *******
    *********
       ***
      *****
     *******
       ***
      *****
        *
        *
```

Example: Input 7

- Full triangle of 8 rows
- Spaces + Partial triangle of 6 rows
- Spaces + Partial triangle of 5 rows
- Spaces + Partial triangle of 4 rows
- Spaces + Partial triangle of 3 rows
- Spaces + Partial triangle of 2 rows
- Stand

Example: Input 7

- Total triangle → 6 (N-2+1)
- Full triangle → 1
- Partial triangle → 5 (N-2)
- Stand → 1

Spaces in partial triangle? (Input: 7)

- Full triangle of 8 rows
- 1 Space offset + Partial triangle of 6 rows
- 2 Space offset + Partial triangle of 5 rows
- 3 Space offset + Partial triangle of 4 rows
- 4 Space offset + Partial triangle of 3 rows
- 5 Space offset + Partial triangle of 2 rows
- Stand

Spaces in partial triangle? (Input: 3)

Input:

3

Output:

```
     *
    ***
   *****
*******
-  ***
-*****
     *
     *
```

Logic (Input: n)

- 1 * Full triangle (n + 1) rows
- Space offsets + (n - 2) * Partial triangles
- Stand

Pseudo Code

```
offset = 1
print_full_triangle(n + 1)
for j = n - 1 to 2:
        print_partial_triangle(j, offset)
        offset = offset + 1
print_stand(n)
```

Printing a full triangle: 4 rows

Output:

```
---*
--***
-*****
*******
```

Logic

- (row - i) * spaces
- (2i - 1) "*"

Pseudo code: print_full_triangle(row)

```
for i = 1 to row:
        print (row - i) * spaces
        print (2i - 1) "*"
        print newline
```

Printing a partial triangle: 2 rows

Output:

```
  ***

*****
```

Logic

- (row - i) * spaces
- (2i + 1) "*"

Pseudo code: print_partial_triangle(row)

```
for i = 1 to row:
        print (row - i) * spaces
        print (2i + 1) "*"
        print newline
```

Printing a partial triangle: 2 rows

Output:

_ ***

_*****

Logic

- Offset * spaces
- (row - i) * spaces
- (2i + 1) "*"

Pseudo code: print_partial_triangle(row)

```
for i = 1 to row:
        print offset * spaces
        print (row - i) * spaces
        print (2i + 1) "*"
        print newline
```

Printing the stand

- Always of height 2
- Each row has one "*"

Logic

- n * spaces

Pseudo code: print_stand(n)

```
for i = 1 to 2
        print n * spaces
        print "*"
        print newline
```

Queries?

Thank You…!